

**DESIGN AND IMPLEMENTATION OF A VISION SYSTEM FOR  
MICROASSEMBLY WORKSTATION**

by  
**ERAY DOĞAN**

Submitted to the Graduate School of Engineering and Natural Sciences  
in partial fulfillment of  
the requirements for the degree of  
Master of Science  
Sabancı University  
Spring 2006

DESIGN AND IMPLEMENTATION OF A VISION SYSTEM FOR  
MICROASSEMBLY WORKSTATION

APPROVED BY

Assoc. Prof. Dr. Mustafa ÜNEL .....  
(Thesis Advisor)

Prof. Dr. Asif ŞABANOVIĆ .....  
(Thesis Co-Advisor)

Assist. Prof. Dr. Kemalettin ERBATUR .....

Assist. Prof. Dr. Güllü Kızıлтаş ŞENDUR .....

Assist. Prof. Dr. Hakan ERDOĞAN .....

DATE OF APPROVAL: .....

©Eray Dođan 2006  
All Rights Reserved

*to my father*  
*He is always with me...*

## Acknowledgements

I would like to express my gratitude to everyone who contributed in any way to the completion of this thesis. First of all, I would like to express my deepest gratitude to Assoc. Prof. Dr. Mustafa Ünel, who was much more than a thesis advisor. His research enthusiasm and the discussions, not only about research but also life itself, at his room were always very exciting and enlightening for me.

I would also like to thank Prof. Dr. Asif Şabanoviç for his support to me since freshman year of my undergraduate education. Whenever I need to talk or discuss, he was there to listen. Among all, the conversation before my engagement was the greatest one I ever had.

I would like to thank The Scientific & Technological Research Council of Türkiye (TÜBİTAK) for its support to my Master Degree education.

Among all members of the Faculty of Engineering and Natural Sciences, I would gratefully acknowledge Assist. Prof. Dr. Kemalettin Erbatur, Assist. Prof. Dr. Güllü Kızıldaş Şendur and Assist. Prof. Dr. Hakan Erdoğan for spending their valuable time to serve as my jurors.

I would like to thank every single graduate student, whom I know, at Mechatronics Program one by one and İlker Abi. Especially, I would like to single out Emrah Deniz Kunt and Ahmet Altınışık for being more like brothers to me since we came to Sabancı University in 1999 (Our louder conversations were never understood by others as normal discussions among good friends); Teoman Naskali for his way of seeing life as networking and his magical hands as massage tools; Kazım Çakır for his engineering point of view in the face of every single incident; Burak Yılmaz for motivating me to do my thesis experiments as I hope I did during his thesis progress; Nusrettin Güleç for his wiser usage of Latex; Erol Özgür for helping me with his good skills in computer and vision.

Very special thanks go to my fiancée, Ahu Ceylan. Her support could not be signified by words. I see myself the most fortunate man in the world because of having her love.

Finally, I am deeply grateful to my beloved family. There is not a comparable thing in this world with what my mother, Berna, has done and she is still doing for me. She was both my mother and my father after my father's death. I would also like to thank my sister Seden and brother-in-law Cem for their support in every instant of my life.

# DESIGN AND IMPLEMENTATION OF A VISION SYSTEM FOR MICROASSEMBLY WORKSTATION

Eray DOĞAN

Thesis Advisor: Assoc. Prof. Dr. Mustafa ÜNEL

Keywords: Microassembly workstation, micromanipulation, vision based control, camera calibration, hough transform, elliptic fourier descriptors, autofocusing, motion parameter estimation, intrinsic image estimation and image deblurring

## **Abstract**

Rapid development of micro/nano technologies and the evolvement of biotechnology have led to the research of assembling micro components into complex microsystems and manipulation of cells, genes or similar biological components. In order to develop advanced inspection/handling systems and methods for manipulation and assembly of micro products and micro components, robust micromanipulation and microassembly strategies can be implemented on a high-speed, repetitive, reliable, reconfigurable, robust and open-architecture microassembly workstation.

Due to high accuracy requirements and specific mechanical and physical laws which govern the microscale world, micromanipulation and microassembly tasks require robust control strategies based on real-time sensory feedback. Vision as a passive sensor can yield high resolutions of micro objects and micro scenes along with a stereoscopic optical microscope. Visual data contains useful information for micromanipulation and microassembly tasks, and can be processed using various image processing and computer vision algorithms.

In this thesis, the initial work on the design and implementation of a vision system for microassembly workstation is introduced. Both software and hardware issues are considered. Emphasis is put on the implementation of computer vision algorithms and vision based control techniques which help to build strong basis for the vision part of the microassembly workstation. The main goal of designing such a vision system is to perform automated micromanipulation and microassembly

tasks for a variety of applications. Experiments with some teleoperated and semi-automated tasks, which aim to manipulate micro particles manually or automatically by microgripper or probe as manipulation tools, show quite promising results.

# MİKROMONTAJ İŞ İSTASYONU İÇİN BİR GÖRÜ SİSTEMİ DİZAYNI VE UYGULAMASI

Eray DOĞAN

Tez Danışmanı: Doç. Dr. Mustafa ÜNEL

Anahtar Kelimeler: Mikromontaj İş İstasyonu, mikroişlem, görü tabanlı kontrol, kamera kalibrasyonu, hough dönüşümü, eliptik fourier tanımlıyıcıları, otomatik odaklama, hareket parametre hesaplaması, içkin görüntü hesaplaması ve görüntü bulanıklığının giderilmesi

## Özet

Mikro/nano teknolojilerdeki hızlı gelişme ve biyoteknolojideki evrimleşme, mikro elemanların kompleks mikrosistemler içine montajı ve hücrelerin, genlerin veya benzeri biyolojik elemanların işlenmesi alanlarındaki araştırmalara öncülük etmiştir. Mikro ürünlerin ve mikro elemanların işlenmesi ve montajı için gelişmiş denetleme/uygulama sistemleri ve metodları geliştirmek için yüksek hızlı, tekrarlanan, güvenilir, gürbüz, serbest yapılı ve tekrar ayarlanabilen mikromontaj iş istasyonu gürbüz mikromanipülasyon ve mikromontaj stratejileri ile uygulanabilir.

Doğruluk gereksinimlerinden ve mikroölçek dünyasını yöneten özgül mekanik ve fiziksel yasalardan dolayı, mikromanipülasyon ve mikromontaj görevleri gerçek zamanlı algılayıcı geri beslemesi tabanlı gürbüz kontrol stratejilerine gereksinim duyarlar. Pasif algılayıcı olarak görü, stereoskopik mikroskop ile beraber, mikro nesnelerin ve mikro sahnelerin yüksek çözünürlüklerini elde eder. Görsel veriler, mikromanipülasyon ve mikromontaj görevleri için yararlı bilgi içerir. Görsel veriler, ayrıca görüntü işleme ve bilgisayarlı görü algoritmalarında uygulanabilirler.

Bu tezde, mikromontaj iş istasyonu için bir görü sistemi tasarımı ve uygulaması hakkında başlangıç olarak yapılan çalışmalar tanıtılmıştır. Hem yazılım hem de donanım meselelerine değinilmiştir. Vurgu, bilgisayarlı görü ve görü tabanlı kontrol tekniklerinin uygulamaları üzerine konulmuştur. Bu teknikler mikromontaj iş istasyonunun görü sistemi için sağlam bir temel inşa edilmesine yardımcı olur.

Böyle bir görü sistemi tasarımı yapılmasındaki asıl amaç, çeşitli uygulamalar için otomatikleştirilmiş mikromanipülasyon ve mikromontaj görevlerini gerçekleyebilmektir. Birkaç, mikro parçacıkların elle veya otomatik olarak mikro tutgaç (microgripper) veya mikro iğne (probe) ile manipüle edilmesini amaçlayan, uzaktan kumanda (teleoperated) ve yarı otomatik (semi-automated) mikromanipülasyon ve mikromontaj görev deneyleri, umut verici sonuçlar alınarak gerçekleştirilmiştir.

# Table of Contents

Acknowledgements	v
Abstract	vii
Özet	ix
<b>1 Introduction</b>	<b>1</b>
<b>2 Vision Fundamentals</b>	<b>4</b>
2.1 Imaging Optics . . . . .	5
2.2 Camera Imaging Geometry . . . . .	6
2.3 Camera Calibration . . . . .	8
2.3.1 Extrinsic Parameters . . . . .	8
2.3.2 Intrinsic Parameters . . . . .	9
2.4 Image Features . . . . .	10
2.4.1 Edge Detection . . . . .	11
2.4.2 Hough Transform . . . . .	13
2.4.3 Elliptic Fourier Descriptors . . . . .	16
2.5 Motion Parameter Estimation . . . . .	18
2.5.1 Motion Models . . . . .	18
2.5.2 Planar Structure . . . . .	19
2.5.3 Composite Image Feature Dynamics . . . . .	20
2.6 Autofocusing . . . . .	21
<b>3 Vision Based Control</b>	<b>24</b>
3.1 Static Look & Move Control Structure . . . . .	25
3.2 Dynamic Look & Move Control Structure . . . . .	27
3.3 Visual Task Functions . . . . .	28
<b>4 Experimental Setup of MAW</b>	<b>29</b>
4.1 Software Setup of MAW . . . . .	29
4.1.1 Motion Control Software . . . . .	30
4.1.2 Vision Software . . . . .	31

4.2	Hardware Setup of MAW . . . . .	35
4.2.1	Vision System . . . . .	36
4.2.2	Control Unit . . . . .	45
4.2.3	Motion Unit . . . . .	46
4.2.4	Manipulation Unit . . . . .	48
4.2.5	Teleoperation Unit . . . . .	51
<b>5</b>	<b>Implementation of Vision System on MAW</b>	<b>53</b>
5.1	Camera Calibration . . . . .	53
5.1.1	Formulation of Camera Calibration . . . . .	53
5.1.2	Implementation of Camera Calibration . . . . .	56
5.1.3	Discussions . . . . .	60
5.2	Image Features . . . . .	61
5.2.1	Circular Hough Transform . . . . .	61
5.2.2	Elliptic Fourier Descriptors . . . . .	64
5.2.3	Discussions . . . . .	66
5.3	Automation Task Applications on MAW . . . . .	67
5.3.1	Teleoperated Tasks . . . . .	67
5.3.2	Semi-Automated Tasks . . . . .	69
5.3.3	Discussions . . . . .	71
5.4	Motion Parameter Estimation . . . . .	72
5.4.1	Formulation of Sliding Mode Observers . . . . .	74
5.4.2	Implementation of Motion Parameter Estimation . . . . .	75
5.4.3	Discussions . . . . .	78
5.5	Autofocusing . . . . .	78
5.5.1	First Case . . . . .	80
5.5.2	Second Case . . . . .	81
5.5.3	Discussions . . . . .	82
5.6	Intrinsic Image Estimation & Image Deblurring . . . . .	82
5.6.1	Intrinsic Image Estimation . . . . .	84
5.6.2	Image Deblurring . . . . .	87
5.6.3	Discussions . . . . .	90
<b>6</b>	<b>Conclusions</b>	<b>92</b>
	<b>Bibliography</b>	<b>94</b>

## List of Figures

2.1	Thin Lens Model . . . . .	6
2.2	Perspective Projection of a 3-D Point . . . . .	7
2.3	Real and Edge Images of Micro Particles with Different Magnifications	12
2.4	Mapping Between $x-y$ Space and $a-b$ Space . . . . .	14
2.5	Polar Form of a Line . . . . .	14
2.6	Mapping for Circular Hough Transform . . . . .	15
2.7	The Idea of EFDs . . . . .	16
3.1	Static Look & Move Control Structure . . . . .	25
3.2	Dynamic Look & Move Control Structure . . . . .	27
4.1	Experimental Setup Chart of MAW . . . . .	29
4.2	Software Structure of MAW . . . . .	30
4.3	User Interface of Vision Software . . . . .	32
4.4	Camera View Windows . . . . .	35
4.5	Hardware Structure of MAW . . . . .	37
4.6	Vision System Hardware Structure . . . . .	39
4.7	Stereoscopic Optical Microscope . . . . .	40
4.8	Coarse and Fine View Digital Cameras . . . . .	42
4.9	Transparent Micro particles Obtained Using Reflected Light Illumi- nation . . . . .	43
4.10	Micro particles Obtained Using Backlighting Illumination . . . . .	44
4.11	Auto Focus Device . . . . .	45
4.12	Hardware Equipment of Control Unit . . . . .	46
4.13	Motion unit of Manipulation Tool . . . . .	47
4.14	Motion unit of Manipulation Surface . . . . .	48

4.15 Manipulation Unit . . . . .	49
4.16 Pick & Place Manipulation Tool: Microgripper . . . . .	49
4.17 Pushing Manipulation Tools . . . . .	50
4.18 Teleoperation Unit & Pantograph (5 Bar Linkage) Mechanism . . . . .	51
5.1 Micro Pattern Images captured from Coarse and Fine Cameras for 2× Magnification . . . . .	57
5.2 Determination of Region of Interest and Extraction of Edge Points . . . . .	57
5.3 Detected Lines by Hough Line Detection Algorithm . . . . .	58
5.4 Detected Lines by Point-to-Line Line Fitting Algorithm . . . . .	58
5.5 Detected Corner Points by Point-to-Line Fitting Algorithm . . . . .	59
5.6 Circular Hough Transform of Micro Particles (4× Magnification) . . . . .	62
5.7 Edge Image and Circular Hough Transform of 2 Aggregated Micro Particles (5× Magnification) . . . . .	63
5.8 Circular Hough Transform of 4 Aggregated Micro Particles (7× Mag- nification) . . . . .	64
5.9 EFD of Micro Particles (4× Magnification) . . . . .	65
5.10 EFD of 2 Aggregated Micro Particles (5× Magnification) . . . . .	66
5.11 Steps of Pick & Place Teleoperated Task using Microgripper . . . . .	69
5.12 Steps of Pushing Teleoperated Task using Probe . . . . .	70
5.13 Steps of Pushing Semi-Automated Task using Probe for One Micro Particle . . . . .	72
5.14 Steps of Making a Line Pattern by Semi-Automated Tasks . . . . .	73
5.15 Estimated Motion along $x$ - and $y$ -axes . . . . .	76
5.16 Actual (red) and Estimated (blue) Angular Velocities, $\omega$ and $\hat{\omega}$ . . . . .	77
5.17 Actual (black) and Estimated (blue) Angular Velocities, $\omega$ and $\hat{\omega}$ . . . . .	77
5.18 Estimated Motion Trajectory . . . . .	78
5.19 Focal Planes of The Manipulation Tool and The Micro Particles for the First Case . . . . .	80
5.20 The Normalized Variance Graph of the First Case . . . . .	81
5.21 Focal Planes of The Manipulation Tool and The Micro Particles for the Second Case . . . . .	81
5.22 The Normalized Variance Graph of the Second Case . . . . .	82

5.23	Micro Pattern Images from Coarse & Fine Cameras . . . . .	84
5.24	Decomposition of Intrinsic Images from [64] . . . . .	85
5.25	Sample Microscopic Images (Fixed Reflectivity & Distinct Illumination)	85
5.26	Sample Gradient Images . . . . .	86
5.27	Estimated Reflectivity Image . . . . .	87
5.28	Sample Illumination Images calculated by Estimated Reflectivity Image	88
5.29	Resultant Image calculated by Tikhonov Regularization . . . . .	88
5.30	Resultant Image calculated by Edge-Preserving Regularization . . . . .	89

## List of Tables

4.1	Stereoscopic Optical Microscope Specifications . . . . .	40
4.2	$\mu m$ Field of View of Coarse and Fine Camera Views for Each Magnification . . . . .	41
5.1	Camera Matrices and Error Values of Pixel and World Coordinates for $2\times$ Magnification . . . . .	60
5.2	$\mu m$ Measurement of One Pixel for All Magnifications . . . . .	60
5.3	Pixel and $\mu m$ Errors for Image and World Coordinates for Each Magnification . . . . .	61
5.4	Radius and Center of the Micro Particles by Circular Hough Transform ( $4\times$ Magnification) . . . . .	62
5.5	Radius and Center of 2 Aggregated Micro Particles by Circular Hough Transform ( $5\times$ Magnification) . . . . .	63
5.6	Radius and Center of 4 Aggregated Micro Particles by Circular Hough Transform ( $7\times$ Magnification) . . . . .	64
5.7	EFD Coefficients of the Upper Micro Particle for Each Harmonics ( $4\times$ Magnification) . . . . .	65
5.8	EFD Coefficients of the Lower Micro Particle for Each Harmonics ( $4\times$ Magnification) . . . . .	66
5.9	EFD Coefficients of 2 Aggregated Micro Particle for Each Harmonics ( $5\times$ Magnification) . . . . .	67

## List of Abbreviations

MEMS	:	Microelectromechanical Systems
DNA	:	Deoxyribonucleic Acid
MAW	:	Microassembly Workstation
EFDs	:	Elliptic Fourier Descriptors
IBSLM	:	Image Based Static Look & Move
PBSLM	:	Position Based Static Look & Move
PBDLM	:	Position Based Dynamic Look & Move
PC	:	Personal Computer
SMC	:	Sliding Mode Control
DOF	:	Degree of Freedom
ISA	:	Industry Standard Architecture
MFC	:	Microsoft Foundation Class
fps	:	Frame Per Second
OpenCV	:	Open Computer Vision Library
CCD	:	Charged Coupled Device
PWM	:	Pulse Width Modulation
SVD	:	Singular Value Decomposition

# Chapter 1

## Introduction

Rapid development of micro and nano technologies have led to microsystems with micro components and increasingly complex micro structures. As the trend towards producing more complex microsystems continues, assembling the micro components into a functional microsystem is needed. Furthermore, the evolvement of biotechnology in the field of cell and gene engineering in last two decades gave raise to the research of manipulation in microscale world. The fact that direct human handling of the micro size parts, cells or genes is not possible, the automated micromanipulation and microassembly operations are key to success. Assembly of MEMS devices, such as miniature drug pumps, actuators, sensors, optical devices; gathering chemical compounds together, such as polymers; micro injection, manipulation of cell, embryo, nucleus, chromosome and DNA are present examples where the automated micromanipulation and microassembly operations are must in the fields of microsystems and biotechnology [1–5].

In order to develop advanced inspection/handling systems and methods for manipulation and assembly of micro products and micro components, robust micromanipulation and microassembly strategies are necessary with a high-speed, repetitive, reliable, robust, reconfigurable and open-architecture MAW. Additionally, these strategies have to overcome critical issues that are not applicable to manipulation in macroscale world.

The following two critical issues must be addressed:

- Much higher relative positioning accuracy required. In macroscale world  $10^{-3}$  meter (millimeter) range accuracies can be achieved with sensor-less robotic manipulators. Nevertheless,  $10^{-6}$  to  $10^{-9}$  meter (micrometer to nanometer)

accuracies are required for manipulation in microscale world.

- Difference in mechanics and physics of object interactions. The principle obstacle of microscale world is that gravitational force is negligible in comparison with adhesion forces such as Van der Waals, surface, electrostatic and capillary forces and there are not exact mathematical models for these forces [6].

Both of these differences conclude the necessity to use micromanipulation and microassembly strategies guided by real-time sensory feedback. In other words, closed loop control strategies are required to compensate for poor kinematic models and thermal effects in addition to extremely accurate and precise 3-D robotic systems. Whatever the employed techniques, and the environment (in air, liquid or vacuum) are, micromanipulation and microassembly systems often suffer from a lack of repeatability due to high accuracy requirements and specific mechanical and physical laws which govern the microscale world. That is why micromanipulation and microassembly tasks require robust control strategies based on real-time sensory feedback. But many kinds of sensors, which are used in macroscale world, have insufficient resolution and size considering to the environment in microscale world. The vision sensor can yield high resolutions of micro objects and micro scenes along with stereoscopic optical microscope. Visual data contains useful information for micromanipulation and microassembly tasks, and can be processed using variety of image processing and computer vision algorithms.

As it is mentioned above, visual sensing plays an important role in assembly and manipulation in microscale, both in the motion control of MAW actuators, as well as in position and orientation of micro parts and other MAW features. It has been verified that visual sensing as feedback has many advantages when it is used in the control loop of robotic systems. First of all, vision is a non-contact sensing modality, in other words a passive sensor, that provides dense information about the environment and enables recognition of micro objects and to microassembly that measures geometry, motion, and spatial relation at the microscale world [7]. Therefore, micromanipulation and microassembly can not be possible without seeing and sensing the scene and the micro objects. Secondly, using visual feedback in the control loop gives flexibility and robustness to the robotic systems by reducing the dependence on precise calibration of the overall system. On the other hand, the visual data extracted

from digital image captured from stereoscopic optical microscope optics is essential to high-level task understanding. Under certain circumstances, visual data can also provide non-contact force measurement by measuring object deformations [8–10].

In the light of previous discussions on why and how much important is visual information for the automated micromanipulation and microassembly tasks, it is substantiated that there is a need for accurate and precise design and implementation of a vision system for MAW. It is generally required that below micrometer range accuracies must be achieved for pixelwise measurements. Moreover, position and orientation information of specific manipulation tools or micro parts must be extracted by means of the vision system. This information, named as visual features, is used as visual feedback in the control loop of robotic system of MAW. Thus, vision based control is attained.

The objective of this thesis work is to play an initiative role in research which relates to the design and implementation of a vision system for MAW and to demonstrate that repetitive and robust visually guided automated micromanipulation and microassembly tasks with desired accuracies can be achieved with the proposed vision system.

The outline of this thesis is as follows: In Chapter 2, computer vision fundamentals, which are essential for designing of vision system for MAW, are mentioned in detail. Chapter 3 deals with vision based control techniques and their technical specifications. The experimental setup of MAW is the subject of Chapter 4. The software implementation of vision system is presented in Chapter 5 with simulations and experimental results. Chapter 6 is dedicated to conclusion and discussion about the thesis work and its contribution. It also includes comments on further works that should be done with the proposed vision system.

## Chapter 2

### Vision Fundamentals

The aim of this chapter is to give general overview about the fundamentals of computer vision. This basic information will be the basis for how and what is done as the design and implementation for vision system of MAW. A good start for this chapter can be done by defining of the problem(s) that computer vision tackles:

“*Computer vision* studies the creation and understanding of model for 3-dimensional real world from 2-dimensional *digital images*.”

Therefore, there is a strong relation between digital image processing and computer vision, because both aim to use digital images as tool. However, the content of the subjects are quite different. Digital image processing involves manipulating digital image to generate another digital image while computer vision uses the processed digital image to generate model for real world by extracting numeric or symbolic data.

At this point, one needs to understand digital image fundamentals to study computer vision. As stated earlier, a digital image is a representation of a 2-dimensional array as a finite set of digital values, which are called pixels. The types of digital images which are generally used in computer vision applications are *binary*, *grayscale*, and *color* images. These images can be grouped as “*intensity*” images. Acquisition tools for this kind of digital images are types of digital cameras. There are also “*range*” images whose pixels express the distance between a known frame and a visible point in the scene and which can be captured by sonar sensors or laser scanners [11]. In this thesis, range images will not be matter of concern because intensity images are used in the content of computer vision applications for MAW. The intensity image formation process can be examined in three aspects:

1. Optical Aspects
2. Geometrical Aspects
3. Radiometric Aspects (How much light is reflected by objects and how much reflected light reaches to image plane)

In the following section, first two aspects will be explained in detail. Radiometric formulation of a digital image will not be explained in this thesis since it is not related to the work done.

## 2.1 Imaging Optics

Computer vision generally uses the *pinhole camera model* as optical model because of its simple structure. The reason is that a single point reflects light in many different ways, and some of those rays can enter the camera. However, to be able to see this single point as sharp as possible, all entered rays must converge to a single point on the image plane. This can be provided by a pinhole camera model by reducing the camera aperture to a point or designing an optical system composed of lenses or other elements which actually tries to mimic the pinhole camera model to make all rays entering to the camera from the same 3-D point and converging onto the same point on the image plane.

Because of complexity of the standard optical systems, the basic features can be investigated from the simplest optical system, *thin lenses*. Figure 2.1 shows the elements and imaging principle of the thin lenses.

As it can be seen from Figure 2.1,  $O$  represents the center of the lens,  $f_l$  and  $f_r$  are called as left and right focus points respectively. The distance between center of the lens and focus points are named as focal length and shown as  $f$ .  $Z$  is the distance of point  $P$  from left focus point and  $z$  is the distance of point  $p$  from right focus point. When considering the similarity of the triangles, the following result is obtained:

$$Zz = f^2 \tag{2.1}$$

If we define the distances of the points of interest from center of the lens as  $\hat{Z} = Z + f$

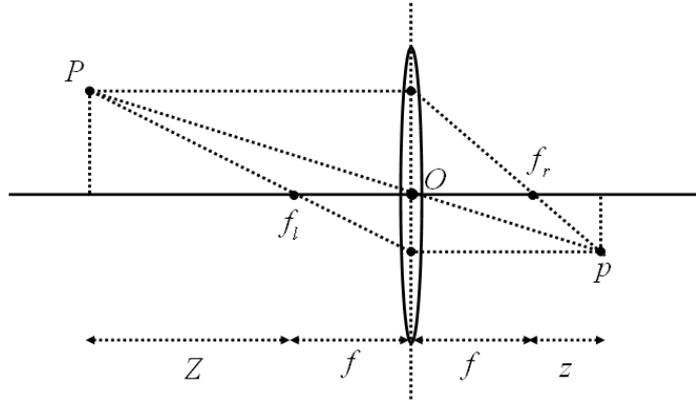


Figure 2.1: Thin Lens Model

and  $\hat{z} = z + f$ , we get the main formula for thin lenses:

$$\frac{1}{\hat{Z}} + \frac{1}{\hat{z}} = \frac{1}{f} \quad (2.2)$$

In addition to the basic principle of the thin lenses, another aspect is the concept of *field of view* which is the effective portion of the thin lenses. Let  $d$  be the effective lens diameter which is generally smaller than the physical diameter of the lens. The field of view of the lens, which is an angular measure of the portion of 3-D space actually seen by the camera shown as  $w$  and can be calculated as:

$$\tan w = \frac{d}{2f} \quad (2.3)$$

## 2.2 Camera Imaging Geometry

Once the optical system is realized, the mapping of 3-D real world onto 2-D image plane can be studied next. The location of a 3-D point  $(X, Y, Z)$ , which is defined with respect to camera reference frame, at a 2-D point  $(x, y)$  on the image plane is found by computing the coordinates  $(x, y)$  of the intersection of the line of sight passing through the scene point  $(X, Y, Z)$  with the image plane as shown in Figure 2.2 [12].

The distance of the point  $(X, Y, Z)$  from  $Z$ -axis is  $R = \sqrt{X^2 + Y^2}$ , and the distance of the projected point  $(x, y)$  from the origin of the image plane is  $r =$

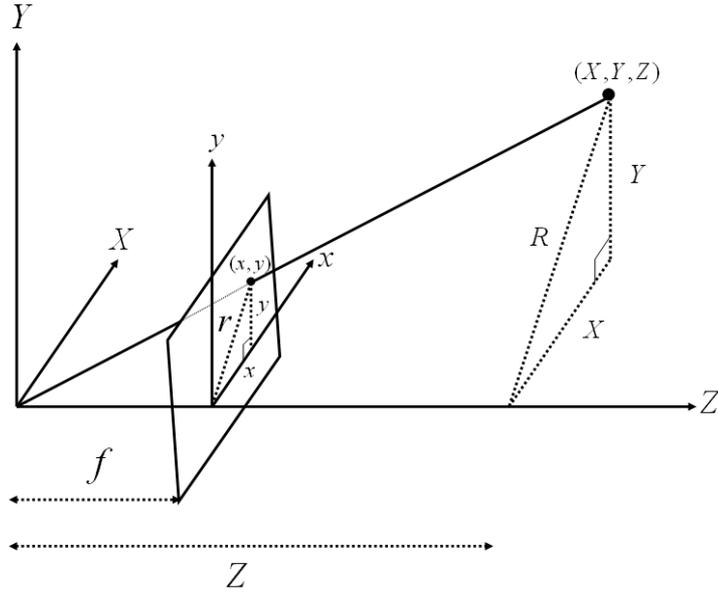


Figure 2.2: Perspective Projection of a 3-D Point

$\sqrt{x^2 + y^2}$ . The  $Z$ -axis, the line of sight to point  $(X, Y, Z)$ , and the line segment of length  $R$  from point  $(X, Y, Z)$  to the  $Z$ -axis form a triangle. The  $Z$ -axis, the line of sight to point  $(x, y)$  in the image plane, and the line segment of length  $r$  from point  $(x, y)$  to the  $Z$  axis form another triangle. From the similarity between these two triangles, following equation can be written:

$$\frac{f}{Z} = \frac{r}{R} \quad (2.4)$$

The triangle formed by the  $X$  and  $Y$  coordinates and the perpendicular distance  $R$  and the triangle formed by the image plane coordinates  $x$ ,  $y$  and the perpendicular distance  $r$  are also similar triangles. Therefore we have:

$$\frac{x}{X} = \frac{y}{Y} = \frac{r}{R} \quad (2.5)$$

Combining equations 2.4 and 2.5 yields the equations for perspective projection:

$$\frac{x}{X} = \frac{f}{Z}, \text{ and } \frac{y}{Y} = \frac{f}{Z} \quad (2.6)$$

The position of a 3-D point  $(X, Y, Z)$  in the image plane is given by the equations

$$x = \frac{f}{Z}X \quad (2.7)$$

$$y = \frac{f}{Z}Y \quad (2.8)$$

Note that when the focal length becomes quite large, i.e.  $f \rightarrow \infty$ , the perspective projection is reduced to an orthographic projection given by

$$x = X, \quad y = Y \quad (2.9)$$

When the depth of objects,  $\delta z$ , is much smaller than their average distance,  $\bar{Z}$ , from the camera along the optical axis, i.e.  $|\delta z| \ll \bar{Z}$ , the full perspective camera model can be replaced by the weak-perspective or so called scaled orthographic camera model.

$$x = f \frac{X}{\bar{Z}}, \quad y = f \frac{Y}{\bar{Z}} \quad (2.10)$$

## 2.3 Camera Calibration

As stated before, computer vision algorithms need a mapping between coordinates of a 3-D point from real world and corresponding coordinates of a 2-D image point. These equations, which are calculated in the previous section, are written in the camera frame. (See Equations 2.7 and 2.8) The camera calibration problem includes both the exterior and interior orientation problems, since the position and orientation of the camera must be determined to relate image plane coordinates to world frame (absolute) coordinates. The location of the principal points, the focal length, and the lens distortions must be determined to relate image array locations (pixel coordinates) to positions in the image plane. The camera calibration problem involves calculation of two sets of parameters: *extrinsic* and *intrinsic* camera parameters.

### 2.3.1 Extrinsic Parameters

Extrinsic parameters of the camera give location and orientation of the camera reference frame with respect to world reference frame (or some other known frame) by using only image pixel information. The transformation between camera reference frame and world reference frame can be described as follows:

- a  $3 \times 1$  translation vector,  $T$ , describing the relative positions of the origins of the two reference frames.

- a  $3 \times 3$  rotation matrix,  $R$ , an orthogonal matrix ( $R^T R = R R^T = I$ ), describing the orientation between two reference frames.

Combination of these two is known as “*Camera Extrinsic Matrix*”, and shown as:

$$M_{Ext} = [R \mid T] = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{pmatrix} \quad (2.11)$$

### 2.3.2 Intrinsic Parameters

Intrinsic parameters of the camera can be defined as the set of parameters to characterize the optical, geometric and digital characteristics of the a digital camera. Intrinsic parameters are the focal length,  $f$ , the principal points,  $o_x$  and  $o_y$ , the effective size of the pixels,  $s_x$  and  $s_y$ . Combination of these parameters into matrix form known as “*Camera Intrinsic Matrix*”, and shown as:

$$M_{Int} = \begin{pmatrix} \frac{f}{s_x} & 0 & o_x \\ 0 & \frac{f}{s_y} & o_y \\ 0 & 0 & 1 \end{pmatrix} \quad (2.12)$$

Combining 2.11 and 2.12, the total camera matrix can be found and the relation between a 3-D point from real world in the scene and a 2-D point on the image can be written as:

$$P = M_{Int} M_{Ext} \quad (2.13)$$

It follows that,

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = P \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.14)$$

and

$$x = \frac{x_1}{x_3}, \quad y = \frac{x_2}{x_3} \quad (2.15)$$

where  $x$  and  $y$  are the coordinates of the point in the image plane.

Estimation of intrinsic and extrinsic parameters of the camera, known as camera calibration problem, is implemented as explained in [13] with simple model specifically designed for microscopic vision systems in Section 5.1.

## 2.4 Image Features

Once the mapping between 3-D real world and 2-D digital images is estimated, next step in computer vision is to find global or local, meaningful, detectable part(s) of the image, which are indeed named as *image features*. These features are used in other computer vision algorithms like object recognition or tracking. Image features can be global, if our interest is towards finding, for instance, mean or standard deviation of the whole image. These features are called *global image features*. (See Sections 2.6 and 5.5) On the other hand, they can be local, if we are interested in points, lines, curves or contours that exist in some part of the image. These features are called *local image features*. (See Section 5.2)

The feature extraction is one of the main problems in computer vision because the extracted feature(s) must be meaningful in the sense that they must be relevant to estimate the model, structure or properties of an object or an instant of interest in the image. At the same time, they must be detectable in such a way that we have to be sure that what we have as image feature(s) are what we aim to have. In order to extract different features, there are different algorithms in the literature. Here, the selection of the appropriate feature extraction algorithm is significant.

As indicated above, there are several image feature extraction algorithms in computer vision literature. The selection of appropriate algorithms is depending on a priori information that you have about the scene or application that you are developing. If the interest is towards extracting the boundaries of an object, edge detection is frequently used in computer vision society. However, on the other hand, level sets or active contours are another option which is generally preferred in medical applications or hand written character recognition to extract boundaries of curves as stated in [14]. If our aim is to extract the properties of geometrical objects like lines, circles or curves, hough transform or elliptic fourier descriptors can give us good results. Of course, again that does not mean that the literature has only these algorithms for detecting lines, circles or curves, but these are appropriate for MAW implementations with a priori information we have about the overall system and possible micromanipulation and microassembly tasks.

Kasaya et al. used generalized Hough transform to detect position and orientations of manipulation tool(s) and micro objects in [15]. Fatikow et al. in [16]

used Fourier descriptors for modeling of closed curves instead of template matching because of its fast response. Ogawa et al. selected image moments to find position and orientation of one micro object in [17]. If there are two or more objects in the image, they switch to self windowing algorithm to detect each micro object. Lee et al. in [18] and Xudong et al. in [19] used pattern matching as the recognition algorithm for micro parts. Ellipse fitting algorithm is used by Arai et al. in [20]. Burkle et al. used curvature zero crossing points as meaningful image features in [21, 22] because of their robustness to occlusion.

In the following sections, the image feature extraction algorithms of our selection for implementations of MAW will be explained in detail. The implementation results of these algorithms can be found in Section 5.2.

### 2.4.1 Edge Detection

Edge detection is an algorithm to detect edge points or simply edges, in which occur generally between two regions that have sharp variations in image pixel values. Although edge detection has attracted many researchers in computer vision society for several decades, it is still an unresolved problem in general sense [23, 24]. One of the important characteristics of edges are to be connected chains which gives easiness to model lines, curves or contours. Edge detection algorithms generally is the first step of recovering boundary information of objects or regions for recognition.

There are several edge detection operators in literature such as *Prewitt*, *Sobel*, *Robert* and *Laplacian*. Among them *Canny Edge Detection Operator* gives better detection and better localization with only one response to a single edge [25, 26]. Steps in Canny Edge Detection Operator can be summarized as:

1. Compute the gradient of image  $f(x, y)$  by convolving it with the first derivative of Gaussian in  $x$  and  $x$  directions.

$$\begin{aligned} f_x(x, y) &= f(x, y) \left( \frac{-x}{\sigma} \right) e^{-\frac{(x^2+y^2)}{2\sigma^2}} \\ f_y(x, y) &= f(x, y) \left( \frac{-y}{\sigma} \right) e^{-\frac{(x^2+y^2)}{2\sigma^2}} \end{aligned} \quad (2.16)$$

2. Perform non-maxima suppression on the gradient magnitude.
3. Apply hysteresis thresholding to the non-maxima suppressed magnitude. Scan the image from left-right, top-bottom. If the gradient magnitude at the pixel

is above the high threshold, declare that as an edge point. Then recursively look at its neighbors (4 connected or 8 connected). If the gradient magnitude is above the low threshold, declare that as an edge point.

In Figure 2.3, a real image and an edge image, that is obtained using Canny edge detection operator, of micro particles whose diameters are approximately  $70 \mu m$  are displayed for different magnifications of the stereoscopic optical microscope.

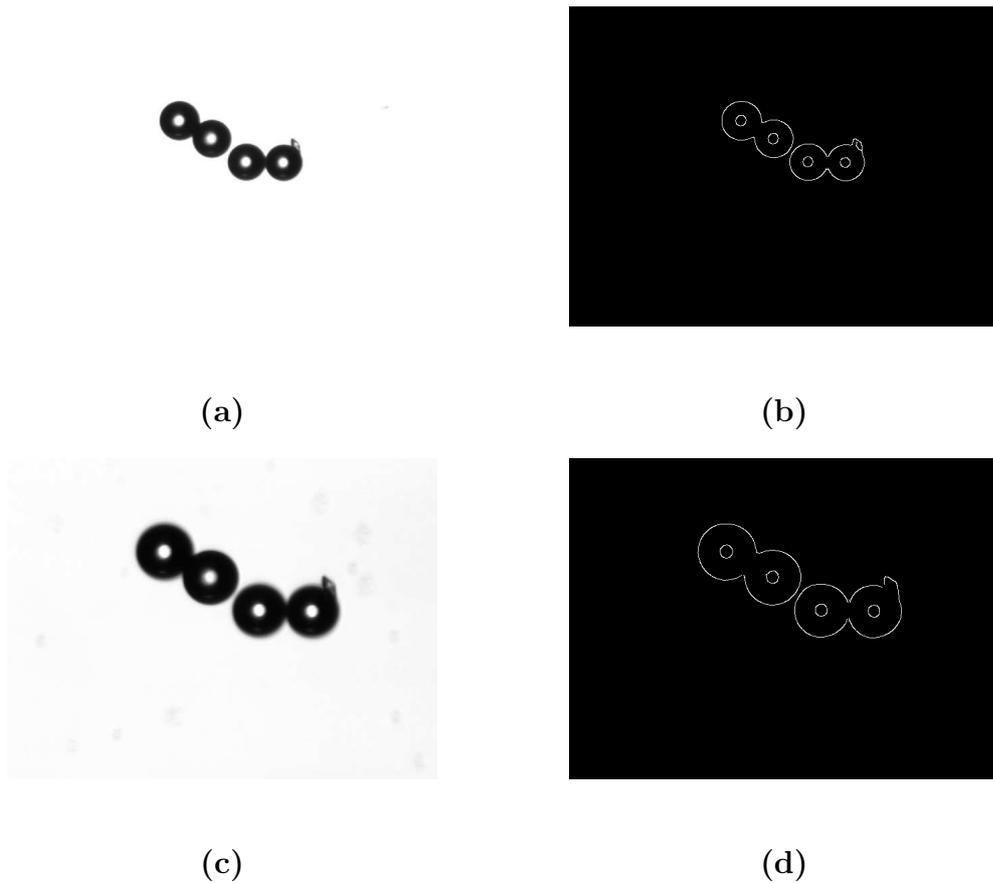


Figure 2.3: Real and Edge Images of Micro Particles with Different Magnifications: (a),(b) Real and Edge Images with  $4\times$  Mag. (c),(d) Real and Edge Images with  $7\times$  Mag.

In the following two sections, two different shape representation techniques will be explained in detail by using extracted chained connective edge points obtained from Canny edge detection operator.

## 2.4.2 Hough Transform

The Hough transform is a general technique for identifying the locations and orientations of certain types of features in a digital image. Developed by Paul Hough in 1962 and patented by *IBM*, the transform consists of parameterizing a description of a feature at any given location in the original image space. The invention of universal usage for hough transform was provided by Duda and Hart in [27]. They called the transform as *generalized hough transform*. The algorithm became very popular after the work done by Ballard in [28]. In the implementations of MAW, Hough transform is used to detect especially lines and circles because of its robustness to model these shapes. Software implementation of works done by Matas et al. in [29], and Yuen et al. in [30] are used for MAW.

### Line Detection

In this section, how Hough transform can be used to fit extracted edge points obtained by Canny edge detection algorithm to the equation of a line is described. The equation of a line is given by:

$$y = ax + b \quad (2.17)$$

where  $a$  and  $b$  are the slope of the line and ordinate of the intersection point of the line at  $y$ -axis respectively. The equation 2.17 can be rewritten as:

$$b = (-x)a + y \quad (2.18)$$

The equation 2.18 is considered as a line equation in  $a - b$  space, where  $-x$  is the slope of the line and  $y$  is ordinate of the intersection point of the line at  $b$ -axis. In fact, a point in  $x-y$  space is mapped to a line in  $a-b$  space as shown in Figure 2.4. Suppose we have  $n$  edge points which are assumed to be fit a line, as  $(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots, (x_n, y_n)$  in  $x-y$  space. Each edge point is mapped to a line in  $a-b$  space. These lines intersect at a single point  $(\bar{a}, \bar{b})$  in  $a-b$  space. This point is the estimation of the slope and ordinate of the intersection point of the line at  $y$ -axis fitted by edge points.

However above representation has an inevitable problem which occurs when the slope of the line becomes infinity,  $a = \infty$ , which means that the line is parallel to  $y$ -axis. This situation can not be represented by a computer.

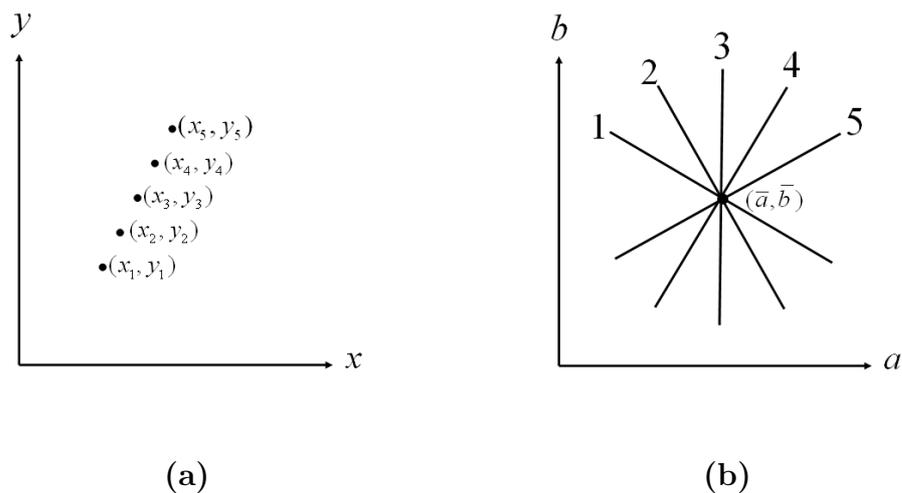


Figure 2.4: Mapping Between  $x$ - $y$  Space and  $a$ - $b$  Space: (a) Points in  $x$ - $y$  Space (b) Lines in  $a$ - $b$  Space

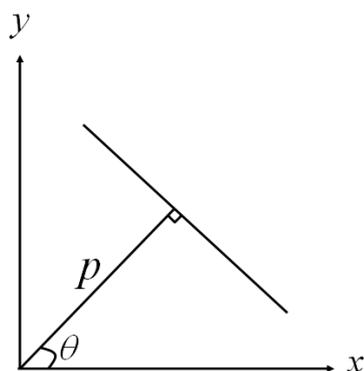


Figure 2.5: Polar Form of a Line

An alternative representation, known as *Polar Form*, of a line shown in Figure 2.5 is given by:

$$x \cos \theta + y \sin \theta = p \quad (2.19)$$

where  $\theta$  is the angle between  $x$ -axis and a perpendicular which is drawn from the origin to the line being detected. The same mapping is valid between  $x$ - $y$  space to  $p$ - $\theta$  space as between  $x$ - $y$  space and  $a$ - $b$  space. The advantages of this parameterization are that both  $p$  and  $\theta$  have finite values and  $\theta$  can be computed during edge detection operation from the gradient angle. Therefore, the computational complexity is reduced.

The line detection is very crucial in camera calibration for MAW. The determination of the corners of micro pattern must be very accurate so that calibration parameters can be estimated precisely. (See Section 5.1)

### Circle Detection

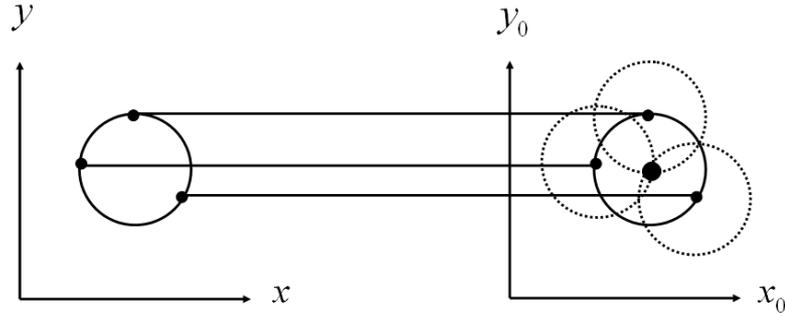


Figure 2.6: Mapping for Circular Hough Transform

The equation for a circle whose center is located at  $(x_0, y_0)$  and radius is  $r$  is given by:

$$(x - x_0)^2 + (y - y_0)^2 - r^2 = 0 \quad (2.20)$$

In this case we have 3-D parameter space composed by unknowns,  $x_0$ - $y_0$ - $r$ . The new parameterized space equations of a circle are given by:

$$x_0 = x - r \cos \theta \quad (2.21)$$

$$y_0 = y - r \sin \theta \quad (2.22)$$

where  $\theta$  is the gradient angle. The mapping for circular hough transform is shown in Figure 2.6.

As shown in Figure 2.3, the micro particles which are aimed to be manipulated in MAW applications have circular form. Because of this, circular hough transform is used to detect and locate them robustly and precisely. (See Section 5.2.1)

### 2.4.3 Elliptic Fourier Descriptors

Modeling of any closed curve structure to known models by finite set of measurements has always been one of the fundamental problems of the computer vision. *Fourier Descriptors* are useful in describing the shapes of 2-D closed contours. The basic idea is representing a closed curve by a periodic function of a continuous parameter  $t$ , or alternatively by a set of *Fourier coefficients* of this function. There are several ways to parameterize a closed curve to Fourier Descriptors. The most common way among them is based on *arc length* or *constant speed* parameterization. There are also several approaches for using arc length parameterization. One of them is *EFDs* [31–33]. These descriptors are good image features for modeling of 2-D objects with closed curve structures. The idea for EFDs is shown in Figure 2.7

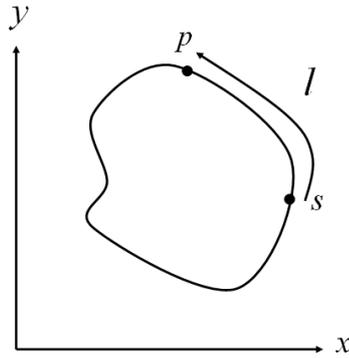


Figure 2.7: The Idea of EFDs

and can be briefly explained as follows: assuming a point  $p$  starts from  $s$  and moving along a 2-D closed contour with perimeter  $L$  assumed to have a constant velocity. Let  $l$  denote the arc length along the contour from  $s$  to  $p$ . Then the Cartesian coordinates of  $p$  can be considered as functions of  $l$  as  $x(l)$  and  $y(l)$ . By defining a parameter  $t$  as  $\frac{2\pi l}{L}$  (arc length parameterization), these two periodic functions can be expressed by means of Fourier harmonics in matrix form as:

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} a_0 \\ b_0 \end{pmatrix} + \sum_{k=1}^{\infty} \begin{pmatrix} a_k & b_k \\ c_k & d_k \end{pmatrix} \begin{pmatrix} \cos kt \\ \sin kt \end{pmatrix} \quad (2.23)$$

Since we have edge points which represent the boundary of a closed curve in a digital image, the discrete mathematical model of the EFDs is necessary. This

representation is given by 2.24. Note that number of fourier harmonics is limited in this case with a finite number.

$$\begin{pmatrix} x_N(k) \\ y_N(k) \end{pmatrix} = \begin{pmatrix} a_0 \\ b_0 \end{pmatrix} + \sum_{n=1}^N \begin{pmatrix} a_n & b_n \\ c_n & d_n \end{pmatrix} \begin{pmatrix} \cos \frac{2n\pi k}{N} \\ \sin \frac{2n\pi k}{N} \end{pmatrix} \quad (2.24)$$

where  $N$  is the maximum number of fourier harmonics,  $k$  represents the  $k^{th}$  number of the edge points,  $(a_0, c_0)$  are the mean values of the coordinates  $x_N(k)$  and  $y_N(k)$  which are the edge points of the boundary of a closed curve, and  $(a_n, b_n, c_n, d_n)$  are elliptic fourier coefficients which are used to model the closed curve as image features. The calculation of these coefficients is given as follows:

$$a_n = \frac{T}{2\pi^2 n^2} \sum_{k=1}^K \frac{\Delta x_k}{\Delta t_k} \left( \cos \frac{2\pi n t_k}{T} - \cos \frac{2\pi n t_{k-1}}{T} \right) \quad (2.25)$$

$$b_n = \frac{T}{2\pi^2 n^2} \sum_{k=1}^K \frac{\Delta x_k}{\Delta t_k} \left( \sin \frac{2\pi n t_k}{T} - \sin \frac{2\pi n t_{k-1}}{T} \right) \quad (2.26)$$

$$c_n = \frac{T}{2\pi^2 n^2} \sum_{k=1}^K \frac{\Delta y_k}{\Delta t_k} \left( \cos \frac{2\pi n t_k}{T} - \cos \frac{2\pi n t_{k-1}}{T} \right) \quad (2.27)$$

$$d_n = \frac{T}{2\pi^2 n^2} \sum_{k=1}^K \frac{\Delta y_k}{\Delta t_k} \left( \sin \frac{2\pi n t_k}{T} - \sin \frac{2\pi n t_{k-1}}{T} \right) \quad (2.28)$$

where  $K$  is again maximum number of edge points which represents the boundary of a closed curve,  $T$  is the total time for whole closed curve to be traveled,  $t_k$  is the necessary time to travel from start point to  $k^{th}$  consecutive edge point after start point,  $\Delta x_k$  and  $\Delta y_k$  are the distances between  $x$  and  $y$  coordinates of  $(k-1)^{th}$  and  $k^{th}$  edge points from start point and  $\Delta t_k$  is necessary time to travel from edge point  $(x_{k-1}, y_{k-1})$  to  $(x_k, y_k)$ . Note that all time calculations are done by assuming that the travel is done with constant velocity.

Additionally, EFDs can be used for object recognition applications. However, curve representation must be normalized with respect to a desired transformation domain so that the EFDs are invariant in the specified domain. Most of the researchers use the similarity transform, that is, the EFDs are normalized so that they are invariant under translation, rotation, scaling, and the location of the starting point in the image plane. The normalization was initially developed by Granlund in [34].

Therefore, EFDs are useful tools for modeling of complex geometric closed shaped curves and for object recognition of these complex structures. EFDs are

used to model micro particles as an alternative solution for MAW applications. It has some advantages and disadvantages in this sense. This will be mentioned when implementation of EFDs is explained in Section 5.2.2.

## 2.5 Motion Parameter Estimation

Motion parameter estimation in computer vision has been initiated by the early works of Ullman [35]. Roach and Aggarwal [36] tested the estimation problem with real images. Motion parameter estimation problem is to estimate the parameters of the specific motion models such as rigid body motion, affine motion or Riccati type motion models. In the following section, formulations for these dynamic motion models are given.

### 2.5.1 Motion Models

If a rigid body is moving with instantaneous translational velocity,  $T$ , and rotational velocity,  $\Omega$ , then the 3-D instantaneous velocity of points on the surface is given by

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix} = \Omega \times \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + T \quad (2.29)$$

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}}_{\triangleq [\Omega]_{\times}} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \underbrace{\begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}}_{\triangleq T} \quad (2.30)$$

where  $\Omega = (\omega_1, \omega_2, \omega_3)^T$  and  $T = (b_1, b_2, b_3)^T$ . Note that  $[\Omega]_{\times}$  is a skew-symmetric, i.e.  $[\Omega]_{\times} + [\Omega]_{\times}^T = 0$ , obtained from  $\Omega$ .

The motion of the projected points in the image plane,  $(\dot{x}, \dot{y})$ , can be obtained from 2.7, 2.8:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = f \begin{pmatrix} \frac{\dot{X} - x\dot{Z}}{Z} \\ \frac{\dot{Y} - y\dot{Z}}{Z} \end{pmatrix}, \quad (2.31)$$

and using 2.29 gives the required 2-D motion equations:

$$\dot{x} = f\omega_2x^2 - f\omega_1xy - f\omega_3y + f\omega_2 + f(b_1 - xb_3)/Z, \quad (2.32)$$

$$\dot{y} = -f\omega_1y^2 + f\omega_2xy + f\omega_3x - f\omega_1 + f(b_2 - yb_3)/Z. \quad (2.33)$$

## 2.5.2 Planar Structure

Note the existence of depth variable,  $Z$ , in the motion field is described in 2.32-2.33. Depth affects the image dynamics. If it is assumed to have a smooth surface structure, locally a linear depth variation can be used. This means that locally about a point  $(X, Y, Z)$ , the surface is approximately planar, i.e.  $Z \approx pX + qY + r$ , with the orientation defined by the surface normal,  $N = (p, q)^T$ .

Now the depth variable  $Z$  can be eliminated from motion fields by plugging

$$\begin{aligned} Z \approx pX + qY + r &\Rightarrow 1 \approx \frac{pX + qY + r}{Z} = px + qy + r/Z \\ &\Rightarrow 1/Z \approx \frac{1 - (px + qy)}{r} \end{aligned}$$

into 2.32- 2.33. This substitution yields so called Riccati dynamics:

$$\frac{d}{dt} \begin{pmatrix} x \\ y \end{pmatrix} \approx \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} + \begin{pmatrix} d_3 & d_4 \\ d_5 & d_6 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} d_7x^2 + d_8xy \\ d_8y^2 + d_7xy \end{pmatrix} \quad (2.34)$$

Image dynamics are governed by differential equations with quadratic right hand sides. Previous work on motion estimation suggests that, in practice, the second order terms play a minor role, to the extent that they often end up fitting the noise or deviations from the planar model [37, 38]. Therefore an affine image dynamics of the form

$$\frac{d}{dt} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \quad (2.35)$$

is usually considered.

Note also that under the scaled-orthographic projection described in 2.10, a 2-D rigid motion

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \end{pmatrix} = \begin{pmatrix} 0 & -\omega \\ \omega & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \quad (2.36)$$

will imply the following motion on the image plane:

$$\begin{aligned} \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} &= \frac{f}{\bar{Z}} \begin{pmatrix} \dot{X} \\ \dot{Y} \end{pmatrix} = \frac{f}{\bar{Z}} \left[ \begin{pmatrix} 0 & -\omega \\ \omega & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \right] \\ &= \begin{pmatrix} 0 & -\omega \\ \omega & 0 \end{pmatrix} \begin{pmatrix} fX/\bar{Z} \\ fY/\bar{Z} \end{pmatrix} + \frac{f}{\bar{Z}} \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} 0 & -\omega \\ \omega & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \bar{b}_1 \\ \bar{b}_2 \end{pmatrix} \end{aligned} \quad (2.37)$$

where, clearly the only difference is in the translational parameters, thus making the image dynamics also of rigid motion type.

### 2.5.3 Composite Image Feature Dynamics

Previous sections have shown that image dynamics can be rigid, affine or Riccati. Note also that motion and shape parameters enter into these dynamics linearly. In other words, these dynamics can be rewritten as

$$\dot{\xi} = f(\xi)\varphi \quad (2.38)$$

where  $\xi \in \mathbb{R}^2$  is the image feature vector, i.e.  $\xi = (x, y)^T$ ,  $f(\xi)$  is a linear (or nonlinear) mapping and  $\varphi$  is the vector of unknown parameters which can be either constant or time-varying. Suppose there are  $n$  parameters to be estimated, i.e.  $\varphi \in \mathbb{R}^n$  and  $f \in \mathbb{R}^{2 \times n}$ . From 2.38 it is clear that each image feature provides 2 equations. For the estimation of  $n$  parameters, it is required to have at least  $m \triangleq n/2$  independent image features. For example, rigid motion 2.36 in 2-D is defined by 3 parameters  $(\omega, b_1, b_2)$  and it is needed to have at least 1.5 image points. In the case of affine motion 2.35, 3 image features imply 6 equations which can be solved for 6 affine parameters,  $a_1, a_2, a_3, a_4, b_1, b_2$ . In the case of Riccati dynamics 2.34, it is needed to have at least 4 image features since they will imply 8 equations which can be used to solve 8 parameters,  $d_1, d_2, \dots, d_8$ .

Let us consider  $m$  image features concatenated in the following dynamical system:

$$\underbrace{\begin{pmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \vdots \\ \dot{\xi}_m \end{pmatrix}}_{\Xi} = \underbrace{\begin{pmatrix} f_1(\xi_1) \\ f_2(\xi_2) \\ \vdots \\ f_m(\xi_m) \end{pmatrix}}_F \varphi \Rightarrow \dot{\Xi} = F\varphi \quad (2.39)$$

where  $\xi_r = \begin{pmatrix} x_r & y_r \end{pmatrix}^T$  and  $r = 1, 2, \dots, k$ . Note that  $F$  is now a square matrix with  $n \times n$ . Since the image features are extracted from images, they are measurable and therefore an output (read-out) equation can be also introduced as

$$Y = \Xi \quad (2.40)$$

In Section 5.4, a variety of dynamical systems which can arise in MAW applications are considered and a novel identification procedure for the estimation of both constant and time varying parameters is developed. The basic procedure introduced for parameter estimation is to recast image feature dynamics linearly in

terms of unknown parameters and construct a sliding mode observer [39,40] to produce asymptotically correct estimates of the observed image features, and then use “*equivalent control*” to explicitly compute parameters.

## 2.6 Autofocusing

Autofocusing is one of the ways to measure vertical position information and to determine focal planes existing in the workspace of a microscope. Autofocusing is also a fundamental technology for automated biological - biomedical applications such as high-throughput screening for the pharmaceutical industry and micromanipulation. In the case of micromanipulation, because of non-planar micro objects and manipulation tool(s) such as probes or microgrippers, there might be several focal planes.

Autofocusing techniques are divided into 2 groups as *Active (Sensor Based)* and *Passive (Image Based)* autofocusing. Active autofocusing uses external measurement tools such as lasers or ultrasound devices to measure the distance between microscope objective and current view of the microscope. On the other hand, passive autofocusing relies on image processing techniques to determine if the micro objects or manipulation tool(s) are focused or not [41]. Autofocusing for automated microscopy has been broadly studied for several decades [42–45]. Autofocusing for micromanipulation has been studied in several works [5, 13, 46–49].

Groen et al. [42], Firestone et al. [43] and Sun et al. [44] had tried to find optimal passive autofocusing technique by comparing each technique based on different aspects like unimodality, accuracy, reproducibility, range, general applicability, insensitivity to other parameters, implementation, number of false maxima, width and noise level. They compared several autofocusing functions which are generally cost functions. Trying to maximize or minimize the cost functions provides the optimal focal plane for an digital image. These functions can be classified into 4 groups as:

### 1. Derivative-Based Algorithms

- (a) Thresholded Absolute Gradient (Santos et al., 1997)
- (b) Squared Gradient (Santos et al., 1997)
- (c) Brenner Gradient (Brenner et al., 1971)

- (d) Tenenbaum Gradient (Tenengrad) (Yeo et al., 1993; Krotov, 1987)
- (e) Sum of Modified Laplace (Nayar and Nakagawa, 1994)
- (f) Energy Laplace (Subbarao et al., 1993)
- (g) Wavelet Algorithms  $W_1, W_2, W_3$  (Yang and Nelson, 2003a,b)

## 2. Statistical Algorithms

- (a) Variance (Groen et al., 1985; Yeo et al., 1993)
- (b) Normalized Variance (Groen et al., 1985; Yeo et al., 1993)
- (c) Auto Correlation (Vollath, 1987, 1988)
- (d) Standard Deviation-Based Correlation (Vollath, 1987, 1988)

## 3. Histogram-Based Algorithms

- (a) Range Algorithm (Firestone et al., 1991)
- (b) Entropy Algorithm (Firestone et al., 1991)

## 4. Intuitive Algorithms

- (a) Thresholded Content (Groen et al., 1985; Mendelsohn and Mayall, 1972)
- (b) Thresholded Pixel Count (Groen et al., 1985)
- (c) Image Power (Santos et al., 1997)

Many of these algorithms have been used by micromanipulation society according to applications and performance of autofocusing functions in those applications. Inoue et al. was able to get good results for autofocusing by using Chromatic Aberration using color information in [5]. Allegro et al. in [13] uses Brenner Sharpness focus function to measure focal planes. Suzuki et al. reached their best results with combination of Tenenbaum Gradient algorithm and Chromatic Aberration algorithm in [46]. Haliyo et al. in [47] used Tenenbaum Gradient algorithm. Sulzman et al. in [48] measured focal planes by a modification of Brenner Sharpness function with combination of lenses used, the tracked structure and optimal light estimation. Parvin et al. preferred to use Derivative-Based functions of wavelet algorithms to find the best focused images in [49].

According to Sun et al. in [44] *Normalized Variance* algorithm provides the best overall performance compared to 17 other autofocusing algorithms. Normalized variance algorithm is also very robust to image noise and low-pass filtering as a preprocessing operation compared to Derivative-Based algorithms. For this reason, Normalized variance algorithm is implemented for MAW applications to measure the focal planes of manipulation tool(s) and micro objects on the manipulation surface; and to calculate the vertical distance in-between.

Variance algorithm computes variations in gray level among image pixels. By normalizing the final output with mean intensity “ $\mu$ ”, normalized variance algorithm compensates for the differences in average image intensity among different images. Formulation of the normalized variance cost function is given by:

$$F_{nor\_var} = \frac{1}{H \cdot W \cdot \mu} \underbrace{\sum}_{Height} \underbrace{\sum}_{Width} (i(x, y) - \mu)^2 \quad (2.41)$$

where  $H$  and  $W$  are the pixel size of image height and image width respectively;  $i(x, y)$  represents the intensity value of image pixel located at  $(x, y)$ . Maximum value of the cost function indicates the focal plane where micro particles on the manipulation surface or manipulation tool(s) are well focused.

Implementation of autofocusing technique of our choice is demonstrated in Section 5.5.

## Chapter 3

### Vision Based Control

This chapter is dedicated to give a brief explanation of vision based control, its categorization, and detailed descriptions of vision based control techniques implemented on MAW.

The control of a robotic manipulator with visual feedback was first named as *visual servoing* by Hill and Park in [50]. If vision system provides feedback about the state of the environment in the control loop of the manipulator, it is called visual servoing. Visual servoing is the fusion of results from many elemental areas including computer vision, digital image processing, forward & inverse kinematics, dynamics, control theory and real-time computing operations. Vision is a non-contact sensing modality, in other words a passive sensor, which provides dense information about the environment and enables recognition of objects. The accuracy of a robotic operation traditionally depends on the accuracy of the robotic manipulator and its internal controller. However, using visual feedback in the control loop increases accuracy, robustness and flexibility of the overall system [51–54]. The basic categorization of vision based control is defined by Sanderson and Weiss in [55]. The classification is as follows:

1. Open Loop (Static Look & Move) Control
  - Image Based Control
  - Position Based Control
2. Closed Loop Control
  - Image Based Dynamic Look & Move Control
  - Image Based Direct Visual Servoing

- Position Based Dynamic Look & Move Control
- Position Based Direct Visual Servoing
- Hybrid Based Visual Servoing

Only Static Look & Move and Position Based Dynamic Look & Move control structures are implemented for MAW from the above list. Therefore, their detailed expositions are denoted in the following sections of this chapter.

The difference between direct visual servoing and dynamic look & move structure is that visual controller directly computes the input to the robotic manipulator joints and robotic manipulator internal control is eliminated in the direct visual servoing while the vision system provides input to robotic manipulator internal controller which then uses joint feedback to internally stabilize the robotic manipulator in dynamic look & move structure. Since, motion control of robotic manipulator is done internally as initial step for MAW, that is why only dynamic look & move control structure is implemented for automation tasks.

### 3.1 Static Look & Move Control Structure

The control is sequential and asynchronous for static look & move structure. This control structure can be described by 3 steps as shown in Figure 3.1. First step is the determination of actual and target positions by the operator for the robotic manipulator to move manually. In the second step, the distance between these points are obtained and given to the robotic manipulator as a reference by visual controller. During the third step, the robotic manipulator is moved to the target point with its

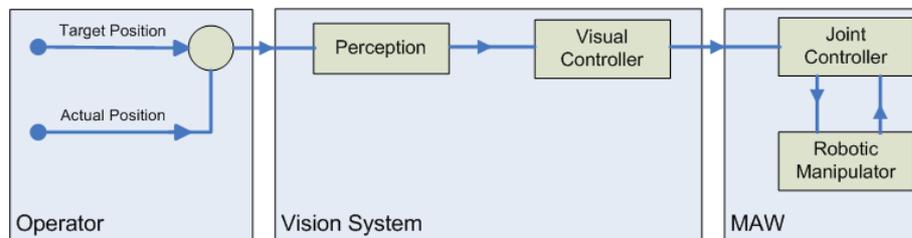


Figure 3.1: Static Look & Move Control Structure

internal motion control. During this motion, no visual information is observed and feed back to the robotic manipulator. When the robotic manipulator's motion is finished, the operator checks the error between actual position and target position of the robotic manipulator. If there still exists error between these positions, static look & move based vision control starts again. This sequential control continues until the error falls into an acceptable accuracy range.

There are two types of static look & move control structure: *Image Based* and *Position Based* Static Look & Move. IBSLM uses pixel coordinates of actual and target positions to determine the error between these positions. By generating a simple proportional control over this error, it gives output of the visual controller as reference to the robotic manipulator as indicated in Equation 3.1.

$$X_{er} = X_{tar} - X_{ac} \Rightarrow \begin{pmatrix} x_{er} \\ y_{er} \end{pmatrix} = \begin{pmatrix} x_{tar} \\ y_{tar} \end{pmatrix} - \begin{pmatrix} x_{ac} \\ y_{ac} \end{pmatrix} \quad (3.1)$$

$$Ref = K|X_{er}|$$

where  $X_{tar}$ ,  $X_{ac}$  and  $X_{er}$  are pixel coordinate vectors of target position, actual position and error respectively.  $Ref$  represents the reference to the robotic manipulator generated by simple proportional controller with gain  $K$ .

On the other hand, PBSLM structure calculates the error between target and actual position by using their 3-D world coordinate values. These values are estimated using inverse of camera calibration matrix discussed in Section 2.3. Error is input to visual controller which is again a simple proportional control by gain  $K$ . The output of the visual controller is used as reference to the robotic manipulator.

$$\begin{aligned} \bar{X}_{er} &= \bar{X}_{tar} - \bar{X}_{ac} \\ Ref &= K|\bar{X}_{er}| \end{aligned} \quad (3.2)$$

and

$$\bar{X}_{er,tar,ac} = P^{-1}X_{er,tar,ac} \quad (3.3)$$

where  $\bar{X}_{tar}$ ,  $\bar{X}_{ac}$  and  $\bar{X}_{er}$  are 3-D world coordinate vectors of target position, actual position and error respectively.  $Ref$ , again, represents the reference to the robotic manipulator, and  $P^{-1}$  is the inverse of camera calibration matrix shown in Section 2.3.

Static look & move control structure is implemented for teleoperated microassembly automation tasks which are presented in Section 5.3.1.

### 3.2 Dynamic Look & Move Control Structure

PBDLM structure differs from the static structure in the fact that in this structure the reference and commanded inputs can be changed synchronously independent of the robotic manipulator's current position or velocity. It also differs from direct visual servoing because vision system provides input to internal controller of the robotic manipulator which then uses joint feedback control to internally stabilize the robotic manipulator in dynamic structure while the robotic manipulator's internal joint controller is eliminated in direct visual servoing. The overall system structure of PBDLM is shown in Figure 3.2.

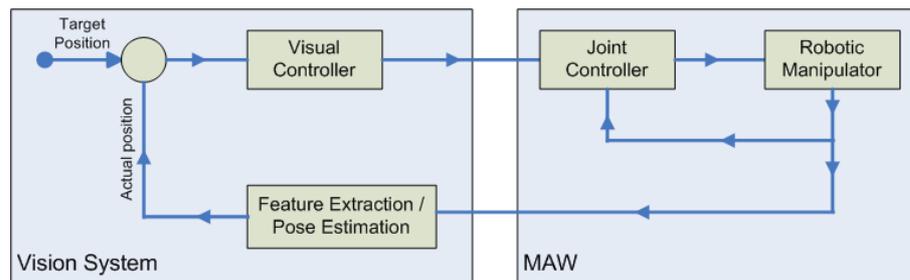


Figure 3.2: Dynamic Look & Move Control Structure

Error to the visual controller can be defined according to the corresponding a priori information that we have about automated micromanipulation or microassembly task(s). Error does not need to be scalar as in the case of static control structure. It can be a vector composed of different errors which are defined depending on different states of the dynamical model of the task. This is explained in detail in the next section.

Dynamic look & move control structure is implemented for semi-automated microassembly automation tasks and explained in Section 5.3.2.

### 3.3 Visual Task Functions

In general, the task in vision based control is to control a robotic manipulator to manipulate its environment using visual feedback as opposed to just observing the environment. A visual task is also referred as a visual task function or a control error function as implied by Chaumette et al. in [56].

For a given micromanipulation or microassembly task, a set of visual features,  $s$ , have to be chosen for achieving the task. These visual features can be tracked over the entire course of the task. The differences between their references that are determined before the task is initiated, and these visual features are formed as error functions which are inputs to the visual controller.

Representing some desired set of features by  $s^*$  and the set of current features with  $s$ , the objective of vision based control is to regulate the task function to zero. When the task is completed, the following equality holds:

$$e(s^* - s) = 0 \tag{3.4}$$

In Section 5.3.2, in order to accomplish specific semi-automated micromanipulation or microassembly task, several visual features are selected to form visual task function of that specific task. These visual features are determined depending on a priori knowledge that we have about the aim of the task.

## Chapter 4

### Experimental Setup of MAW

In this chapter, experimental software and hardware setup of MAW is expounded. As depicted in Figure 4.1, experimental setup for MAW consists of two major parts as software and hardware setups. Software setup is divided into two as motion control software and vision software. On the other hand, hardware setup has five main components which are vision system, control unit, motion unit, manipulation unit and teleoperation unit.

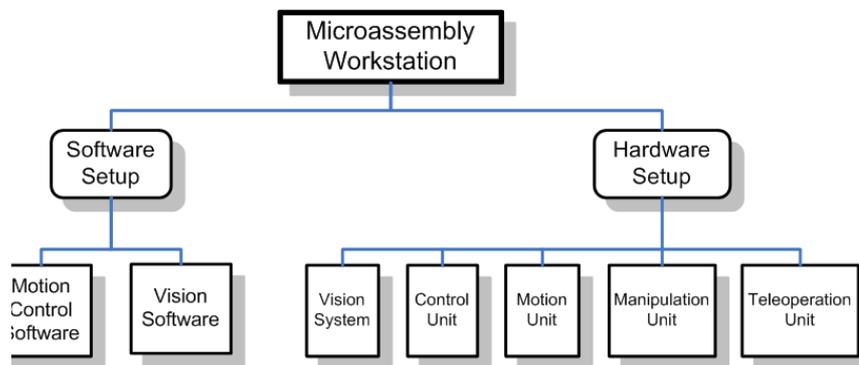


Figure 4.1: Experimental Setup Chart of MAW

#### 4.1 Software Setup of MAW

MAW software structure is composed of two main parts. Motion control software is where all proposed control algorithms for motion, manipulation and teleoperation units are implemented. Vision software is responsible for the implementation of

digital image processing, computer vision and vision based control algorithms and task functions to enable teleoperated or semi-automated micromanipulation and microassembly tasks to be performed. It also comprises the user interface of MAW. Each software is programmed in its own host PC and communicates with each other over ethernet communication protocol by means of a bridge software. Figure 4.2 expresses the basic flow chart of software structure of MAW.

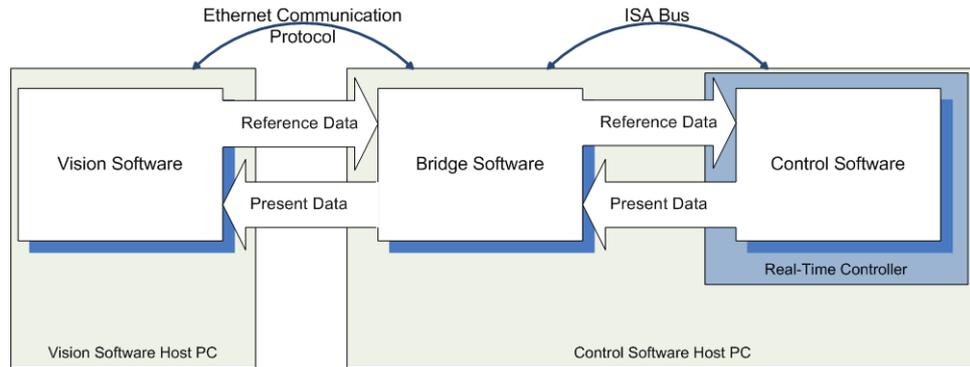


Figure 4.2: Software Structure of MAW

#### 4.1.1 Motion Control Software

Motion control software is coded to implement several control algorithms and techniques, particularly SMC, for motion, manipulation and teleoperation units to have highly accurate, very precise and repeatable motion in micro/nanometer scale. Motion control software is programmed using C programming language and downloaded to embedded real-time controller board. Controller board allows the control of 8 DOF at the same time with desired accuracies.

Once the motion control software is downloaded to the embedded real-time controller board, there is no access to change any part of the code or algorithms. However, there is a possibility to use read/write functions for specific variables defined in the motion control software. A bridge software is written in order to enable read/write functions for those specific variables from motion control software host PC. Position references can be written and actual positions of each DOF can be read

by aid of this software for each DOF over ISA bus. It also performs communication between motion control software host PC and vision software host PC over ethernet communication protocol. In other words, vision software is also able to obtain and control position variables of motion and manipulation units by means of the bridge software.

### **4.1.2 Vision Software**

Vision software is written to implement digital image processing, computer vision and vision based control algorithms; and perform teleoperated and semi-automated micromanipulation and microassembly tasks for MAW applications. It has also easy-to-use user interface on which the operator can see different camera images with different magnifications from the workspace view of MAW; can take pictures and record videos from each camera image; can either activate or deactivate serial & ethernet communications, auto focus device or micromanipulation & microassembly tasks. The operator can also have access to actual position values of motion and manipulation units on the screen and can give reference to them in real-time. This is possible with the bridge software mentioned in the previous section. In Figure 4.3 and Figure 4.4, the user interface and camera images of the vision software are shown respectively. Vision software is coded in Visual C++ programming environment.

The user interface of vision software is designed using MFC Library of Visual C++ programming environment. The user interface has many functionalities although it has visually a rough design. Gray part includes input section for the microscope magnification. When the user interface is opened for the first time, it asks the operator to enter the current microscope magnification at first. When the microscope magnification is entered, specific parameters related to camera calibration matrices, camera features such as brightness, shutter, frame rate; and camera views such as image sizes, positions of each camera image on the monitor, are automatically loaded to vision software. Gray part also contains the initialization buttons for each camera and screens for their frame rates. When they are initialized, images corresponding to its own camera appears on the specific position of the monitor and their corresponding frame rates appear on their own screens.

On red part, ethernet communication of vision software host PC with motion

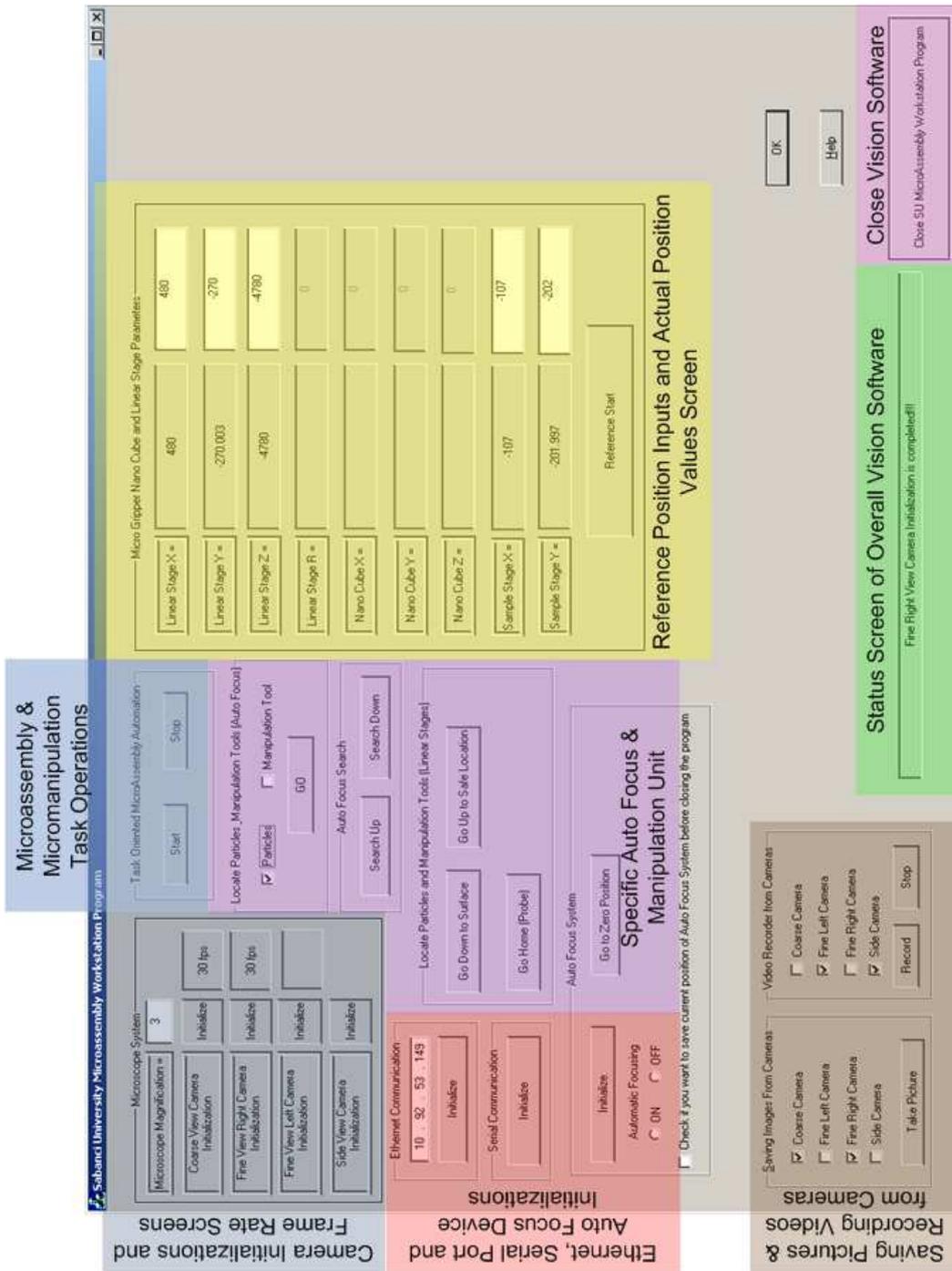


Figure 4.3: User Interface of Vision Software

control software host PC can be initialized by writing IP address of the motion control software host PC. Serial port and auto focus initializations are also done in this part. Serial port communication is necessary to be able to communicate with the auto focus device and to use its functions to operate it. Auto focus device has its own motion controller as hardware, and there are several functions already defined and embedded on this controller. Vision software uses those functions by sending appropriate data to motion controller of auto focus device over serial port. Autofocusing algorithm works while initialization of auto focus device. Processes which are done during initialization of auto focus device are depicted in Section 5.5 in detail.

Brown part is assigned to take pictures of current scene and record videos of implemented micromanipulation and microassembly tasks from each camera view. The only requirement to record videos from cameras is that when each camera is initialized, it is asked to select an appropriate video codec for recording. According to algorithm of the video codec, size and quality of video file can change dramatically.

The operator can start or stop micromanipulation and microassembly task(s) by using buttons in blue part. Task can be stopped before it is completed if an unexpected situation occurs. Otherwise it stops automatically, when it accomplishes its goal. Either teleoperated or semi-automated tasks can be initialized from this part. Details of how these tasks are executed are explained in Section 5.3.1 and 5.3.2.

Lilac part has several particular functions related to auto focus device and manipulation unit. When auto focus device is initialized, the  $z$ -axis positions of both manipulation tool and micro particles are determined by autofocusing algorithm. (See Section 5.5 for details) Functions in this part control either the microscope or manipulation tool. The operator can give motion to the microscope along  $z$ -axis to focus between manipulation tool and micro particles by the aid of auto focus device. At the same time, manipulation tool can be moved to the level of micro particles along  $z$ -axis. This is necessary before micromanipulation and microassembly tasks start. Tasks can be started once we are sure that manipulation tool and micro particles are on the same level along  $z$ -axis. Then the micro particles can be manipulated.

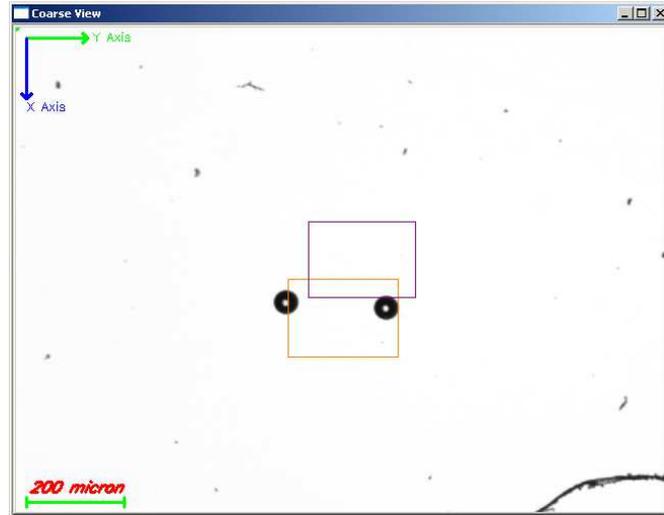
Yellow part is dedicated to control the motion units. The operator can enter

reference position inputs for motion units and send them to real-time controller board with the help of bridge software over ethernet. At the same time, actual position information for each DOF of motion units is displayed on the appropriate screens. So that the operator can see how accurate and precise the motion units are by comparing reference and actual position values.

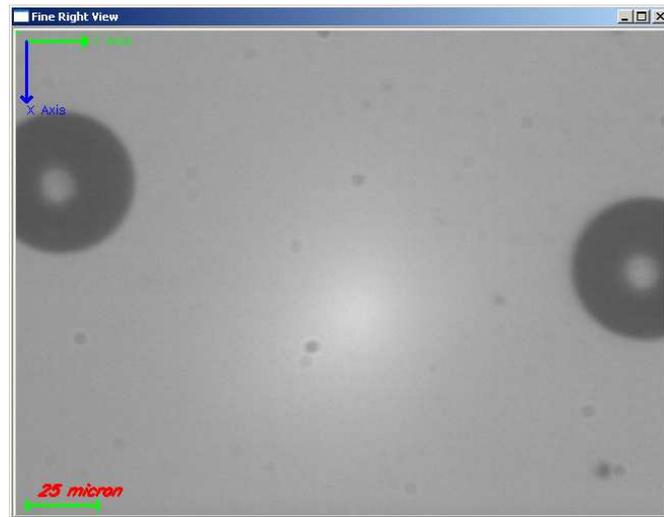
Green part is composed of status screen of overall vision software. When a new operation is started or stopped or any changes occur during any operation, a message about the status of vision software is displayed on the status screen. For some operations, window messages are displayed to stress that something significant just happened. Additionally, the button on pink part closes vision software completely. By that all communications of vision software host PC are cut.

As a part of the user interface, there are camera views which display real-time (30 fps) images captured from each camera when they are initialized as shown in Figure 4.4. Each camera view has default appendages on the images like the coordinate system of MAW and visually the resolution (micrometer per pixel) of the current image. The operator can move motion or manipulation units towards desired direction with desired amount using these visual appendages. On the top-left corners, 2-D (x-y) coordinate system axes of MAW are located.  $z$ -axis is not shown but it is along from image to outwards. On the bottom-left corners, the micrometer which shows the scale of  $200\mu m$  for camera view which has lower magnification, and  $25\mu m$  for camera view which has higher magnification. The micrometers are automatically adjusted if the microscope magnification is changed during vision software is operational. There are also optional region of interest rectangles for camera view with lower magnification. These rectangles express which portion of that camera view is magnified and viewed by cameras with higher magnification.

The implementations of digital image processing, computer vision and vision based control algorithms are done as a part of micromanipulation and microassembly tasks using necessary OpenCV Library functions integrated to Visual C++ programming environment. Vision based controls are coded to fulfill requirements of teleoperated and semi-automated tasks. These algorithms are performed on MAW with several task scenarios and the results of the implementations are demonstrated in Section 5.3.



(a)



(b)

Figure 4.4: Camera View Windows: (a) Coarse View Camera Window (b) Fine View Camera Window

## 4.2 Hardware Setup of MAW

MAW hardware setup is divided into 5 subgroups which are 1) vision system having a stereoscopic optical microscope, CCD cameras, illumination system, auto focus device and vision software host PC; 2) control unit consisting of embedded real-time controller board and its input/output control panel; 3) motion unit having linear

stages and piezo actuated nano positioner; 4) manipulation unit having microgrippers and probes depending on micromanipulation or microassembly applications; and 5) teleoperation unit consisting of force feedback controlled lever. In Figure 4.5, the hardware setup structure for MAW is shown.

### 4.2.1 Vision System

Vision system is designed to attain certain requirements which are essential in order to perform micromanipulation or microassembly in microscale world. They come into the picture of vision system because a stereoscopic optical microscope is the fundamental equipment. These requirements can be summarized:

- **Magnification** refers to the ratio of the distance between two points in the image, lying in a plane perpendicular to the optical axis of the microscope, to the corresponding two points in the specimen. It is selected depending on micro particles to be manipulated or assembled because magnification must be large enough to have clear and detail sight of micro particles to recognize their properties easily.
- **Resolution** is a measure of how fine a detail can be detected. In terms of microscale world, it means that how much small a micro particle can be recognized, a motion or displacement can be measured by the vision system. Basically, it is the value of micrometer per pixel in the image for meaningful, detectable feature.
- **Depth of Field** is the distance between the closest and farthest objects in focus within a scene as viewed by a lens at a particular focus and with given settings. The depth of field varies with the focal length of the lens and numerical aperture and the wavelength of light. It is very important in the sense of autofocusing. If it is narrow enough, the sharpest image can be obtained to catch that field. On the other hand, if it is too narrow, it is very difficult to focus that small range.
- **Field of View** is the maximum area visible through a lens or the eyepiece of an optical instrument; it is usually represented by an angle as depicted in

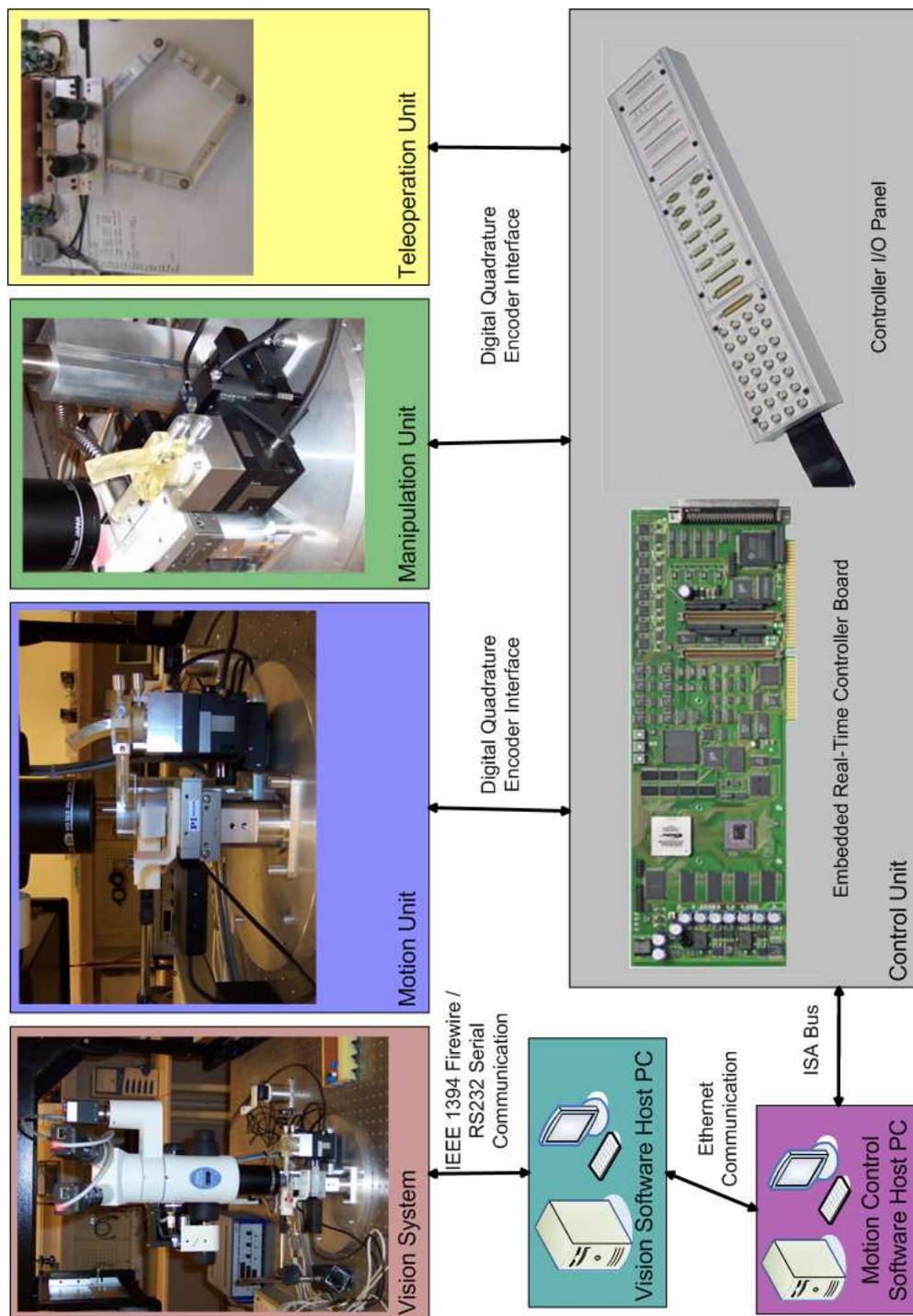


Figure 4.5: Hardware Structure of MAW

Section 2.1. Higher magnification yields to a narrower field of view. However, it is necessary to have a detailed image and wider field of view at the same time for micromanipulation and microassembly tasks. It is achieved by having different camera views with different magnifications. Required resolution can be obtained by camera view with higher magnification (it will be called fine camera view), while required field of view can be retrieved by camera view with lower magnification (it will be called coarse camera view).

- **Working Distance** is the clear distance between the specimen being viewed and the first optical element of the objective lens. It must be large enough for motion and manipulation units to move in the workspace of the microscope freely without crashing to anything.
- **Illumination** is simply the application and distribution of light to a micro particle. It is very important to have uniformly distributed light on the workspace of MAW since clear images can be captured and micro particles' and manipulation tools' boundaries on the images can be extracted effectively with uniform lighting.

In addition to these optical requirements, there are two other necessities for design of vision system to be taken into consideration. First one is to have high speed high resolution cameras attached to the stereoscopic optical microscope, which provide real-time digital images of MAW workspace. Secondly, an auto focus device should be included in the vision system design to be able to execute autofocusing operation or move the microscope along  $z$ -axis manually. Auto focus device is also used to control the magnification and focus of the microscope automatically.

In the light of design necessities mentioned above, vision system hardware setup for MAW has the following hardware structure as shown in Figure 4.6. In the next sections, each vision system hardware equipment and their technical specifications will be explained in detail.

### **Stereoscopic Optical Microscope**

Specifications of the selected stereoscopic optical microscope are denoted in Table 4.1. It provides very high magnification with wide working distance which

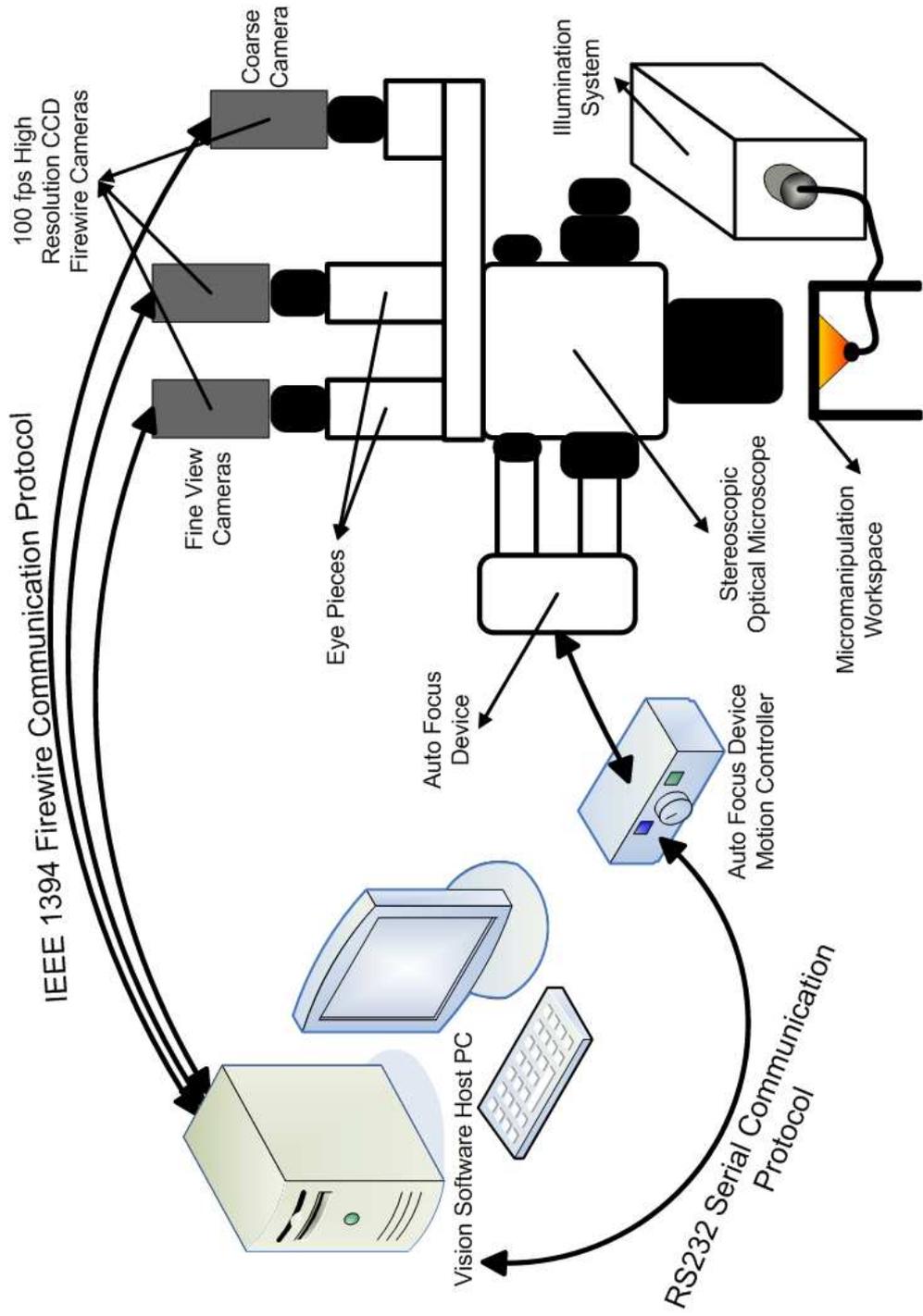


Figure 4.6: Vision System Hardware Structure

Table 4.1: Stereoscopic Optical Microscope Specifications

<b><i>Optical System</i></b>	Parallel-Optics Zoom System
<b><i>Zoom Range</i></b>	0.75× - 11.25×
<b><i>Eyepieces</i></b>	10×, 20×, 30×
<b><i>Objective</i></b>	1.5×, 2×
<b><i>Total Magnification</i></b>	1.5× - 675× (Depending on eyepiece and objective used)
<b><i>Working Distance</i></b>	30 mm - 22 mm
<b><i>Illumination System</i></b>	Halogen Illuminator Light Source & Fiber Pair

enables for manipulation tools to move relatively free in the workspace of the microscope. Field of view of both camera with low magnification and cameras with

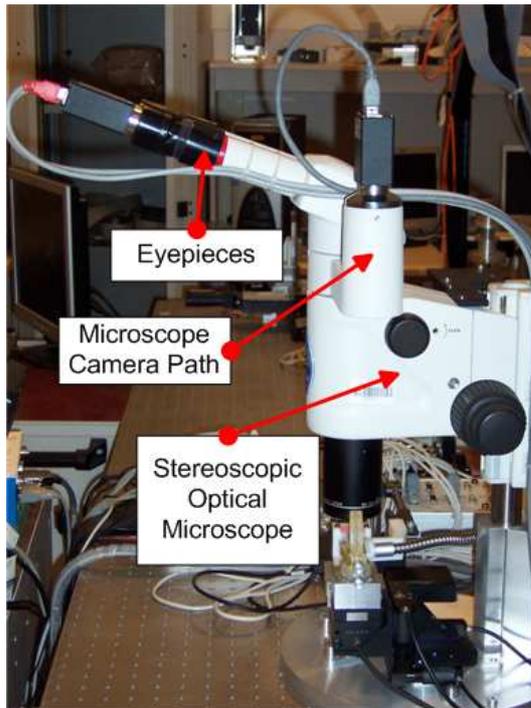


Figure 4.7: Stereoscopic Optical Microscope

high magnification are denoted in Table 4.2. When we consider the widest and the narrowest areas for camera views, MAW can perform micromanipulation and

microassembly tasks of micro particles whose size can change between  $1mm$  and  $1\mu m$ .

Table 4.2:  $\mu m$  Field of View of Coarse and Fine Camera Views for Each Magnification

<b>Magnification</b>	<b>Coarse Camera View</b> ( $640 \times 480$ Image)	<b>Fine Camera View</b> ( $640 \times 480$ Image)
0.75 $\times$	$5204.16\mu m \times 3903.12\mu m$	$828.16\mu m \times 621.12\mu m$
1 $\times$	$3939.84\mu m \times 2954.88\mu m$	$610.56\mu m \times 457.92\mu m$
2 $\times$	$1987.84\mu m \times 1490.88\mu m$	$314.88\mu m \times 236.16\mu m$
3 $\times$	$1354.496\mu m \times 1015.872\mu m$	$209.92\mu m \times 157.44\mu m$
4 $\times$	$983.68\mu m \times 737.76\mu m$	$157.44\mu m \times 118.08\mu m$
5 $\times$	$792.64\mu m \times 594.48\mu m$	$140.16\mu m \times 105.12\mu m$
6 $\times$	$658.24\mu m \times 493.68\mu m$	$117.12\mu m \times 87.84\mu m$
7 $\times$	$564.672\mu m \times 423.504\mu m$	$101.12\mu m \times 75.84\mu m$
8 $\times$	$492.864\mu m \times 369.648\mu m$	$88.96\mu m \times 66.72\mu m$
9 $\times$	$438.336\mu m \times 328.752\mu m$	$81.6\mu m \times 61.2\mu m$
10 $\times$	$395.84\mu m \times 296.88\mu m$	$73.28\mu m \times 54.96\mu m$
11.25 $\times$	$346.56\mu m \times 259.92\mu m$	$64.32\mu m \times 48.24\mu m$

The reason why a stereoscopic optical microscope, which has two eyepieces attached to binocular tube, is selected instead of monocular microscope with only one eyepiece is to benefit from stereo vision provided by digital cameras attached to the eyepieces.

### **Digital Cameras**

There are 3 digital high speed ( $100$  fps) high resolution cameras attached to vision system for MAW. Two of them are attached to the eyepieces of the microscope with couplings. These cameras are named as *Fine Cameras* and have higher magnification. The last camera is attached to the microscope from its own camera path. It has approximately  $6\times$  lower magnification than *Fine Cameras* and is called as *Coarse Camera*.

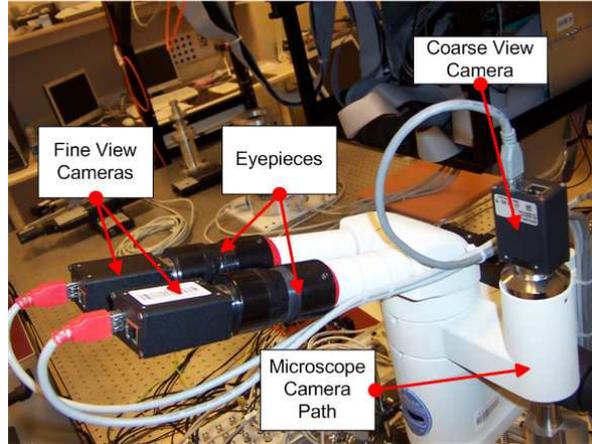


Figure 4.8: Coarse and Fine View Digital Cameras

According to the mathematical calculations of magnifications for each camera, the difference of magnifications between coarse camera and fine cameras should be the same as the amount of eyepieces magnification. However, in reality, it is approximately  $6\times$  when  $30\times$  eyepieces are attached. There are two possible reasons for this. First one is that the relative position of the fine cameras according to focal length of the eyepieces is not the correspond to the position where the refraction of the light can create visible  $30\times$  magnification. Second reason is that since each camera has its own digital magnification depending on how much portion of CCD sensors are used, coarse camera can have larger digital magnification than fine cameras on the monitor.

Depending on the necessity, one extra camera with a macro lens can be added to the vision system to visualize the overall workspace of MAW from outside. It is the *Side Camera*.

The resolution for fine and coarse cameras is indicated in Table 5.2. The image sizes for each camera view are  $640 \times 480$  pixel. Although cameras can be used operate at 100 fps, because of bandwidth of the bus and image capturing card limitations, all cameras are adjusted to worked at 30 fps which is in fact adequate to implement real-time digital image processing, computer vision and vision based control algorithms.

## Illumination System

As stated in above table, fiber pair with halogen light source is used as illumination system for MAW. At the very beginning of design process of vision system, the most important problem was the illumination. The micro particles were illuminated from upside to downward by using the fiber pair. This technique is called as *Reflected (Episcopic) Light Illumination*. Images captured under this illumination technique is depicted in Figure 4.17. The micro particles are seen as transparent in this case

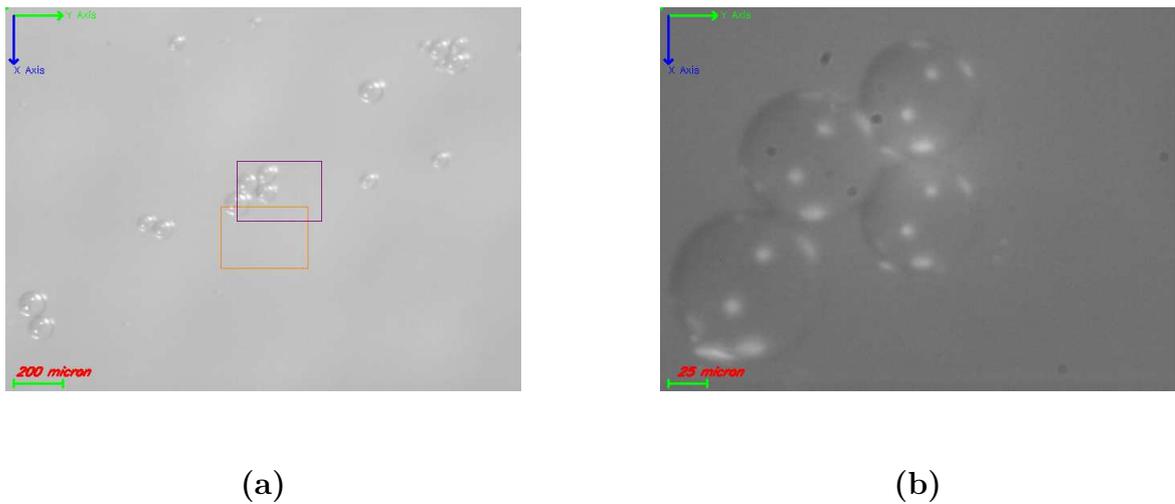


Figure 4.9: Transparent Micro particles Obtained Using Reflected Light Illumination: (a) Coarse View Camera Image (b) Fine View Camera Image

in which segmentation of the boundaries of the micro particles is very difficult and very complex digital image processing techniques, which can cause loss of frame rate, should be implemented. After that, some other illumination techniques are tested with different types of microscopes. Finally, *Backlighting Illumination* technique which is in fact feeding the light from downside to upward is selected because of its good performance. Since the manipulation surface, on which the micro particles are deposited, is a cover glass, feeding the light from downside does not cause loss of the amount of illumination, giving better results as shown in Figure 4.10. It is very simple for this kind of digital images to extract the boundaries of the micro particles because the intensity values of background and micro particles are quite different.

In reality, illumination must be changed for different applications because for different applications from different disciplines such as material science, bio-technology,

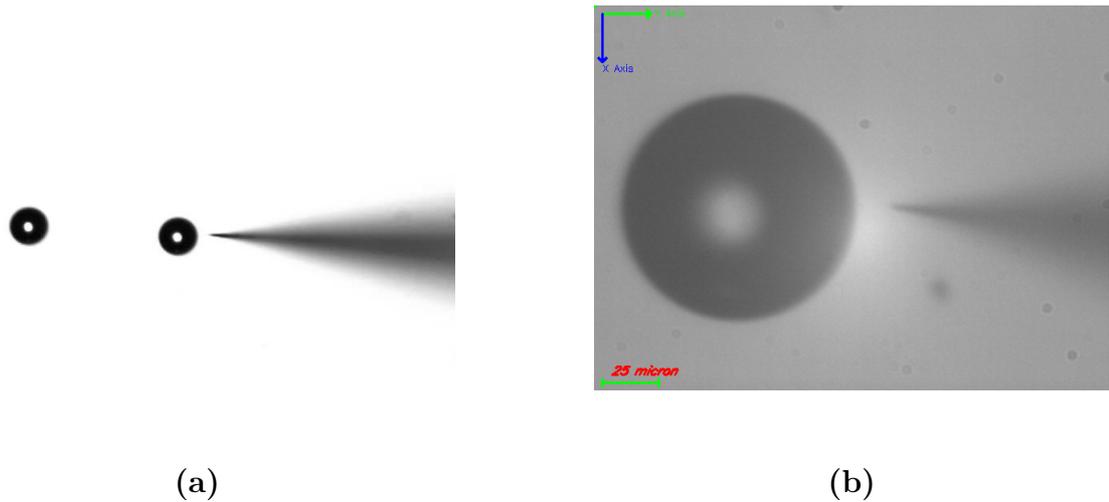


Figure 4.10: Micro particles Obtained Using Backlighting Illumination: (a) Coarse View Camera Image (b) Fine View Camera Image

microsystems, special illumination techniques have been developed. In case of manipulation and assembly of micro particles, backlighting gave the best results.

### Auto Focus Device

Auto focus device is specifically designed to provide a high resolution ( $0.8\mu m$ ), and high repeatability, by means of controlling the focus/ $Z$  position of the stereoscopic optical microscope. It is attached to knobs of the magnification and focus/ $Z$  of the microscope. It has its own controller as a separate unit.

It has following features:

1. Precise control of the microscope's focus is obtained through the use of a closed-loop DC servomotor employing high resolution encoders for positioning feedback.
2. By using closed-loop control of the focus position, there is no chance that the focus point can be lost as in the case with open-loop stepper motors.
3. Communication with vision software host PC through *RS232* serial port communication protocol

The control of DC servomotors is done in controller unit of auto focus device. As it was mentioned in Section 4.1.2, functions to control auto focus device have

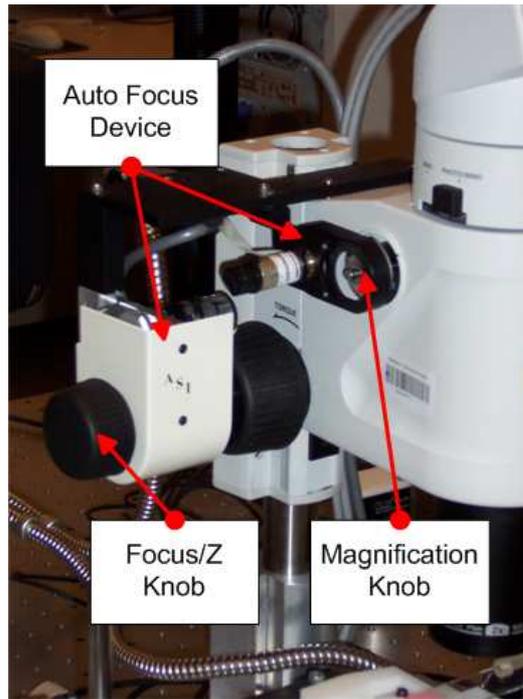


Figure 4.11: Auto Focus Device

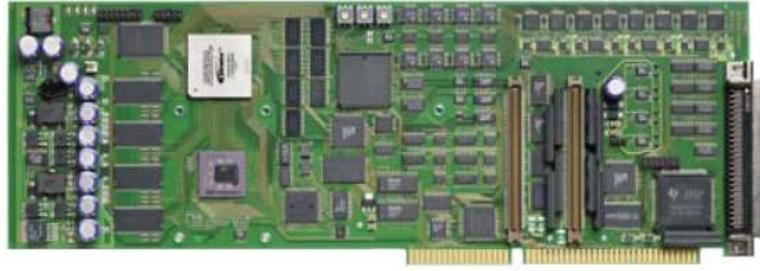
already been defined in the controller. The control of auto focus device from vision software host PC is done by sending appropriate data that contains information about the function to be executed, to auto focus device over serial port.

Auto focus device has also foot switch attachments that control the motors of auto focus device manually if it is necessary.

#### 4.2.2 Control Unit

Control unit consists of a real-time controller board embedded into motion control software host PC and an Input/Output control panel. Real-time motion control of each DOF is done in the control unit simultaneously. Real-time controller board has the specifications of:

1. Single-board system with comprehensive I/O
2. PWM-synchronous or externally triggered I/O strobe
3. CAN interface and serial interfaces



(a)



(b)

Figure 4.12: Hardware Equipment of Control Unit: (a) Real-Time Controller Board  
(b) I/O Control Panel

4. Interfaces for connecting incremental encoders
5. High I/O speed and accuracy

### 4.2.3 Motion Unit

Motion unit consists of two different parts as motion unit of manipulation tool(s) and motion unit of manipulation surface. Motion unit of manipulation tool has two different types of motion classified as coarse positioning and fine positioning. This unit includes linear stages for coarse positioning, and piezo actuated nano positioner for fine positioning. 3 linear stages give coarse motion in  $XYZ$  directions, and nano positioner gives highly precise fine motion in those directions for manip-

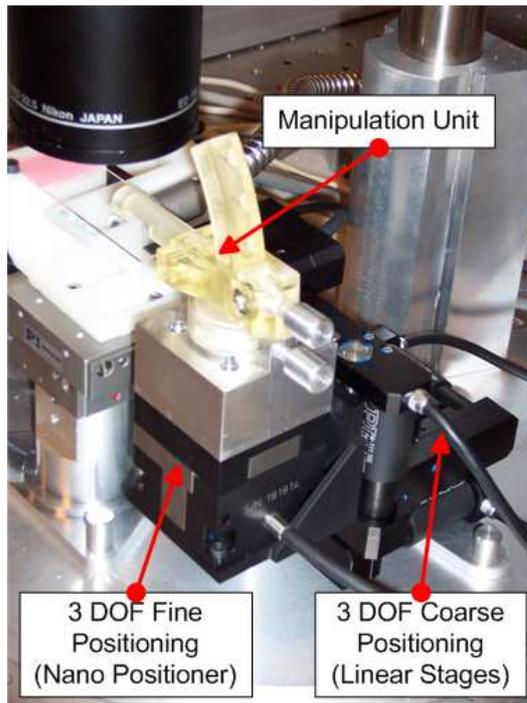


Figure 4.13: Motion unit of Manipulation Tool

ulation tool(s). 2 other linear stages are attached to each other for coarse motion of manipulation surface, where micro particles are deposited, in  $XY$  directions. In total, there are 8 DOF motion for MAW. Precision of linear stages are 7 nm while it is 1 nm for nano positioner.

Linear stages has the following specifications:

1. 7 nm Minimum Incremental Motion
2. 15 mm Travel Range
3. Velocity to 1.5 mm/sec
4. Closed-Loop DC Motors

while nano positioner has the following specifications:

1.  $100 \times 100 \times 100 \mu m$  Travel Range
2. Ultra-Compact:  $44 \times 44 \times 44$  mm (Closed-Loop Version)

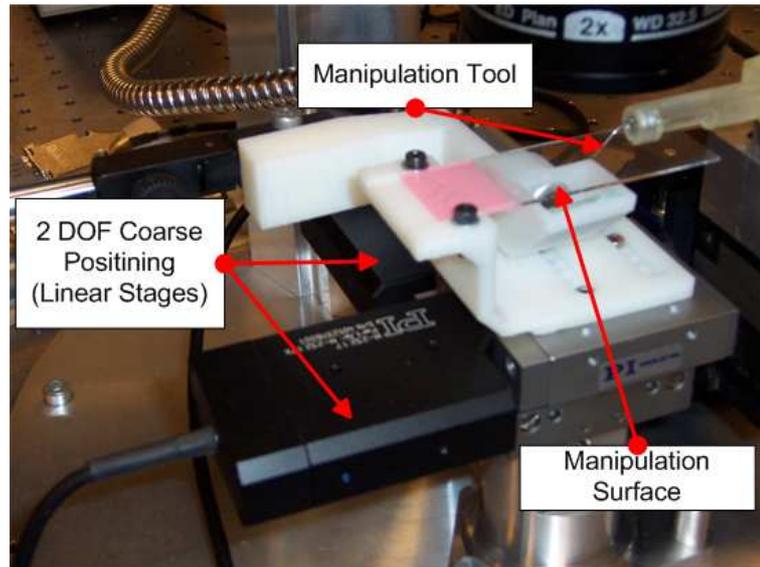


Figure 4.14: Motion unit of Manipulation Surface

3. 1 nm Resolution

#### 4.2.4 Manipulation Unit

Manipulation unit is very crucial part of hardware setup of MAW. To be able to manipulate micro particles, very precise and appropriate tools has to be selected. For micromanipulation and microassembly, 3 types of manipulation tools are used interchangeably. These are microgrippers, probes, and active probes. Probes and active probes provide pushing operation while microgrippers are used for pick & place operations of the micro particles. Each of them is attached to motion unit of manipulation tool(s) depending on the task to be implemented. Microgrippers have specifications of:

1. Available in a range of tip openings and thickness, as large as  $500 \mu m$ , to below  $1 \mu m$  openings.
2. Actuation of the gripper arms is available with power-on opening and power-on closing motion
3. Ideal grippers for handling delicate microscale components

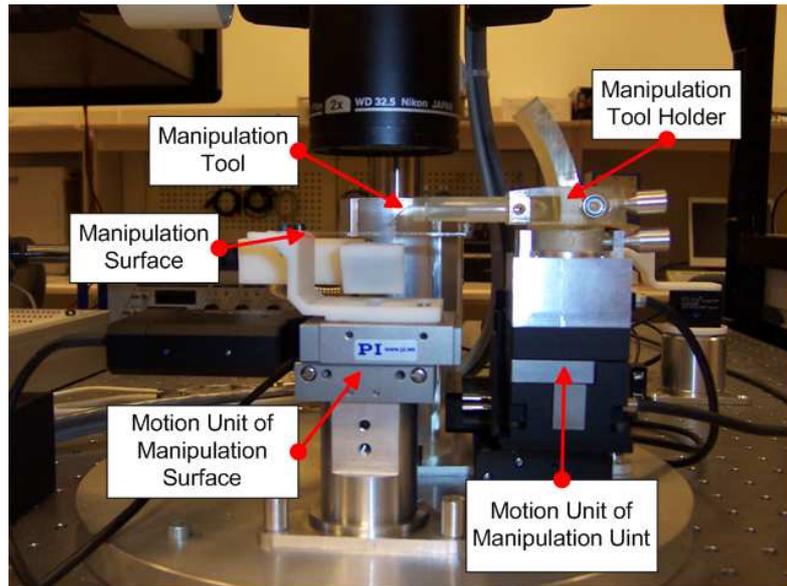


Figure 4.15: Manipulation Unit

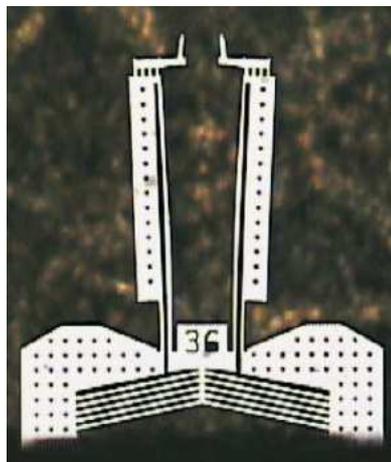
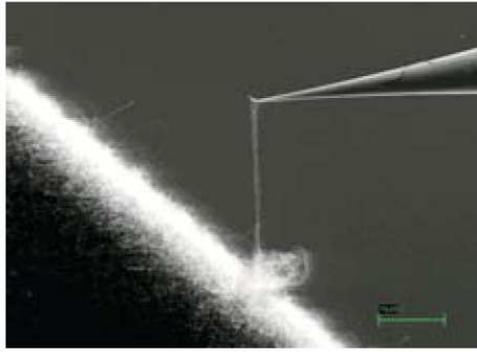


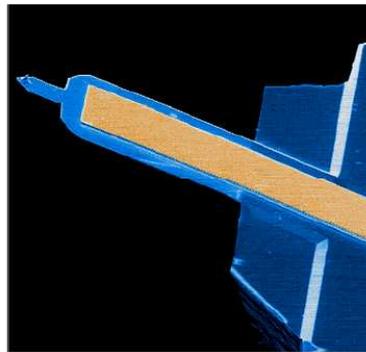
Figure 4.16: Pick & Place Manipulation Tool: Microgripper

- 4. Suitable for micro assembly operations and micro & nano particle manipulations

Probes are electrochemically etched 0.25 mm Tungsten polycrystalline wires used for probing integrated circuits, and research and development of nanoscale electrical



(a)



(b)

Figure 4.17: Pushing Manipulation Tools: (a) Probe (b) Active Probe

and mechanical systems. The key features are:

1. Tip radius of less than 50 nm
2. Capable of manipulating multi-wall and single wall carbon nanotubes and nanoparticles

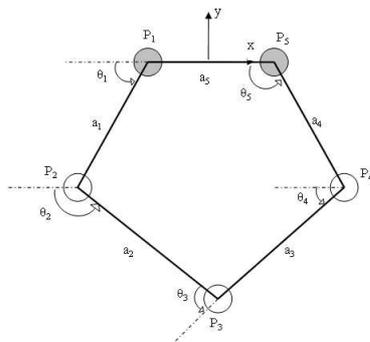
The active probe is a MEMS device, micro-machined from bulk silicon with a piezoelectric film patterned along a portion of the micro-cantilever. At the free end of the cantilever is a pyramidal tip with nanometer scale sharpness optimally shaped for high-resolution imaging. The micro-cantilever is capable of bending by the way of bimorph actuation resulting in a controlled vertical displacement of the tip.

On the other hand, the micro particles are another important part of manipulation. They are either based on *Polymethacrylate* or *Polystyrene*. They are available

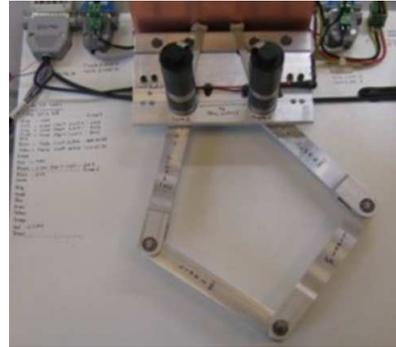
in different sizes. Polymethacrylate based micro particles have size of 40 and 70  $\mu m$  while Polystyrene based micro particles have size of 25, 8, and 2  $\mu m$ . They all have spherical shapes as depicted in Figure 4.10.

#### 4.2.5 Teleoperation Unit

Teleoperation unit has been designed to have force feedback while pushing or picking operations are being performed. Using the teleoperation unit, the optimum way to manipulate the micro particles without having deformation or stiction is determined according to the amount of force feedback. The unit consists of 2 DOF *Pantograph mechanism*, which actually is a 5 bar linkage driven by 2 motors as shown in Figure 4.18.



(a) Mechanism



(b) Unit

Figure 4.18: Teleoperation Unit & Pantograph (5 Bar Linkage) Mechanism

The key specifications for teleoperation unit are given by:

1. The haptic device should be free if simulating free space. In other words, the natural dynamics of the system such as friction, backlash and inertia should not be felt on the user side.
2. The device should be able to exert enough force to make the user feel really stiff, when simulating stiff objects. 20N/cm stiffness will be enough to convince a person that he/she is touching concrete wall.
3. The force that can be exerted by the device should not be easily saturated, so as to say, the force that can be exerted should be sufficient in amount to

represent most haptic interaction.

Teleoperation unit is implemented separately from whole MAW setup, and will be connected in the future. A simple analog joystick is used for the time being to control either 2 DOF coarse motion motion units for manipulation tool(s) or motion units for manipulation surface.

## Chapter 5

### Implementation of Vision System on MAW

This chapter provides simulation and experimental results about the ideas and techniques introduced in Chapter 2 and 3. Additionally, at the end of this chapter, special intrinsic image estimation and image deblurring techniques are implemented to get rid of nonuniform and blurring effects of illumination when reflected (episcopic) light illumination technique was being used for MAW.

#### 5.1 Camera Calibration

As stated in Section 3.1, the parameterization of the feedback is very crucial part for position based vision based control. Camera Calibration estimates the parameters that map pixels into physical units as shown in 2.3. In other words, camera calibration is the problem of calculation of every element of the camera matrix which is indeed multiplication of camera intrinsic and extrinsic matrices. The most preferred solution for camera calibration problem among computer vision society was proposed by Tsai [57, 58] who modeled the camera by 11 elements and solved this model to find extrinsic and intrinsic parameters of a digital camera. On the other hand, Zhang [59] introduced a flexible alternative method to find the parameters of a camera matrix by using homography technique.

##### 5.1.1 Formulation of Camera Calibration

It is very difficult to implement referred above techniques on MAW because they require a calibration pattern images, which generally have checkerboard pattern, with various orientations (preferably angles larger than 30 degrees) in 3-D space.

Capturing that kind of images from a microscope is not possible because of very small depth of field. For only very small angle variations, captured image can be seen as focused. Therefore, these techniques can provide unstable solutions for MAW. On the other hand, since we are imaging only focal planes, which can be either manipulation surface or manipulation tool(s), by the aid of microscope, the depth is known and constant for either case. Thus, we can approximate the image formation process by scaled orthographic projection model as indicated in Section 2.2. The depth which can be defined as the physical distance between manipulation surface and manipulation tool(s) is measured by autofocusing algorithm before teleoperated or semi-automated tasks are started.

To be able to establish a scaled orthographic projection between world space via the microscope into the camera space, we need to make some assumptions:

1. The coordinate systems of world frame and camera frame must be aligned with each other.
2. The projection is linear and depends only on used objective.
3. Since we work on focal planes,  $z$ -position is constant and known from prior information obtained by autofocusing operations.

Therefore our intrinsic camera matrix becomes as follows after making these assumptions:

$$M_{Int} = \begin{pmatrix} s_{ix} & 0 & 0 & 0 \\ 0 & s_{iy} & 0 & 0 \\ 0 & 0 & 0 & const \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.1)$$

where the scale factors  $s_{ix}$  and  $s_{iy}$  are dependent on the employing  $i^{th}$  magnification of the stereoscopic optical microscope. To work with this matrix, the image coordinates vector shown in equation 2.14 is formed to  $[x_1, x_2, x_3, 1]^T$ .

For the extrinsic matrix, we assume that  $X$ - $Y$  plane of the world coordinate system is parallel to image plane, which leads us to have rotation in  $Z$ -axis of the world coordinate frame with angle  $\theta$  to align  $X$ - $Y$  coordinate to each other, as well as coincidence of  $X$ - $Y$  plane with any of the focal planes of the microscope. This

results in zero value for the  $Z$  component for translation vector of extrinsic matrix which has the form of:

$$M_{Ext} = \begin{pmatrix} \cos \theta & -\sin \theta & 0 & t_1 \\ \sin \theta & \cos \theta & 0 & t_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.2)$$

Therefore, overall mapping for microscope vision system becomes as follows:

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 1 \end{pmatrix} = \underbrace{M_{Int} M_{Ext}}_P \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} s_{ix} \cos \theta & -s_{iy} \sin \theta & 0 & s_{ix} t_1 \\ s_{iy} \sin \theta & s_{ix} \cos \theta & 0 & s_{iy} t_2 \\ 0 & 0 & 0 & const \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (5.3)$$

which leads to the following equations:

$$x_1 = X s_{ix} \cos \theta - Y s_{iy} \sin \theta + s_{ix} t_1 \quad (5.4)$$

$$x_2 = X s_{iy} \sin \theta + Y s_{ix} \cos \theta + s_{iy} t_2 \quad (5.5)$$

$$x_3 = const \quad (5.6)$$

As it is seen from above equations,  $Z$  component of the world coordinates has no affect of calculation of image pixels. In addition to this,  $Z$  position is known a priori for the inverse transformation. We are also not interested in  $x_3$ , therefore we can reduce our world coordinate space by  $Z$  components. By this, we have mapping between 2-D image space to 2-D world space. Remember that this is possible because the representation of image formation is depicted by a linear model.

Our new reduced camera matrix, denoted by  $P^*$ , is given by:

$$P^* = \begin{pmatrix} s_{ix} \cos \theta & -s_{iy} \sin \theta & s_{ix} t_1 \\ s_{iy} \sin \theta & s_{ix} \cos \theta & s_{iy} t_2 \\ 0 & 0 & 1 \end{pmatrix} \quad (5.7)$$

then the mapping can be shown as follows:

$$\begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix} = P^* \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \quad (5.8)$$

It is also possible to determine the world coordinates by using inverse of our camera matrix, which is literally known as *reconstruction*. This is given by:

$$\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = (P^*)^{-1} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix} \quad (5.9)$$

After simplifying our camera calibration problem to our microscope vision system, estimation of elements of camera matrix  $P^*$  is left to complete camera calibration. It is done by least-square approximation which provides the closest estimated values to actual values. A priori it is known that last row of the camera matrix is  $[P_{31}^* \ P_{32}^* \ P_{33}^*]=[0 \ 0 \ 1]$ . Every image pixel coordinates and world coordinates pair yields two equations, therefore we need at least 3 pairs to solve 6 unknowns,  $P_{11}^*$ ,  $P_{12}^*$ ,  $P_{13}^*$ ,  $P_{21}^*$ ,  $P_{22}^*$ ,  $P_{23}^*$ , left for camera matrix. Certainly using more pairs makes the estimation more accurate and robust. Consequently, we have the following form of equation to solve for camera matrix:

$$\underbrace{\begin{pmatrix} x_1^1 & x_1^2 & \dots & x_1^n \\ x_2^1 & x_2^2 & \dots & x_2^n \\ 1 & 1 & \dots & 1 \end{pmatrix}}_I = P^* \underbrace{\begin{pmatrix} X^1 & X^2 & \dots & X^n \\ Y^1 & Y^2 & \dots & Y^n \\ 1 & 1 & \dots & 1 \end{pmatrix}}_W \quad (5.10)$$

where  $I$  and  $W$  are  $3 \times n$  matrices and  $P^*$  is  $3 \times 3$  matrix. From equation 5.10,  $P^*$  is solved by pseudo inverse operation:

$$\begin{aligned} I &= P^*W \\ IW^T &= P^*WW^T \\ P^* &= IW^T(WW^T)^{-1} \end{aligned} \quad (5.11)$$

By estimating the camera matrix  $P$ , we actually found the mapping from world coordinate frame to image coordinate frame. We need to have inverse mapping for teleoperated and semi-automated task because we extract pixel coordinates of image features and need to convert pixel values to physical units to be able to use them in the control loop of MAW.

### 5.1.2 Implementation of Camera Calibration

In order to have good camera calibration results, the following steps are applied:

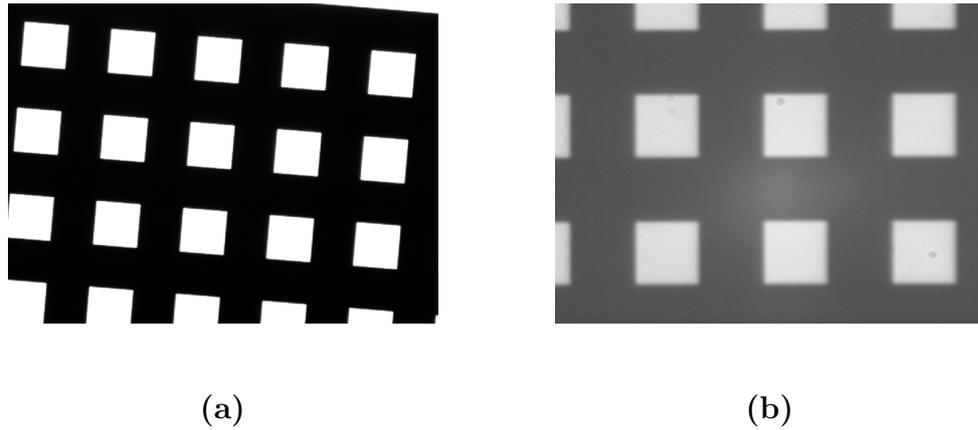


Figure 5.1: Micro Pattern Images captured from Coarse and Fine Cameras for  $2\times$  Magnification: (a) Coarse View Camera Image (b) Fine View Camera Image

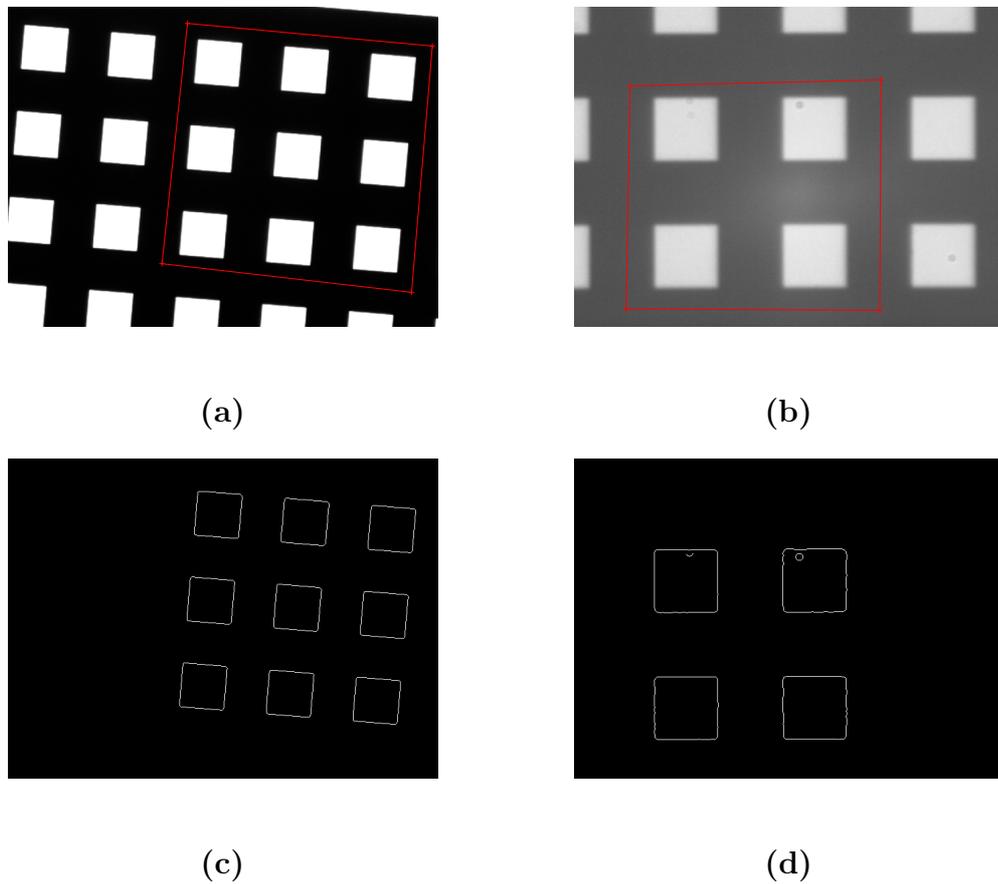


Figure 5.2: Determination of Region of Interest and Extraction of Edge Points: (a),(c) Coarse View Camera Image (b),(d) Fine View Camera Image

1. Micro pattern images are captured for each magnification of the stereoscopic optical microscope from both coarse camera and fine cameras. (See Figure 5.1)

- Edge points of selected regions are extracted to be used in Hough line detection algorithm for each captured micro pattern images. (See Figure 5.2)

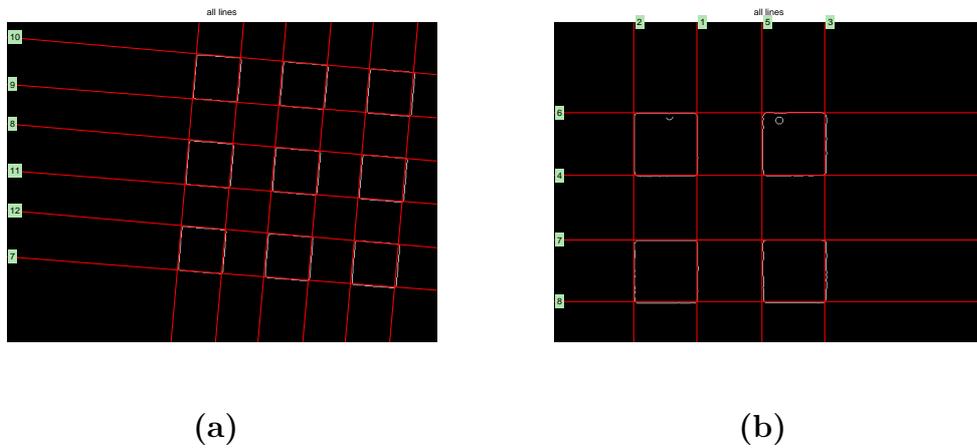


Figure 5.3: Detected Lines by Hough Line Detection Algorithm: (a) Coarse View Camera Image (b) Fine View Camera Image

- In order to find corners in the micro pattern images, lines are detected and their intersection points are obtained as corner points by Hough Line Transform. (See Figure 5.3)

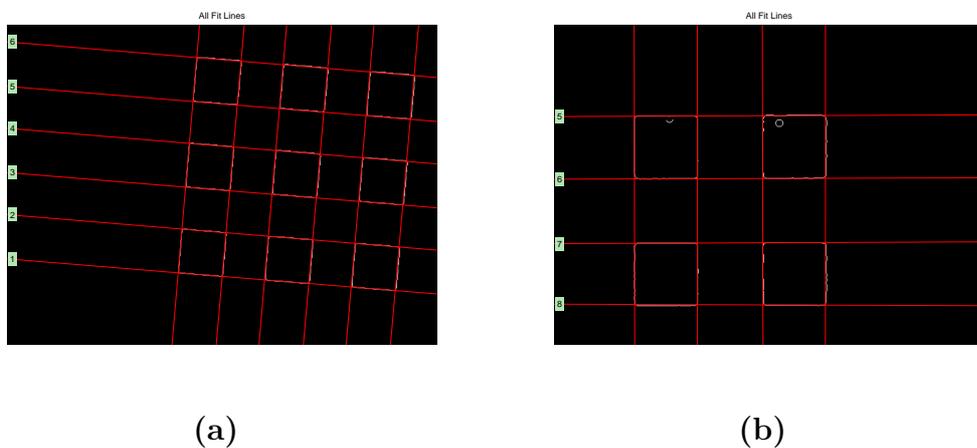


Figure 5.4: Detected Lines by Point-to-Line Line Fitting Algorithm: (a) Coarse View Camera Image (b) Fine View Camera Image

- Lines that are obtained by Hough Line Transform are observed that they are not fitted well to edge points on the edges of the squares. Nevertheless, it is possible to have better results. The idea is to find all edge points, which have

certain distance from lines found by Hough Line Transform, for each line, and then by using point-to-line fitting algorithm, new line equation parameters are found. This algorithm simply performs following steps:

- (a) A number of  $n \times 3$  matrices are formed from edge points according to each corresponding line found by Hough Line Transform. ( $n$  is number of edge points and one row of this matrix looks like:  $(x_n \ y_n \ 1)$ )
- (b) 2 unitary matrices ( $U$  and  $V$ ) and a diagonal matrix are calculated by SVD.
- (c) Column number where the element of diagonal matrix which has minimum value among others is located is found.
- (d) The corresponding column vector of  $V$  matrix is taken as line parameters and thus line fitting is completed. (See Figure 5.4)

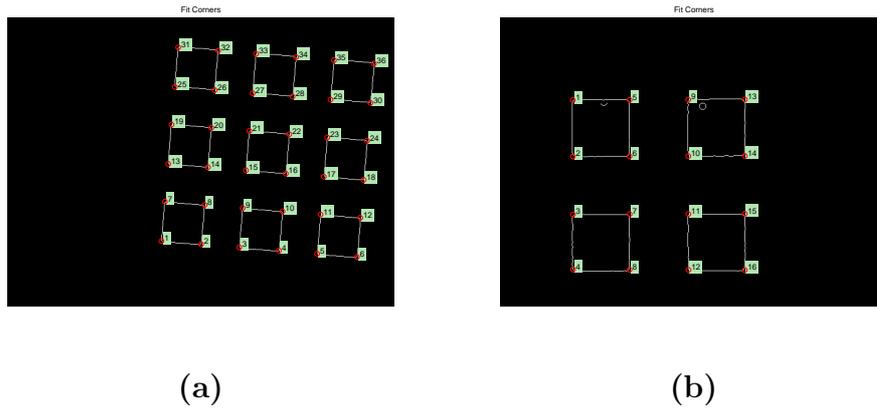


Figure 5.5: Detected Corner Points by Point-to-Line Fitting Algorithm: (a) Coarse View Camera Image (b) Fine View Camera Image

5. The intersection of these lines provide more precise and robust corner points with sub-pixel accuracies. Corner points found by line fitting algorithm are used in pseudo inverse formula given in equation 5.11 with given corresponding world coordinates. (See Figure 5.5) to estimate camera matrix  $P^*$ .
6. Accuracy of the system is verified in the sense of both pixel coordinates and world coordinates errors for each magnification for sample images. (See Table 5.1)

Table 5.1: Camera Matrices and Error Values of Pixel and World Coordinates for 2× Magnification

	<i>Coarse Camera Image</i>	<i>Fine Camera Image</i>
<i>Camera Matrix</i>	$P^* = \begin{pmatrix} 0.323 & -0.0279 & 255.641 \\ 0.0268 & 0.3224 & 372.130 \\ 0 & 0 & 1 \end{pmatrix}$	$P^* = \begin{pmatrix} 1.9024 & 0.0019 & 119.776 \\ -0.0037 & 1.8963 & 136.975 \\ 0 & 0 & 1 \end{pmatrix}$
<i>Inv. Camera Matrix</i>	$(P^*)^{-1} = \begin{pmatrix} 3.074 & -0.266 & -88.487 \\ 0.255 & 3.08 & 1080.64 \\ 0 & 0 & 1 \end{pmatrix}$	$(P^*)^{-1} = \begin{pmatrix} 0.5256 & -0.0005 & -62.887 \\ 0.001 & 0.5273 & -72.355 \\ 0 & 0 & 1 \end{pmatrix}$
<i>Av. Pixel Coord. Er.</i>	0.4233 pixel	0.4834 pixel
<i>Av. World Coord. Er.</i>	1.3116 $\mu m$	0.2544 $\mu m$

Table 5.2:  $\mu m$  Measurement of One Pixel for All Magnifications

$\frac{\mu m}{pixel}$	Coarse Image ( $\frac{1}{s_x}$ )	Coarse Image ( $\frac{1}{s_y}$ )	Fine Image ( $\frac{1}{s_x}$ )	Fine Image ( $\frac{1}{s_y}$ )
0.75×	8.1923	8.217	1.3806	1.3783
1×	6.1342	6.1399	1.0388	1.0404
2×	3.0859	3.0908	0.5242	0.5259
3×	2.0734	2.077	0.3548	0.3524
4×	1.54	1.5458	0.2661	0.2599
5×	1.2395	1.2375	0.2125	0.2147
6×	1.0229	1.023	0.1808	0.1814
7×	0.8789	0.8799	0.1486	0.1541
8×	0.7689	0.7698	0.129	0.1341
9×	0.6814	0.6815	0.1136	0.1087
10×	0.6143	0.6148	0.1042	0.1042
11.25×	0.5403	0.5429	0.0917	0.0934

### 5.1.3 Discussions

The following two tables show the accuracy of camera calibration technique that is implemented for each magnification. First table shows  $\mu m$  correspondence of one pixel in coarse and fine camera images for each magnification of stereoscopic optical microscope. (See Table 5.2) Second table indicates estimated pixel and  $\mu m$  value errors for each magnification. (See Table 5.3) For both coarse and fine camera images sub-pixel accuracies are reached and world coordinate errors are changing between 4  $\mu m$  and 0.1  $\mu m$  for coarse camera images, and are changing between 0.5

Table 5.3: Pixel and  $\mu m$  Errors for Image and World Coordinates for Each Magnification

	Coarse Im. Pixel Er.	Coarse Im. $\mu m$ Er.	Fine Im. Pixel Er.	Fine Im. $\mu m$ Er.
0.75 $\times$	0.4231	3.952	0.3491	0.482
1 $\times$	0.4935	3.0383	0.2355	0.2449
2 $\times$	0.4184	1.2968	0.4221	0.2274
3 $\times$	0.1325	0.2761	0.3861	0.1367
4 $\times$	0.3616	0.6494	0.2173	0.0696
5 $\times$	0.0981	0.1215	1.1111	0.2367
6 $\times$	0.2264	0.2316	0.2424	0.0441
7 $\times$	0.1683	0.148	0.207	0.0313
8 $\times$	0.2433	0.1872	0.1394	0.0185
9 $\times$	0.1854	0.1263	0.135	0.015
10 $\times$	0.2806	0.1725	0.222	0.231
11.25 $\times$	0.2714	0.1468	0.109	0.0101

$\mu m$  and 10 nm for fine camera images.

## 5.2 Image Features

In this section, image features extraction and object recognition implementations, which are essential parts of teleoperated and semi-automated MAW tasks, are demonstrated. As the implementations of edge detection and hough line transform were illustrated in the previous section while finding the best corners for given pattern images, circular hough transform and EFDs are described in detail in the next sections.

### 5.2.1 Circular Hough Transform

Thus far, it is clearly figured out that one of the main a priori information of MAW applications is that the micro particles to be manipulated have spherical shapes in 3-D world space and circular shapes in 2-D image. It is also well known fact that circular hough transform gives fast, very robust and precise response to circular closed form curves in the images. Circular hough transform also achieves sub-pixel accuracies for estimation of radius of the circles. This is the reason why it is selected to be performed for detection of the micro particles for MAW automation tasks.

Circular hough transform like any other hough transform takes edge image as

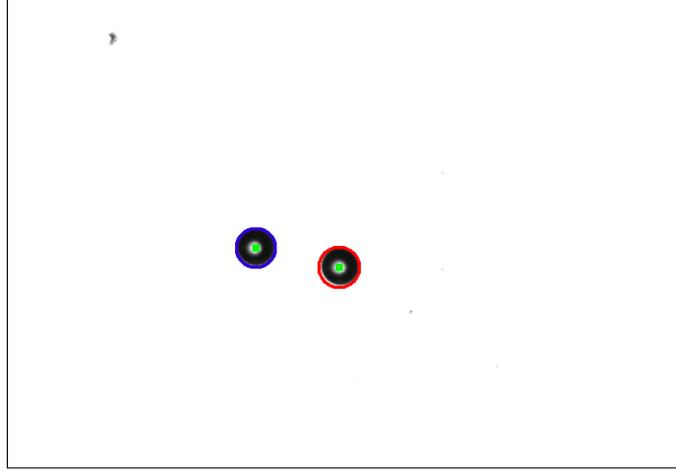


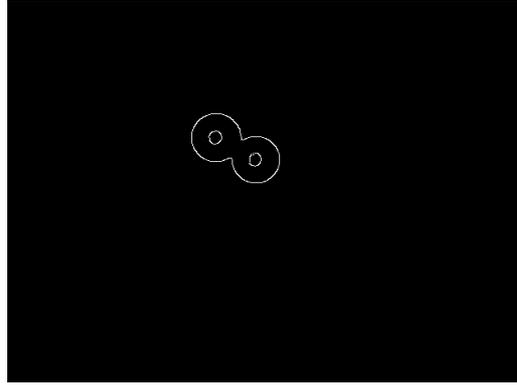
Figure 5.6: Circular Hough Transform of Micro Particles (4× Magnification)

input, and returns radius,  $R$ , and center pixel coordinates,  $(x_c, y_c)$  of a circle as image features. Figure 5.6 and Table 5.4 present detection of image features extracted from coarse camera view image in which there are two micro particles with approximately  $35\mu m$  radius. Each circle is denoted with different colors, but all centers are represented with green colors.

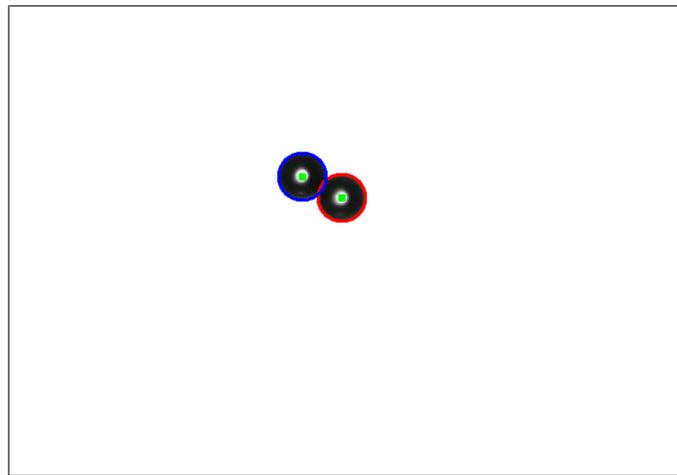
Table 5.4: Radius and Center of the Micro Particles by Circular Hough Transform (4× Magnification)

	$R$ in pixel	$R$ in $\mu m$	$x_c$	$y_c$
Upper Particle	25.318	38.98972	306	294
Lower Particle	24.3516	37.5015	204	270

It is always expected to have aggregated particles which stick to each other somehow as shown in Figure 5.7 in microscale world because of the presence of adhesion forces. Hence, MAW automation task algorithms should be as flexible as possible to handle every expected but undesirable situations like having aggregated particles in the workspace of the microscope. Although edge image of aggregated particles has edge points that represent only one closed curve instead of 2 distinct circles, circular hough transform returns 2 separate circles. It also must be pointed out that circular hough transform finds the circles even if their boundary edge points are not extracted fully. (See Figure 5.7) Table 5.5 designates extracted image



(a)



(b)

Figure 5.7: Edge Image and Circular Hough Transform of 2 Aggregated Micro Particles ( $5\times$  Magnification): (a) Edge Image (b) Circular Hough Transform

features from circular hough transform for both micro particles.

Table 5.5: Radius and Center of 2 Aggregated Micro Particles by Circular Hough Transform ( $5\times$  Magnification)

	$R$ in pixel	$R$ in $\mu m$	$x_c$	$y_c$
Upper Particle	29.0172	35.9668	304	198
Lower Particle	28.6007	35.4506	256	172

The following case is more complicated than the previous one. 4 micro particles are very close to each other and even some of them are stuck to each other. Circular

ough transform again gives very good results to this case as shown in Figure 5.8 and Table 5.6.

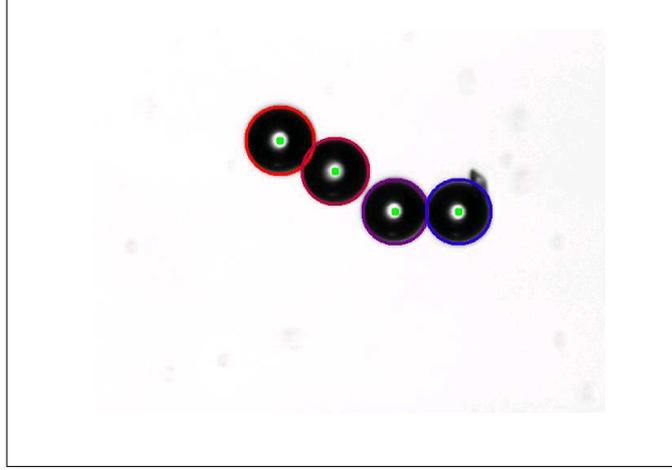


Figure 5.8: Circular Hough Transform of 4 Aggregated Micro Particles ( $7\times$  Magnification)

Table 5.6: Radius and Center of 4 Aggregated Micro Particles by Circular Hough Transform ( $7\times$  Magnification)

	$R$ in pixel	$R$ in $\mu m$	$x_c$	$y_c$
1 <sup>st</sup> Particle	42.1545	37.0495	234	140
2 <sup>nd</sup> Particle	41.0122	36.0456	302	178
3 <sup>rd</sup> Particle	39.8121	34.9908	376	228
4 <sup>th</sup> Particle	40.025	35.1779	454	228

### 5.2.2 Elliptic Fourier Descriptors

EFDs are well known for modeling any 2-D closed curve shapes. They are very good tools for recognition of very complex closed curves because they can model every small details of closed curves. However they can model undesirable noise too depending on how many number of harmonics are selected. Hence, the number of harmonics must be selected beforehand if the shape of the object is known a priori. Until now, only circular micro particles are concerned in MAW applications. Therefore, EFDs are not used in MAW automation tasks. Below 2 experimental

results are demonstrated to show difference between circular hough transform and EFDs.

In the first experiment, separated particles are modeled with 3 harmonics. (See Figure 5.9) If it is not known a priori that they are same micro particles, it can be decided by comparing the difference between each harmonic. (See Table 5.7 and Table 5.8) The standard deviation for difference of harmonics is 0.0403 for separated particles. By determining a threshold value higher than this standard deviation, the recognition of micro particles is possible.

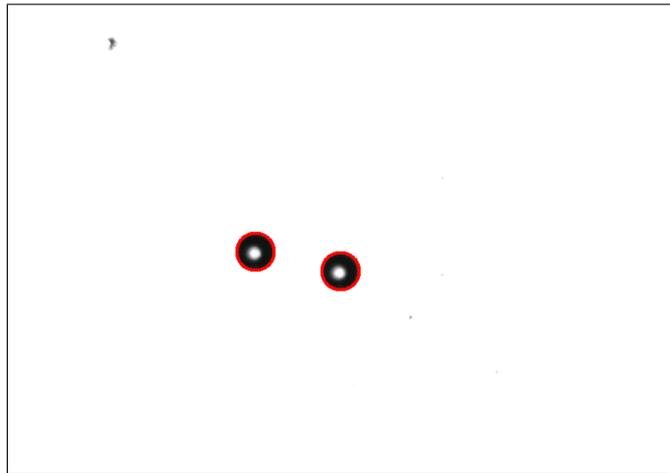


Figure 5.9: EFD of Micro Particles (4× Magnification)

Table 5.7: EFD Coefficients of the Upper Micro Particle for Each Harmonics (4× Magnification)

	$a_n$	$b_n$	$c_n$	$d_n$
1 <sup>st</sup> Har.	-4.12736	-23.0162	-23.1334	4.19657
2 <sup>nd</sup> Har.	-0.0513678	0.108738	0.158888	-0.166982
3 <sup>rd</sup> Har.	0.048114	-0.11599	0.140264	0.108356
	$x_c$		$y_c$	
	204		268	

In the case of aggregated particles, because EFDs model closed curves from their boundaries and the boundary of aggregated particles forms a different closed curve than circles, the closed curve shape is modeled as shown in Figure 5.9. EFDs perceive the aggregated particles as if they are only one micro object.

Table 5.8: EFD Coefficients of the Lower Micro Particle for Each Harmonics (4× Magnification)

	$a_n$	$b_n$	$c_n$	$d_n$
1 <sup>st</sup> Har.	-3.84169	-23.1142	-22.9512	4.01941
2 <sup>nd</sup> Har.	0.16192	0.100901	0.0650327	-0.0390597
3 <sup>rd</sup> Har.	0.0915017	-0.0961166	0.213565	0.0629452
	$x_c$		$y_c$	
	307		292	

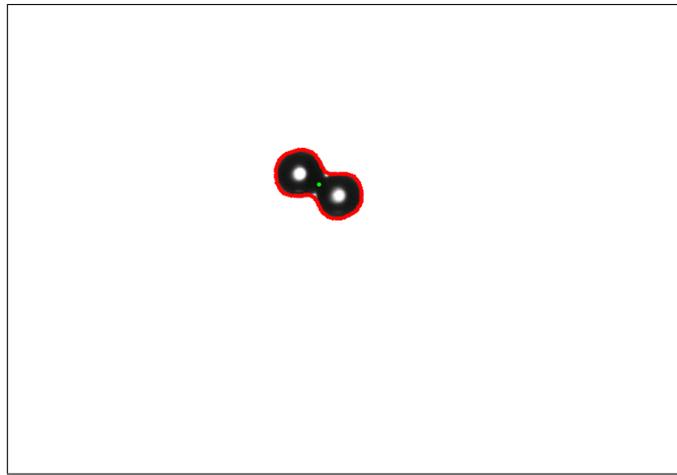


Figure 5.10: EFD of 2 Aggregated Micro Particles (5× Magnification)

The standard deviation between harmonics for upper micro particle in separated particles image and harmonics for aggregated particles is estimated as 6.7246 while the standard deviation between harmonics for lower micro particle in separated particles image and harmonics for aggregated particles is estimated as 6.7403. When these values are compared with the value estimated in the first case, it is obvious that they are different from each other.

### 5.2.3 Discussions

As demonstrated above, it is essential to use circular hough transform instead of EFDs to model micro particles for MAW automation tasks because of its robust response even for aggregated particles case. On the other hand, although EFDs model aggregated particles as they are formed as one micro object, they are proven to

Table 5.9: EFD Coefficients of 2 Aggregated Micro Particle for Each Harmonics ( $5\times$  Magnification)

	$a_n$	$b_n$	$c_n$	$d_n$
1 <sup>st</sup> Har.	-33.6402	-33.6846	-34.1196	11.0231
2 <sup>nd</sup> Har.	0.453389	-0.12998	-0.305526	0.183142
3 <sup>rd</sup> Har.	4.0255	-5.16066	-8.60738	-1.11431
	$x_c$		$y_c$	
	307		292	

be a beneficial tool for recognition of any 2-D closed curves by experimental results. Furthermore, their sensitivity to small changes can be used to model deformations of the micro particles to estimate force from visual information as future advancement of MAW.

### 5.3 Automation Task Applications on MAW

This section demonstrates experimental results for sample teleoperated and semi-automated MAW tasks. Teleoperated tasks are divided into two parts as teleoperated tasks using microgripper for pick & place applications and teleoperated tasks using probe for pushing operations. In the semi-automated tasks only probe is used as manipulation tool.

#### 5.3.1 Teleoperated Tasks

Tasks, which an operator controls execution of motion and manipulation units manually by means of user interface, are generally named as teleoperated tasks. In these tasks, no visual data is used in the control loop of MAW as feedback. Teleoperation tasks are examples of static look & move structure vision based control. The operator decides the reference value of the state(s) and there is no visual feedback until the actual value of the state(s) reaches to the reference value with the motion control of MAW inside. The general structure of teleoperated tasks for MAW is illustrated in Figure 3.1.

### **Teleoperated Tasks using Microgripper as Manipulation Tool**

During this task, the operator controls the microgripper by either clicking mouse for home and target points to generate reference trajectory or joystick. As it will be described in Section 5.5, the level of microgripper and the micro particles along  $z$ -axis must be same. It is provided from the vertical distance obtained by autofocusing. Once, they are on the same level along  $z$ -axis, in other words on the same focal plane, the operator moves the microgripper very close to the micro particle as a first step. In the second step, the operator determines the opening of the microgripper according to the diameter of the micro particle. The opening of the microgripper should be larger than the diameter for the micro particle to be located in such a way that fingers of the microgripper seize the micro particle without touching it. After that the opening of the microgripper is decreased to hold the micro particle in the third step. The pick part of the teleoperated task is completed by that. Place task is just to move the microgripper to desired position and do the opposite operations for pick task. The illustration of pick & place teleoperated task is shown in Figure 5.11.

### **Teleoperated Tasks using Probe as Manipulation Tool**

In this task, the operator controls the probe by mouse clicking of home and target points to generate reference trajectory as in the case of the microgripper in the previous task example. Since the camera calibration is done, the reference as the difference between home and target points is calculated in 3-D world frame. The difference of this teleoperated task from the previous task is that the micro particles are recognized by the vision system using circular hough transform and their corresponding target positions are generated automatically. A simple trajectory is denoted by a line between center of the micro particle and the target point. The operator moves the probe very close to the micro particle and push the micro particle along the trajectory line with a meaningful amount. Some steps of this teleoperated task is shown in Figure 5.12.

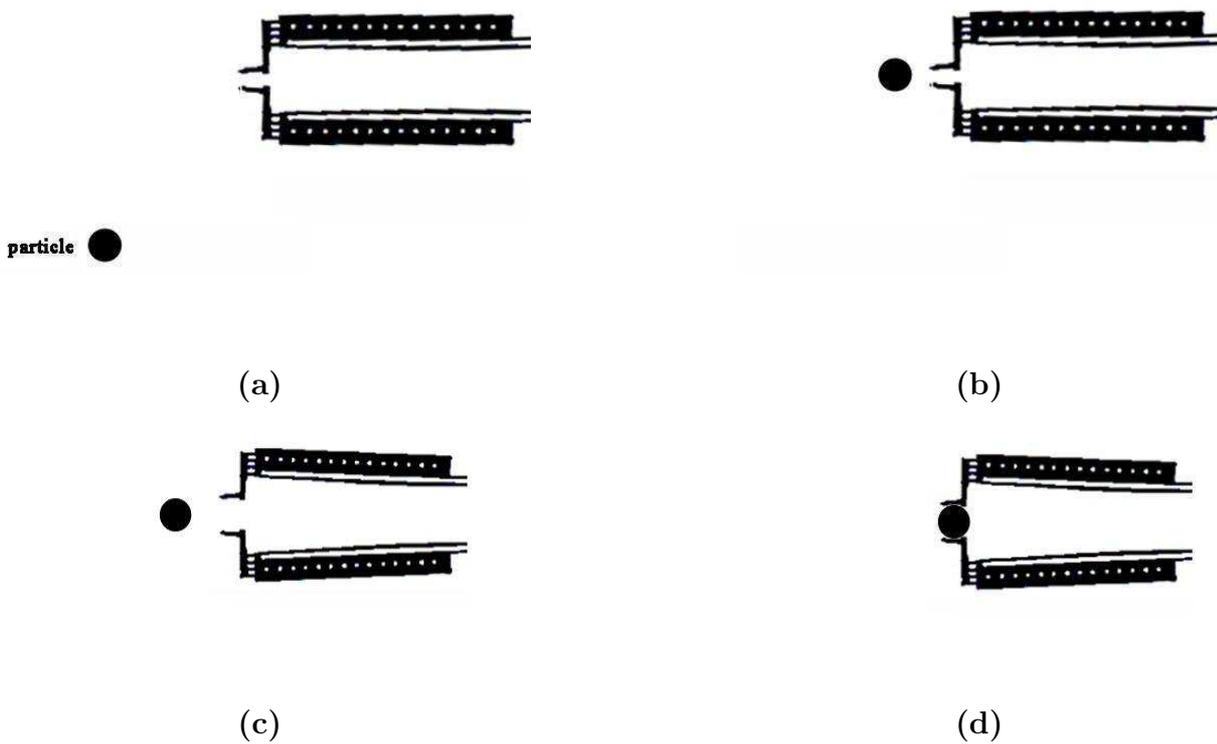


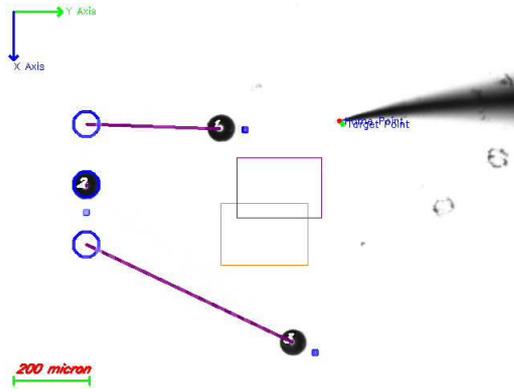
Figure 5.11: Pick & Place Teleoperated Task using Microgripper: (a)Initial State (b) Approaching to the Micro Particle (c) Arranging the Opening of the Microgripper (d) Picking the Micro Particle

### 5.3.2 Semi-Automated Tasks

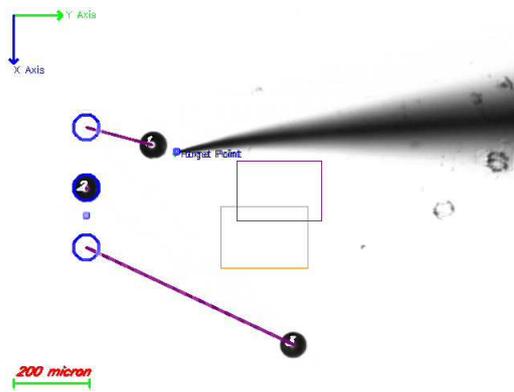
If a task is initiated by the operator, but it is executed automatically until it is finished afterwards, this task is called as semi-automated task for MAW. Semi-automated task is an example of Dynamic Look & Move structure vision based control. Because visual feedback is converted to physical units, it is, in fact, PB-DLM structure. The operator selects the micro particle and its target position, and then, by the aid of probe as manipulation tool, the micro particle is manipulated automatically until it reaches to its target position.

As mentioned in the previous section, vision system detects the micro particle and its target position. The trajectory of the micro particle is represented by a line drawn between center of the micro particle and its target position. Steps of the semi-automated task can be summarized as:

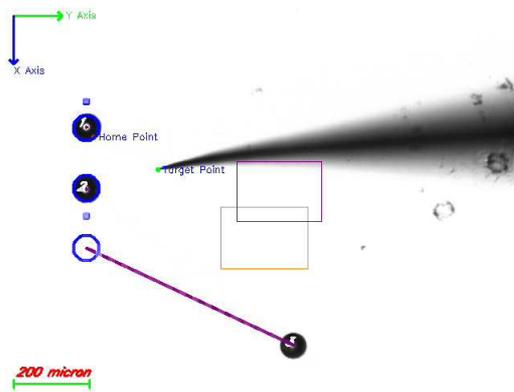
1. Push position for the probe is determined according to the direction of the trajectory line. Push position is located outside of the boundary of the micro



(a)



(b)



(c)

Figure 5.12: Steps of Pushing Teleoperated Task using Probe: (a) Initial State (b) Middle of the Task (c) End of the Task

particle with a certain amount along the trajectory line. (For the experiments depicted below, push position is  $5\mu m$  far from the boundary of the micro particles)

2. The probe is moved to the push position, and it pushes the micro particle with  $15\mu m$  in the direction of the trajectory line. The relative position reference of  $X$  and  $Y$  in 3-D world frame is calculated each time according to slope of the trajectory line.
3. After pushing the micro particle, the probe is moved back to its previous position by moving it with  $-15\mu m$  in the opposite direction of the trajectory line. By the time, new direction of the trajectory line is determined according to new position of the micro particle relative to its target position.
4. Once new push position is determined, the probe is moved to new push point and performs the same procedure until it reaches to target position.

In this case, the probe is not detected and tracked by the vision system during the semi-automated task, yet the micro particle is tracked. Initial position of the probe is a priori known. The verification of whether the probe is moved to desired position or not is done by checking the actual position data received from control unit of MAW. Since the camera calibration has sub-pixel accuracy and the estimation of a pixel point to a world coordinate is precise enough for initial semi-automated tasks, the method is trustable to complete sample tasks as shown in Figure 5.13 and Figure 5.14 into some extent.

In the first experiment, a single micro particle is manipulated towards a selected target position and forming a line pattern is the goal in the second experiment. Instants from semi-automated tasks of each particle is presented.

### 5.3.3 Discussions

The succes of teleoperated and semi-automated tasks depend on several environmental conditions. Above experiments conclude that the accuracies of the control unit and vision system are enough to perform automation tasks. However, some external effects can prevent these tasks to be accomplished. If the micro particles

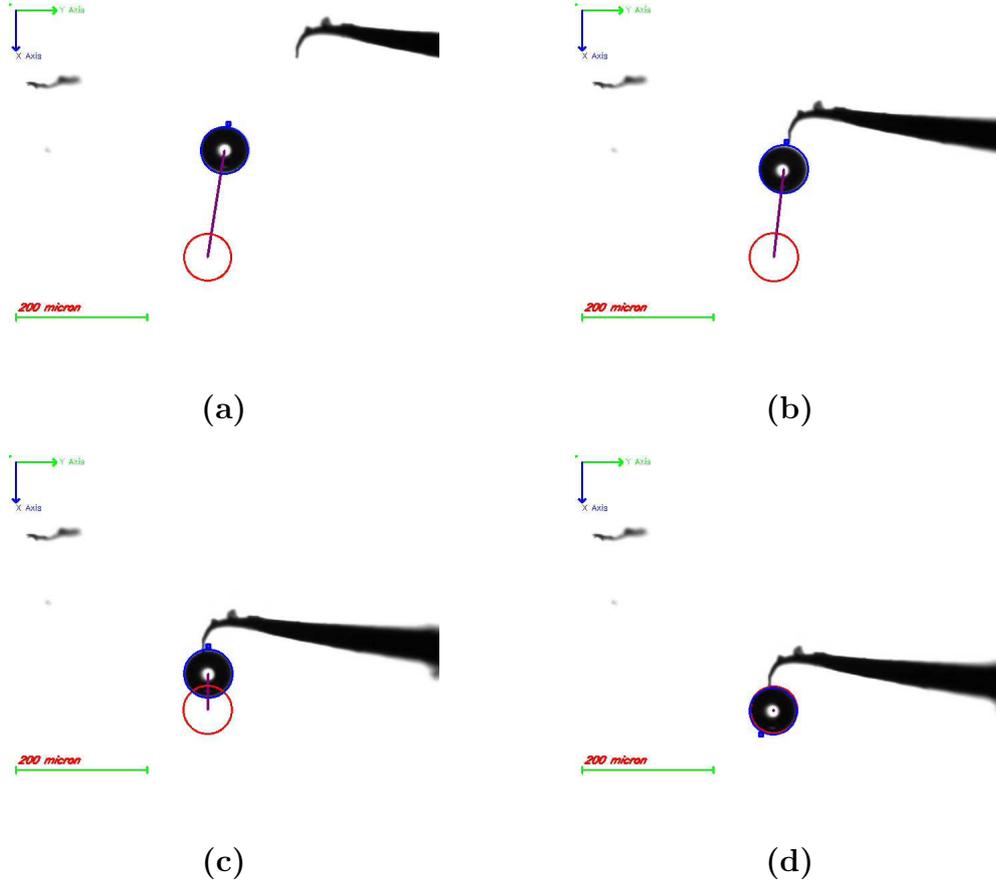


Figure 5.13: Steps of Pushing Semi-Automated Task using Probe for One Micro Particle: (a) Initial State (b) Approaching to the Micro Particle (c) Pushing the Micro Particle (d) Final State

stick to the manipulation surface by any reason, even small impulse is able to throw the micro particle to outside of the workspace. On the other hand, the micro particles can stick to the microgripper and the probe since adhesion forces are dominant in micro scale. For the case of automated tasks, motion planning has to be done in such a way that if there are any obstacles exist between a micro particle and its target position, the trajectory cannot be expressed as line. It has to be defined by more complex function(s). For future applications, these conditions must be taken into consideration to increase accuracy and repeatability of the overall system.

## 5.4 Motion Parameter Estimation

In this section, implementation of motion parameter estimation on MAW will be illustrated. Why motion parameter estimation can be important for MAW applica-

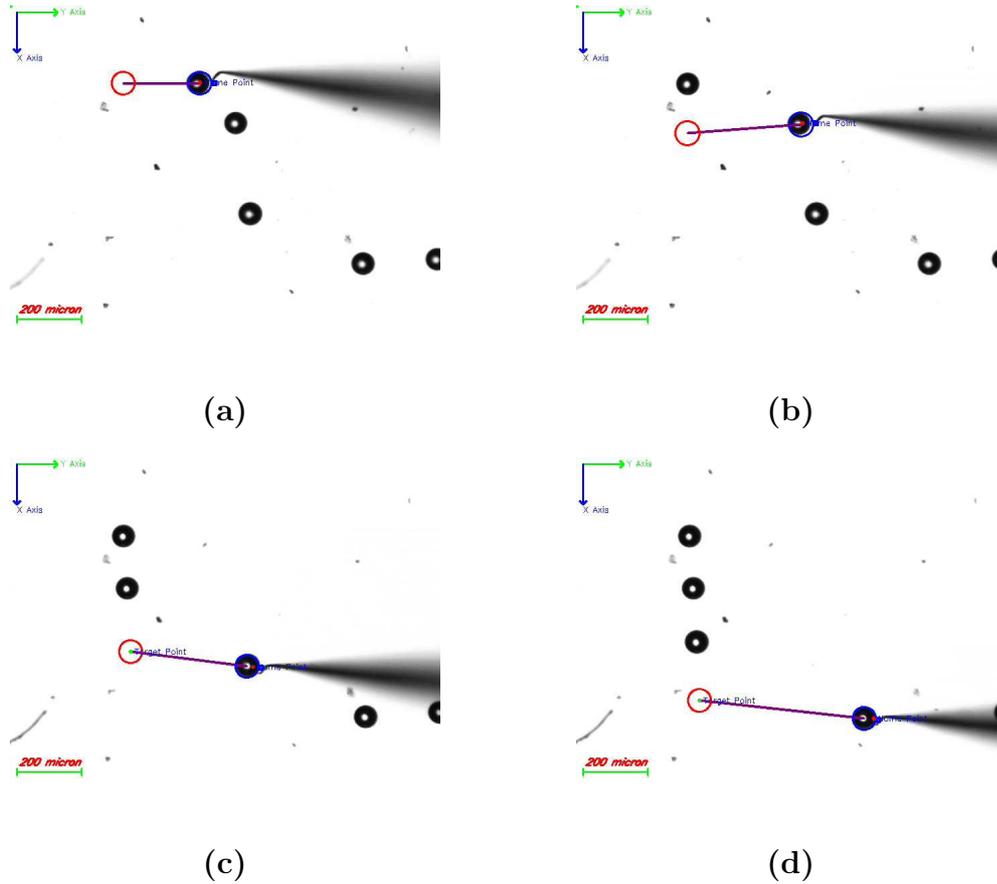


Figure 5.14: Steps of Making a Line Pattern by Semi-Automated Tasks: Manipulation of (a) First Particle (b) Second Particle (c) Third Particle (d) Fourth Particle

tions is because for example, consider a MAW application in which there are micro particles on the manipulation surface that is moving, a sample automation task can be defined as that a microgripper tries to grasp one of the micro particles to place somewhere else on the manipulation surface. In order to achieve this, the motion of the microgripper must be synchronized with the motion of the manipulation surface. Hence, the motion of manipulation surface is needed to be estimated by visual information. In this case, fast and robust motion parameter estimation algorithm is necessary for synchronization.

As stated in Section 2.5.3, the basic procedure introduced for motion parameter estimation is to recast image feature dynamics linearly in terms of unknown parameters and construct a sliding mode observer to produce asymptotically correct estimates of the observed image features, and then use “equivalent control” to explicitly compute parameters.

### 5.4.1 Formulation of Sliding Mode Observers

In this section, an observer whose state follows the state of motion of an extracted image feature as closely as possible is constructed. SMC in the design of the observer will be employed to stabilize the state estimation error around zero. Basically the existence image feature dynamics, which was constructed by equations 2.39 and 2.40, is copied and is tried to control this copied version by SMC. More precisely, let the observer be

$$\dot{\hat{\Xi}} = u \quad (5.12)$$

where  $u$  will be designed using SMC so that  $\tilde{\Xi} = \Xi - \hat{\Xi} \rightarrow 0$  as  $t \rightarrow \infty$ .

Let us define the sliding mode manifold as

$$\sigma = Y - \hat{\Xi},$$

which then implies that

$$\dot{\sigma} = \dot{Y} - \dot{\hat{\Xi}} = F\varphi - u. \quad (5.13)$$

Let us pick the following Lyapunov function

$$V = \frac{1}{2}\sigma^T\sigma$$

whose time derivative is

$$\dot{V} = \sigma^T\dot{\sigma}$$

which can be made negative definite by setting  $\dot{\sigma}$  to either  $-MSgn(\sigma)$ , where  $M > 0$  and  $Sgn(\cdot)$  is the signum function, or  $-D\sigma$ , where  $D$  is a positive definite matrix. If  $-MSgn(\sigma)$  is selected, all components of the control are switching between lower and upper bound of control. This may cause unnecessary chattering in the system especially in the discrete-time implementation of the control algorithm. Combination of the  $\dot{\sigma} = -MSgn(\sigma)$  and  $\dot{\sigma} = -D\sigma$  by selecting  $\dot{\sigma} = -D\sigma - \rho(x, t)Sgn(\sigma)$  yields a solution that may combine good properties of both solutions and allows selecting  $\rho(x, t)$  small enough to minimize chattering and at the same time to guarantee the existence of sliding mode.

So by selecting  $\dot{\sigma} = -D\sigma - \rho(x, t)Sgn(\sigma)$ ,  $\dot{V}$  then becomes

$$\dot{V} = -\sigma^T D\sigma - \rho(x, t)\sigma^T Sgn(\sigma) = -\sigma^T D\sigma - \rho(x, t)\|\sigma\|,$$

which is clearly negative definite since  $D > 0$  and  $\rho > 0$ . Therefore for stability,

$$\dot{\sigma} = -D\sigma - \rho Sgn(\sigma) \Rightarrow \dot{\sigma} + D\sigma + \rho Sgn(\sigma) = 0 \quad (5.14)$$

must be satisfied.

As shown in [40],  $u$  can be computed recursively using the algebraic distance  $\dot{\sigma} + D\sigma$ , namely

$$\begin{aligned} u(k) &= u(k-1) + (\dot{\sigma} + D\sigma)|_k \\ &= u(k-1) + \left\{ \frac{\sigma(kT) - \sigma((k-1)T)}{T} + D\sigma(kT) \right\} \end{aligned} \quad (5.15)$$

or,

$$u(k) = u(k-1) + \frac{1}{T} \{ (I + TD)\sigma(kT) - \sigma((k-1)T) \} \quad (5.16)$$

where  $I$  is the identity matrix and  $T$  is the sampling time.

In light of (5.14), the algebraic distance used in (5.4.1) as  $\dot{\sigma} + D\sigma + \rho Sgn(\sigma)$  can be modified and the control can be computed again recursively as:

$$\begin{aligned} \hat{u}(k) &= \hat{u}(k-1) + (\dot{\sigma} + D\sigma + \rho Sgn(\sigma))|_k \\ &= \hat{u}(k-1) + \frac{1}{T} \{ (I + TD)\sigma(kT) - \sigma((k-1)T) + \rho Sgn(\sigma(kT)) \} \end{aligned} \quad (5.17)$$

Since  $F$  is square and invertible, plugging (5.13) and (5.17) in (5.14) implies

$$F\hat{\varphi} - \hat{u} + D\sigma + \rho Sgn(\sigma) = 0 \Rightarrow \hat{\varphi} = F^{-1}(\hat{u} - D\sigma - \rho Sgn(\sigma)). \quad (5.18)$$

Note that when  $\dot{\sigma} + D\sigma + \rho Sgn(\sigma) \rightarrow 0$  then  $\hat{u} \rightarrow \hat{u}_{eq}$ , namely we have the “*equivalent control*”,  $\hat{u}_{eq}$ , on the sliding manifold. In other words, the parameters are being estimated using the equivalent control.

## 5.4.2 Implementation of Motion Parameter Estimation

Some initial real time experiments were conducted with the vision setup shown in Figure 4.6. What is observed as a motion is created by a nano positioner, which can provide nanometer range accuracy motion and positioning in XYZ directions as indicated in Section 4.2.3. The points of interest to be extracted and tracked are chosen from micrometer, on which has a micron range checker board pattern, on top of nano positioner using some image processing operations. Each (1/30) second, new image is captured, processed, and the points of interest are extracted from the

image data. To synchronize the sampling time of capturing images and the sampling time of the control algorithm, the well known interpolater,  $G(s) = \frac{w_n^2}{s^2 + 2\zeta w_n s + w_n^2}$  with  $w_n = 50, \zeta = 1$ , which creates smooth transition among the values of the states between two consecutive frames, is used.

### Linear Motion along x-Axis

In this case, nano positioner moves along  $x$ -axis with  $6 \mu\text{m}/\text{s}$  for a while and stops, and then turns back to opposite direction again with the same constant velocity. Since there are no linear motion along  $y$ -axis, the estimated linear velocity along  $y$ -axis is around zero. Furthermore, since there is no rotational motion, the estimated angular velocity is around zero too. (See Figure 5.15 and Figure 5.16)

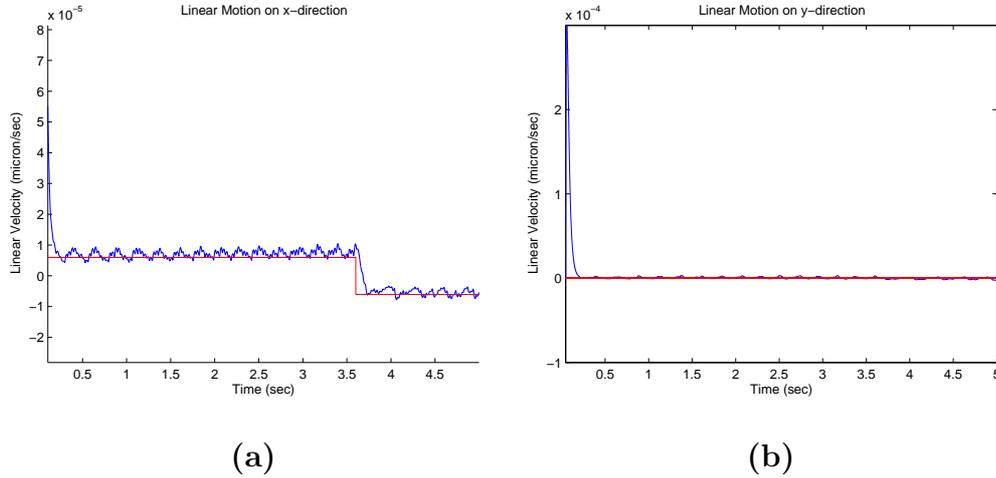


Figure 5.15: Actual and Estimated Motion along x- and y-axes: (a) Actual (red) and Estimated (blue) linear velocity along  $x$ -axis, (b) Actual (red) and Estimated (blue) linear velocity along  $y$ -axis

### Pure Rotational Motion

In this case, nano positioner undergoes a pure rotational motion with a constant angular velocity,  $\pi/5$  rad/s in counterclockwise. The radius of the circular motion is  $10 \mu\text{m}$ . Actual and estimated angular velocities are superimposed in Figure 5.17.

Although, the reference motion is pure rotation, the estimated angular velocity becomes very large for small time interval at the very beginning. Then it converges

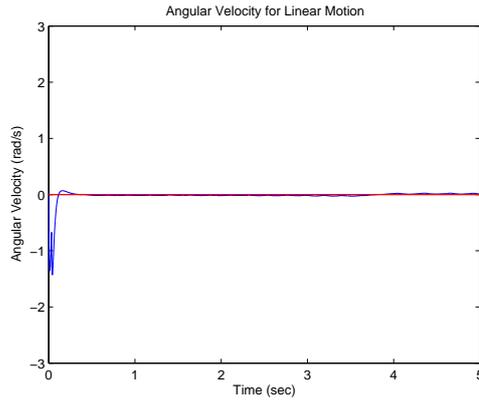


Figure 5.16: Actual (red) and Estimated (blue) Angular Velocities,  $\omega$  and  $\hat{\omega}$

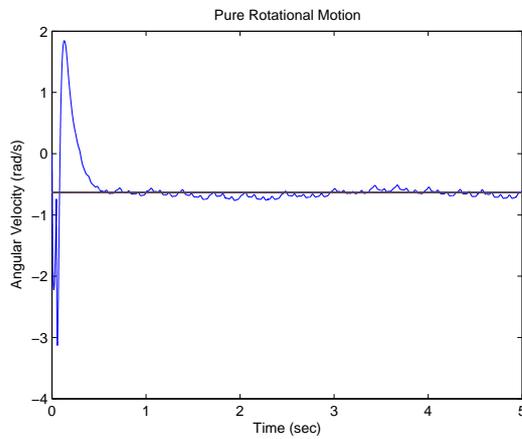


Figure 5.17: Actual (black) and Estimated (blue) Angular Velocities,  $\omega$  and  $\hat{\omega}$

to actual value very quickly. At the same time, the estimated trajectory seems to undergo linear motion (See Figure 5.18). The reason is that the given initial pixel points, which are given randomly as initial conditions for the motion parameter estimation algorithm, are different from the actual pixel points on the circle. Therefore estimated points show some linear behavior until the dynamic of the system reaches the sliding manifold. When the system is on the sliding manifold, the estimated values converge to the actual values and hence the motion becomes circular.

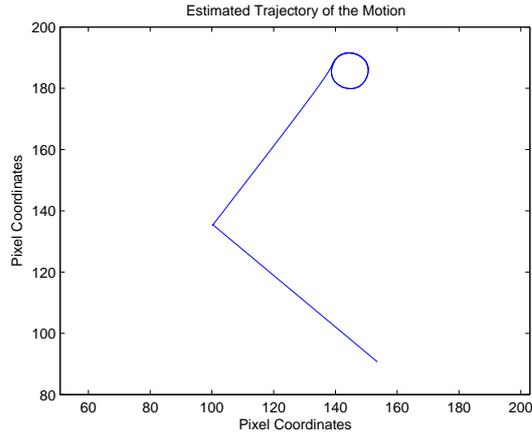


Figure 5.18: Estimated Motion Trajectory

### 5.4.3 Discussions

A new framework is developed to solve motion and structure estimation problems in machine vision using sliding mode observers [60,61]. It has been shown that most of the dynamics encountered in MAW applications can be recast as dynamical systems which are linear in terms of parameters to be estimated. An appropriate sliding observer whose output follows the output of the original dynamics asymptotically is then constructed. Control which drives the observer has been computed recursively based on the algebraic distance  $\dot{\sigma} + D\sigma + \rho Sgn(\sigma)$ .

## 5.5 Autofocusing

It is denoted in Section 2.6 that autofocusing is a tool to estimate the vertical distance and to determine location of the focal plane(s). In the case of MAW, because of non-planar micro particles and manipulation tool, there are at least 2 focal planes. According to works done about the comparison of which autofocusing algorithm has the best performance, it is concluded that normalized variance method has the most robust response among others to optical microscopic images.

The manipulation tool(s) and micro particles are away from each other before MAW teleoperated or semi-atomated tasks is started. The first condition to be completed before the tasks are initialized is to move the manipulation tool to level

of micro particles along  $z$ -axis. The distance between them is estimated by autofocusing operation. In order to estimate  $z$  distance, autofocusing algorithm must be able to locate focal planes of both the manipulation tool and the micro particles. The distance between the micro particles and the manipulation tool is calculated with accuracy of  $0.8 \mu m$ . In fact this is the accuracy of auto focus device. (See Section 4.2.1)

The steps performed by autofocusing algorithm are presented briefly as:

1. The microscope moves to its physical limits along  $z$ -axis and this location is set as the home position which is denoted as " $z=0$ ".
2. The microscope is moved to a specific position where it is a priori known that it does not encounter with any tools, neither the manipulation tool nor the micro particles. During this motion normalized variance is not calculated from captured images.
3. From its new position, it is traveled over a part of its whole workspace where a priori it is known that it definitely runs across both the manipulation tool or the micro particles in downward direction.
4. Over this travel, the normalized variance is calculated from each captured image from coarse camera.
5. According to attained normalized variance values during this travel, local maximum values are calculated. Two largest values represent either the manipulation tool focal plane or the micro particles focal plane. Local maximum values are calculated by checking the first derivative of each normalized variance value with respect to previous and next points. If the derivative in respect of previous point is positive and in respect of next point is negative, this value is labeled as a local maximum.
6. Determination of which local maximum value represents the manipulation tool or the micro particles is very simple. Since the microscope runs across the manipulation tool at first and then the micro particles, the local maximum at small  $z$  value is the manipulation tool.

7. Once the  $z$ -axis locations of the manipulation tool and the micro particles are determined, the vertical distance is just equal to their differences.

In MAW applications, there can be 3 different scenarios which changes according to the position of the manipulation tool and the number of micro particles in the captured microscopic images. In the first case, the normalized variance of the manipulation tool is bigger than the normalized variance of the micro particles. It is the opposite for second case and the normalized variances are equal in the third case. First two cases are implemented and the experimental results are demonstrated below.

### 5.5.1 First Case

In the first case, the manipulation tool covers wider area in the image than the micro particles as shown in Figure 5.19. Therefore, its normalized variance value



Figure 5.19: Focal Planes of The Manipulation Tool and The Micro Particles for the First Case: (a) The Manipulation Tool (b) The Micro Particles

is bigger than the normalized variance value of the micro particles. Because  $z$ -axis position of the manipulation tool is smaller and the normalized variance value of it is bigger in this case, global maximum of the graph shown in Figure 5.20, indicates the manipulation tool. Second peak is where the focal plane of the micro particles is located.

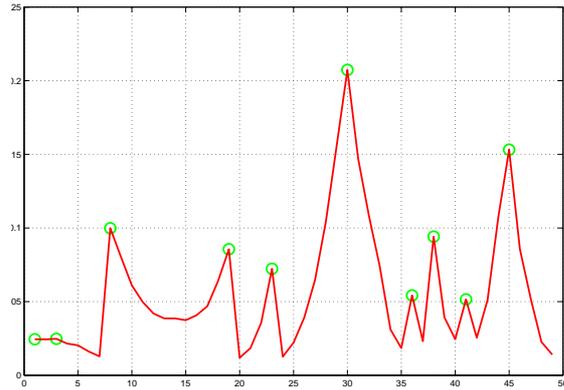


Figure 5.20: The Normalized Variance Graph of the First Case

### 5.5.2 Second Case

In the second case, the normalized variance value of the micro particles is bigger as opposite to the first case. Focal planes are presented in Figure 5.21.

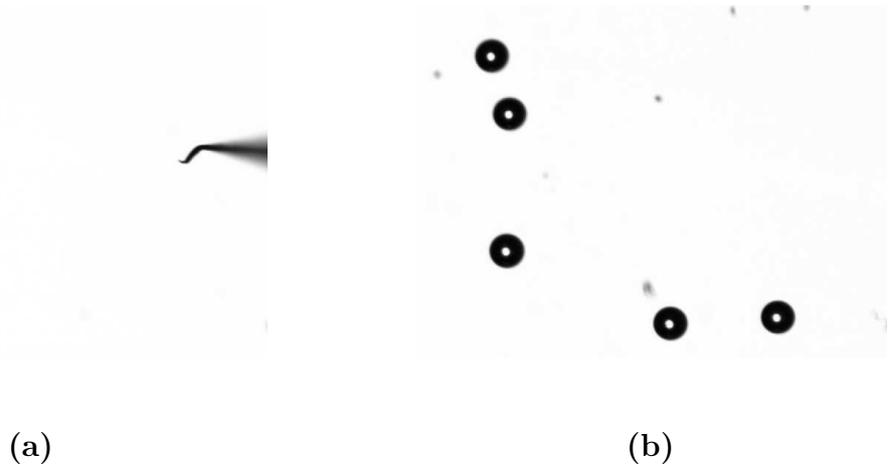


Figure 5.21: Focal Planes of The Manipulation Tool and The Micro Particles for the Second Case: (a) The Manipulation Tool (b) The Micro Particles

Figure 5.22 shows the normalized variance values graph. The biggest peak designates the focal plane of the micro particles in this case.

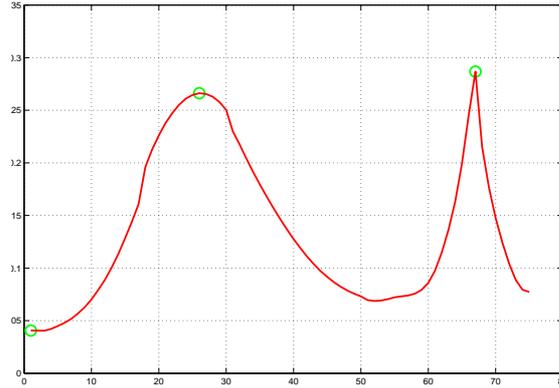


Figure 5.22: The Normalized Variance Graph of the Second Case

### 5.5.3 Discussions

The main problems arise if the manipulation surface where the micro particles are manipulated is not oriented horizontally to the ground of the whole system. It is almost impossible to focus on the micro particles with constant inclination since sharp and blurred regions remain constant over the entire measuring range of the normalized variance cost function. However, when the inclination angle is sufficiently small this problem can be circumvented by focusing on a region of interest only.

In this section, a passive autofocus algorithm is introduced for MAW under a stereoscopic microscope. In the case of MAW applications, at least two different focal planes can occur in the workspace of the microscope. By analyzing the local maxima of the normalized variance values in the workspace of the microscope,  $z$ -axis locations of the micro particles and the manipulation tool are estimated easily. The manipulation tool has to be moved to the level of the micro particles along  $z$ -axis in order to start MAW micromanipulation or microassembly tasks.

## 5.6 Intrinsic Image Estimation & Image Deblurring

It is mentioned in many sections of this thesis that the most significant equipment of micro scale vision systems is the microscopes. However, in general the problem of the microscopes are that the illumination source must be powerful, on the other

hand it is needed to be scattered smoothly uniform to provide desired micro/nano scale reflectivity at desired magnifications. Additionally, illumination source must be focussed to the area of interest perfectly. In practical systems with reflected illumination technique, it is not possible to provide all requirements at the same time. Focusing errors (with the combination of other effects) causes blurring. If the illumination source is not powerful enough, measurement noise will be dominant in the image. Nevertheless, if the illumination source is powerful and does not illuminate every place in the scene smoothly uniform, it causes measured data to be the product of reflectivity and illumination images.

It is very important to separate the effects of reflectivity and illumination for these application and for general machine vision problems because both exist in almost all images and many tasks require to evaluate them separately. For example, the illumination effect should be detached from original image for segmentation implementations. Illumination and reflectivity images are named as intrinsic images [62]. It is a difficult problem to estimate intrinsic images because it is indeed an ill-posed problem. Tappen et al. in [63] proposed to find intrinsic images from a single image by assuming every spatial derivative in the scene caused by either illumination or reflectivity. Weiss in [64] used many images in which reflectivity was fixed, yet illumination was changing to estimate intrinsic images. He achieved to predict the intrinsic images by using statistical features of the illumination.

Because estimated reflectivity is still blurred and noisy, it is not enough just to estimate intrinsic images captured from the stereoscopic microscope under reflected illumination technique. The regularization is common method to handle these problems. The most common regularization method is Tikhonov Regularization [65]. Another regularization method which tries to solve some problems of Tikhonov regularization is named as Edge-Preserving Regularization [66].

In this section, a method for finding the reflectivity image from the measured stereo microscopic data is proposed. First, the two intrinsic images, reflectivity and illumination, are found from number of images. However the reflectivity component here still contains blurring and measurement noise. Because of this, Tikhonov and Edge-Preserving regularization algorithms are used to deal with these problems, and

to estimate the underlying real reflectivity image. The performance of techniques are demonstrated through experimental results on real data.

The motivation behind this work was: The vision system designed for MAW has two different camera views for micromanipulation and microassembly task to be performed as coarse & fine camera views mentioned in Section 4.2.1. Coarse camera view has low magnification therefore it has low resolution. However, on the other hand, it has high quality images. At the same time, fine camera view has approximately  $\sim 6$  times higher magnification, and higher resolution, but because of effect of illumination it has blurred images. (See Figure 5.23) The main aim of

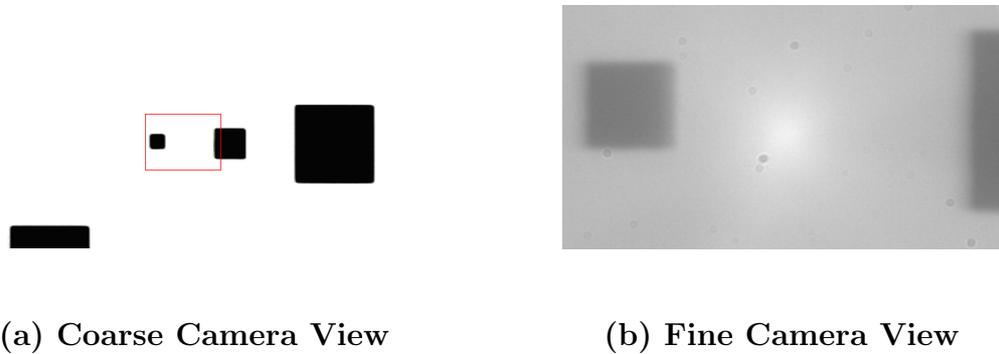


Figure 5.23: Micro Pattern Images from Coarse & Fine Cameras

the work is to obtain a reflectivity image which has no effect of illumination and has sharper edges for the square(s) inside of the sample image. The reason of performing the proposed methods is to be able to use the estimated reflectivity image in object recognition applications. Weiss's method mentioned in [64] is used to estimate the illumination and reflectivity images, and then the real reflectivity image is calculated by using Tikhonov and Edge-Preserving regularization methods.

### 5.6.1 Intrinsic Image Estimation

Recovery of intrinsic images from a single observed image is very difficult problem for computer vision systems. This problem is a classic ill-posed problem: Number of unknowns is two times larger than number of equations. Relationship between intrinsic images and observed image is as follows:

$$I(x, y) = R(x, y) \times L(x, y) \quad (5.19)$$

In equation 5.19,  $I(x,y)$  indicates observed image,  $R(x,y)$  indicates reflectivity image, and  $L(x,y)$  indicates illumination images as shown in Figure 5.24. Weiss proposes to

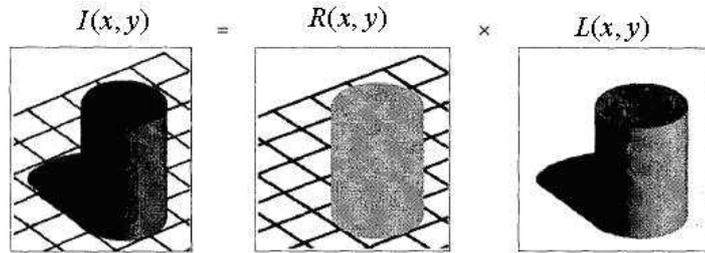


Figure 5.24: Decomposition of Intrinsic Images from [64]

estimate the reflectivity image by using many images in which reflectivity is fixed, illumination is changing. In the experiments, 40 different microscopic images are used as shown in Figure 5.25. In these images, the location of illumination source is fixed and its intensity is changing. The square in the scenes has  $20\mu m$  edges.

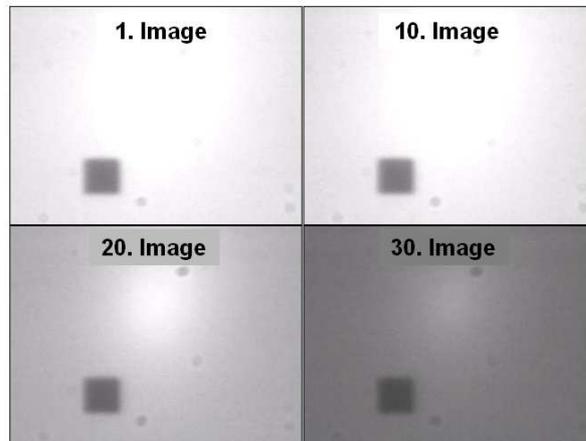


Figure 5.25: Sample Microscopic Images (Fixed Reflectivity & Distinct Illumination)

Algorithm uses pixel-wise median of each horizontal and vertical gradients of observed images with different illumination to calculate reflectivity image. The reason why it uses median is because histograms of the gradients of microscopic images tend

to be sparse as it is similar for nature images. These histograms can be modeled by Laplace distribution. The reflectivity can be found by maximum likelihood theorem if Laplacian model is used. Maximizing the likelihood is equivalent to minimizing the sum of absolute deviations in the gradient images. The sum of absolute values is minimized by the median. Calculation of the horizontal and vertical gradient images is given by equation 5.20:

$$\begin{aligned} F_{hor} &= h_{hor}I(x, y)_{vector} \\ F_{ver} &= h_{ver}I(x, y)_{vector} \end{aligned} \quad (5.20)$$

where  $h_{hor/ver}$  are horizontal and vertical gradient operators. In Figure 5.26, some horizontal and vertical gradient images are shown.

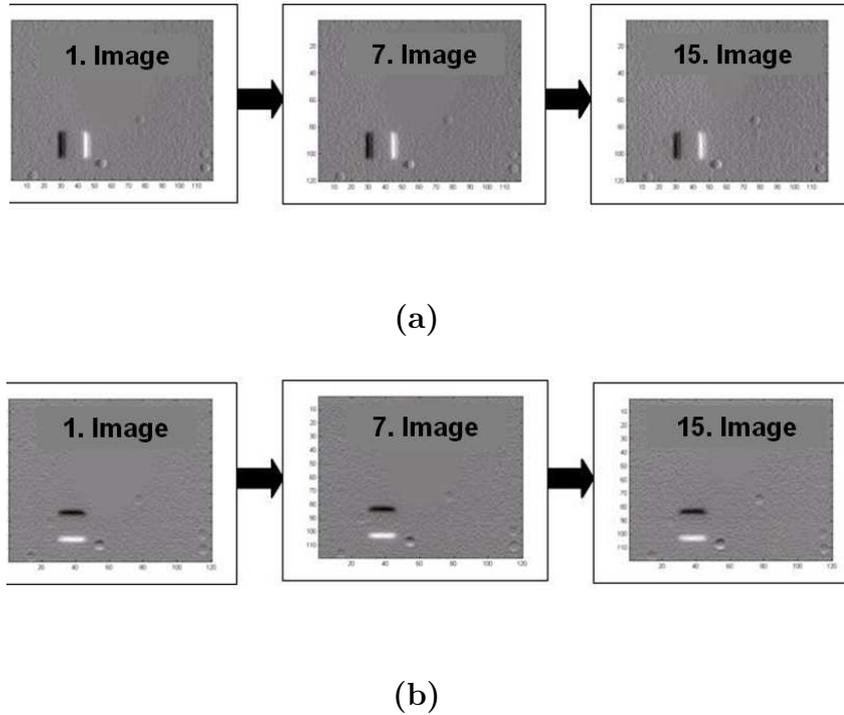


Figure 5.26: Sample Gradient Images: (a)Horizontal Gradient Images (b) Vertical Gradient Images

Median images is calculated from the gradient images as in equation 5.21.

$$\begin{aligned} \hat{r}_{hor} &= median_t F_{hor}(x, y, t) \\ \hat{r}_{ver} &= median_t F_{ver}(x, y, t) \end{aligned} \quad (5.21)$$

Here “ $t$ ” implies the index of observed images, “ $F_{hor/ver}$ ” indicates horizontal and vertical gradients of these images. We know that:

$$\hat{r}_{hor/ver} = h_{hor/ver} \hat{R}(x, y)_{vector} \quad (5.22)$$

At this point, reflectivity image, “ $\hat{R}(x, y)$ ”, is calculated by pseudo inverse operation. When above steps are applied to 40 different microscopic images, the following results are obtained. Estimated reflectivity image is shown in Figure 5.27, while Figure 5.28 displays some illumination images calculated using estimated reflectivity image.

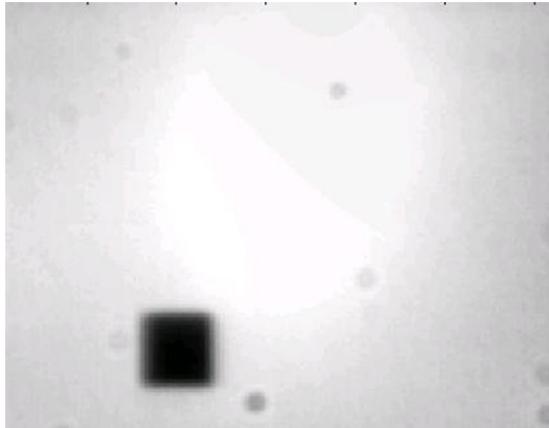


Figure 5.27: Estimated Reflectivity Image

### 5.6.2 Image Deblurring

The intrinsic images are calculated from observed images by the operations explained in previous section. However estimated reflectivity image is still blurred and has observation noise in itself. Two method is introduced here to get rid of these undesirable effects.

First method is well-known regularization method named as Tikhonov regularization described in [65]. A cost function in which takes estimated reflectivity image as input, is needed to be solved:

$$f_{Tik}(\lambda) = \arg \min \{ \|\hat{R}(x, y) - Hf\|_2^2 + \lambda \|\nabla f\|_2^2 \} \quad (5.23)$$

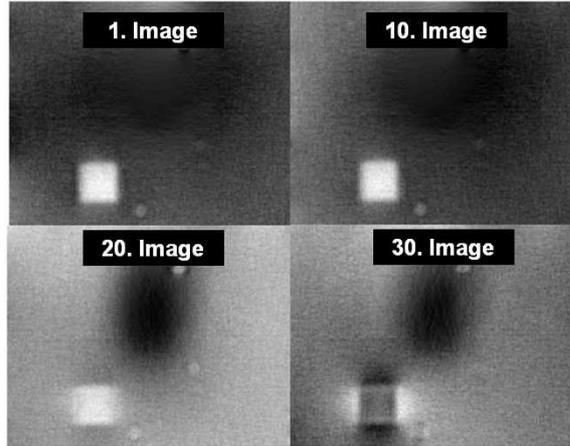


Figure 5.28: Sample Illumination Images calculated by Estimated Reflectivity Image

In the cost function, " $\hat{R}(x, y)$ " represents estimated reflectivity image, " $H$ " represents blurring operator, " $f$ " represents the real image that is target image, " $\lambda$ " is regularization coefficient, and " $\nabla f$ " are the gradients of the real image. The goal of this method is to deblur the image and at the same time to suppress the effect of noise. Because Tikhonov regularization has square cost function, it has closed form solution. The only unknown blurring operator " $H$ " is chosen from candidate operators. The operator whose parameters has the most fidelity to input data is selected.

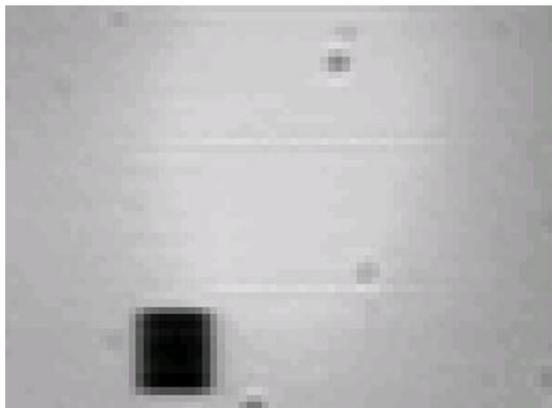


Figure 5.29: Resultant Image calculated by Tikhonov Regularization

If regularization coefficient “ $\lambda$ ” has small values, Tikhonov solution looks like generalized solution. If it has large values, it stresses on gradients and make the whole image smoother. In the start image, edges of the square become more blurred. having the previous information as a priori knowledge Tikhonov coefficient is selected as  $\lambda = 0.001$ . In Figure 5.29 the resultant image is presented. Comparing to estimated reflectivity image, it can be claimed that blurring effect decreased dramatically.

While Tikhonov regularization tries to suppress the noise in the image, unfortunately, it can suppress the edges of objects. The second method is another algorithm which tries to solve this problem of Tikhonov regularization. This method is called as Edge-Preserving regularization. Edge-Preserving regularization uses a cost function too as Tikhonov regularization which is depicted by:

$$f_{Edge-Preserving}(\lambda) = \arg \min \{ \|\hat{R}(x, y) - Hf\|_2^2 + \lambda \|\nabla f\|_p^p \} \quad (5.24)$$

Edge-Preserving is provided by introducing  $L_p$  norms where  $p < 2$  instead of using

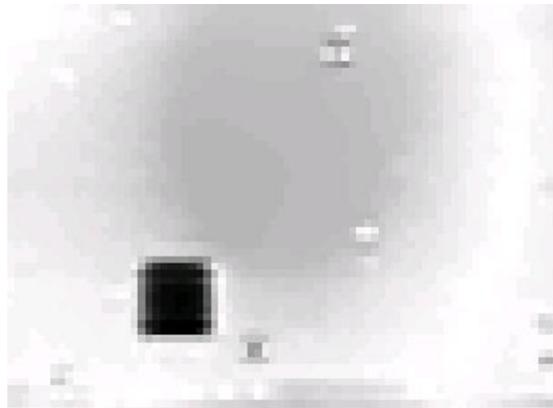


Figure 5.30: Resultant Image calculated by Edge-Preserving Regularization

$p = 2$  as in Tikhonov Regularization. In the experiments  $p = 1$  is used. If  $L_1$  is used in Edge-Preserving regularization, it is called “Total Variation” in [66]. There is no closed form solution for optimization problem in equation 5.24 and iterative algorithm is used to reach the solution. The resultant image estimated by Edge-Preserving regularization is demonstrated in Figure 5.30.

### 5.6.3 Discussions

In this section, a method is introduced to estimate the real reflectivity image captured from high magnification stereo microscope by elimination of illumination effect, blurring and noise [67].

First target was to clean images in which illumination is changing, but reflectivity is fixed for illumination effect. Second target was to eliminate blurring effect on reflectivity image and to sharpen the edges of  $20\mu m$  square in the scene and to make them clear to see. Two different regularization method has been used to achieve the targets. First one was Tikhonov regularization. Second one was Edge-Preserving regularization which aims to protect existed sharpness on the edges and to sharpen edges more. However, this algorithm can make areas which can have different contrasts and have no edges in real more clear. This is an undesired behavior. In our case, Edge-Preserving regularization gives good results when considering the edges of the square. Nonetheless, it the area which has intensive illumination partially clear. On the other hand, Tikhonov regularization is not successful enough to sharpen edges of the square, but it achieves to eliminate illumination effect in whole image.

In this section, the aim was to have clean and better image from blurred and noisy observed image. The edges of the square in the scene was forced to have sharpen edges. This is because it helps to extract the edges to use for object recognition part. Keeping these in mind, Edge-Preserving regularization gives better results compared to Tikhonov regularization.

It is possible to improve the results that are obtained from above processes. First of all, Weiss's algorithms does not have to be the best algorithm for this case. Secondly, reflectivity and location of the illumination source were fixed and illumination intensity was changing in our observed images. More images can be captured by creating new environment like changing the location of the illumination source additionally. Finally, blurring operator was selected from specific parametric models. More general model can be created so that better images can be obtained.

However, it is not possible to use these methods to get better images in real-time. These operations are static operations, and cannot be used in real-time system. This

is the first reason why these algorithms were not used in MAW software. Second reason is that better microscopic images are obtained by changing the illumination technique from reflected illumination to backlighting.

## Chapter 6

### Conclusions

In this thesis, the initial work on the design and implementation of a vision system for microassembly workstation is introduced. Both software and hardware issues are considered. Emphasize is put on the implementation of computer vision algorithms and vision based control techniques which help to build strong basis for the vision part of MAW. The main goal of designing such a vision system is to perform automated micromanipulation and microassembly tasks for a variety of applications. Experiments with some teleoperated and semi-automated tasks show quite promising results.

For teleoperated tasks which have static look & move structure vision based control, the operator controls the motion and manipulation units manually by the help of vision system. A micro particle is either picked and placed by a microgripper or is pushed by a probe to a target position. In this case, visual feedback is not presented in the control loop of MAW.

For semi-automated tasks, the operator only decides which particle will be manipulated to which target position by means selecting the particle and the target position on the screen. The probe manipulates the micro particles automatically by the aid of visual feedback provided by vision system. Semi-automated tasks can be considered as examples of Dynamic Look & Move structure vision based control.

On the other hand, basic computer vision and digital image processing algorithms, which are essential to build basis for teleoperated and semi-automated tasks, are implemented such as camera calibration for mapping between 2-D image plane coordinate space and 3-D world coordinate space, hough transform and EFDs for

image feature extraction and object recognition.

Additionally, autofocusing, which is necessary to estimate vertical distance and existence focal planes, motion parameter estimation to synchronize a moving micro particle with a manipulation tool, and intrinsic image estimation and image deblurring for removing noisy and blurring effect of reflected illumination were other subjects examined in this thesis.

As future research, fully automated micromanipulation and microassembly tasks can be implemented by changing vision based control technique to direct visual servoing. In addition to this, since EFDs give good results, automated micromanipulation of more complex micro particles or micro structures like cells or micro materials is also possible.

# Bibliography

- [1] Vikramaditya, B.; Nelson, B.J.; “Visually Guided Microassembly using Optical Microscopes and Active Vision Techniques”, *Proceedings of IEEE International Conference on Robotics and Automation*, Volume 4, 20-25 April 1997, Page(s):3172 - 3177
- [2] Vikramaditya, B.; Lord, J.G.; Nelson, B.J.; “Visually Servoed Micropositioning for Assembly of Hybrid MEMS: Theory and Experiments”, *Proceedings of the 36th IEEE Conference on Decision and Control*, Volume 1, 10-12 December 1997, Page(s):96 - 101
- [3] Zhao, W.; Zhang, W.J.; Gupta, M.M.; Zong, G.H.; Ouyang, P.R.; “A Micro Visual Servo System for Biological Cell Manipulation: Overview and New Developments”, *7th International Conference on Control, Automation, Robotics and Vision*, 2-5 December 2002, Page(s):429 - 434
- [4] Sun, Y.; Nelson, B.J.; “Autonomous Injection of Biological Cells using Visual Servoing”, *Experimental Robotics VII*, eds. D. Rus and S. Singh, Springer-Verlag London Ltd., 2001, Page(s):169 - 178
- [5] Inoue, K.; Arai, T.; Tanikawa, T.; Ohba, K.; “Dexterous Micromanipulation Supporting Cell and Tissue Engineering”, *IEEE International Symposium on Micro-NanoMechatronics and Human Science*, 7-9 November 2005, Page(s):197 - 202
- [6] Mukundakrishnan, B.; “Design for Visually servoed Microassembly”, Technical Report, University of Minnesota, February 2000

- [7] Vikramaditya, B.; “Micropositioning using Active Vision Techniques”, *Master of Science Thesis*, University of Illinois at Chicago, 1997
- [8] Ge Yang; Gaines, J.A.; Nelson, B.J.; “Optomechatronic Design of Microassembly Systems for Manufacturing Hybrid Microsystems”, *IEEE Transactions on Industrial Electronics*, Volume 52, Issue 4, August 2005, Page(s):1013 - 1023
- [9] Greminger, M.A.; Ge Yang; Nelson, B.J.; ‘ ‘Sensing Nanonewton Level Forces by Visually Tracking Structural Deformations”, *Proceedings of IEEE International Conference on Robotics and Automation*, Volume 2, 11-15 May 2002, Page(s):1943 - 1948
- [10] Greminger, M.A.; Nelson, B.J.; “Vision-Based Force Measurement”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 26, Issue 3, March 2004, Page(s):290 - 298
- [11] Trucco, E.; Verri, A.; “Introductory Techniques for 3-D Computer Vision”, Prentice Hall, 1998
- [12] Jain, R.; Kasturi, R.; Schunck, B.G.; “Machine Vision”, McGraw-Hill, 1995
- [13] Allegro, S.; “Automatic Microassembly by Means of Visually Guided Micro-manipulation”, Doctorate Thesis, Ecole Polytechnique Federale de Lausanne, 1998
- [14] Sethian, J.A.; “Level Set Methods”, Cambridge University Press, 1996
- [15] Kasaya, T.; Miyazaki, H.; Saito, S.; Sato, T.; “Micro Object Handling under SEM by Vision-Based Automatic Control”, *Proceedings of IEEE International Conference on Robotics and Automation*, Volume 3, 10-15 May 1999, Page(s):2189 - 2196
- [16] Fatikow, S.; Seyfried, J.; Fahlbusch, S.; Burkle, A.; Schmoeckel, F.; “A Flexible Microrobot-Based Microassembly Station”, *Proceedings of 7th IEEE International Conference on Emerging Technologies and Factory Automation*, Volume 1, 18-21 October 1999, Page(s):397 - 406

- [17] Ogawa, N.; Oku, H.; Hashimoto, K.; Ishikawa, M.; “Microrobotic Visual Control of Motile Cells Using High-Speed Tracking System”, *IEEE Transactions on Robotics* [see also *IEEE Transactions on Robotics and Automation*], Volume 21, Issue 4, August 2005, Page(s):704 - 712
- [18] Seok Joo Lee; Kyunghwan Kim; Deok-Ho Kim; Jong-Oh Park; Gwi Tae Park; “Recognizing and Tracking of 3D-Shaped Micro Parts Using Multiple Visions for Micromanipulation”, *Proceedings of 2001 International Symposium on Micromechanics and Human Science*, 9-12 September 2001, Page(s):203 - 210
- [19] Xudong Li; Guanghua Zong; Shusheng Bi; “Development of Global Vision System for Biological Automatic Micro-Manipulation System”, *Proceedings of IEEE International Conference on Robotics and Automation*, Volume 1, 2001, Page(s):127 - 132
- [20] Arai, F.; Kawaji, A.; Luangjarmekom, P.; Fukuda, T.; Itoigawa, K.; “Three-Dimensional Bio-Micromanipulation under the Microscope”, *Proceedings of IEEE International Conference on Robotics and Automation*, Volume 1, 2001, Page(s):604 - 609
- [21] Burkle, A.; Fatikow, S.; “Laser Measuring System for a Flexible Microrobot-Based Micromanipulation Station”, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Volume 1, 31 October-5 November 2000, Page(s):799 - 804
- [22] Burkle, A.; Fatikow, S.; “Laser-based Depth Recovery for Fine Positioning of a Microrobot”, *Symposium on Microrobotics at the 2000 ASME Int. Mechanical Engineering Congress and Exposition*, 2000
- [23] Morris, T.; “Computer Vision and Image Processing”, Palgrave Macmillan, 2004
- [24] Shah, M.; “Fundamentals of Computer Vision”, Florida, 7 December 1997
- [25] Canny, J.; “Finding Edges and Lines in Images”, *Master of Science Thesis*, MIT, 1993

- [26] Canny, J.; “A Computational Approach to Edge Detection”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 8, Page(s):679 - 698
- [27] Duda, R.O.;Hart, P.E.; “Use of the Hough Transformation to Detect Lines and Curves in Pictures”, *Communications of Association of Computing Machinery*, Volume 15, January 1972, Page(s):1 - 15
- [28] Ballard, D.H.; “Generalizing the Hough transform to Detect Arbitrary Shapes”, *Pattern Recognition*, Volume 2, Issue 13, April 1981, Page(s)=111 - 122
- [29] Matas, J.; Galambos, C.; Kittler, J.; “Progressive Probabilistic Hough Transform”, *British Machine Vision Conference*, 1998 Page(s):255 - 274
- [30] Yuen, H.K.; Princen, J.; Illingworth, J.; Kittler, J.; “Comparative Study of Hough Transform Methods for Circle Finding”, *Journal of Image and Vision Computing*, Volume 8, 1990, Page(s):71 - 77
- [31] Kuhl, F.P.; Giardina, C.R.; “Elliptic Fourier Features of a Closed Contour”, *Computer Graphics and Image Processing*, Volume 18, 1982, Page(s):236 - 258
- [32] Abidi, M.A.; Gonzalez, R.C.; “Shape Decomposition using Elliptic Fourier Descriptors”, *Proceedings of 18th IEEE Southeast Symposium of Systems Theory*, 1986, Page(s):53 - 61
- [33] Lin, C.S.; Hwang, C.L.; “New Forms of Shape Invariants from Elliptic Fourier Descriptors”, *Pattern Recognition*, Volume 20, No. 5, 1987, Page(s):535 - 545
- [34] Granlund, G.H.; “Fourier Preprocessing for Hand Print Character Recognition”, *IEEE Transactions on Computers*, Volume C-21, February 1972, Page(s):195 - 201
- [35] Ullman, S.; “The interpretation of visual motion”, Cambridge, MA : MIT Press, 1979

- [36] Roach, J.W.; Aggarwal, J.K.; “Determining The Movement of Objects from a Sequence of Images”, *IEEE Transactions on Pattern Analysis and Machine Intelligence* , Volume 6, 1980, Page(s):554 - 562
- [37] Calway A.; “Recursive Estimation of 3D Motion and Surface Structure from Local Affine Flow Parameters”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 27, No. 4, April 2005 Page(s):562 - 574
- [38] Black, M.J.; Jepson, A.; “Estimating Optical Flow in Segmented Images Using Variable-Order Parametric Models with Local Deformations”, *IEEE Transactions on Pattern Analysis and Machine Intelligence* , No. 10, October 1996, Page(s):972 - 985
- [39] Utkin, V.; “Sliding Modes in Control Optimization”, Springer-Verlag, 1992
- [40] Abidi, K.; Sabanovic, A.; “A Study on Discrete Sliding Mode Control: Robustness Analysis and Experimental Investigation”, *Proceedings of IASTED International Conference on Modelling, Identification and Control (MIC 2005)* , Innsbruck, Austria, February 16-18, 2005
- [41] Allegro, S.; Chanel, C.; Jacot, J.; “Autofocus for Automated Microassembly Under a Microscope”, *International Conference on Image Processing Proceedings* , Volume 1, 16-19 September 1996, Page(s):677 - 680
- [42] Groen, F.C.A.; Young, I.T.; Lighthart, G.; “A Comparison of Different Focus Functions for Use in Autofocus Algorithms”, *Journal of Cytometry* , 1985, Page(s):81 - 91
- [43] Firestone, L.; Cook, K.; Culp, K.; Talsania, N.; Preston, K.; “Comparison of Autofocus Methods for Automated Microscopy”, *Journal of Cytometry* , 1991, Page(s):195 - 206
- [44] Sun, Y.; Duthaler, S.; Nelson, B.J.; “Autofocusing in Computer Microscopy: Selecting the optimal Focus Algorithm”, *Journal of Microscopy Research and Technology* , 2004, Page(s):139 - 149

- [45] Tang, X.; LHostis, P.; Xiao, Y.; “An Auto-Focusing Method in a Microscopic Testbed for Optical Discs”, *Journal of Research of the National Institute of Standards and Technology*, Volume 105, Number 4, July/August 2000, Page(s):565 - 569
- [46] Suzuki, A.; Arai, T.; Mae, Y.; Inoue, K.; Tanikawa, T.; “Automated Micro Handling”, *Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 16-20 July 2003, Page(s):348 - 353
- [47] Haliyo, D.S.; Venture, G.; Regnier, S.; Guinot, J.; “An Overview of the Micro-Manipulation System [mu] MAD”, *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2005, Page(s):390 - 395
- [48] Sulzmann, A.; Breguet, J.M.; Jacot, J.; “Micromotor Assembly using High Accurate Optical Vision Feedback for Microrobot Relative 3D Displacement in Submicron Range”, *International Conference on Solid State Sensors and Actuators* Volume 1, 16-19 June 1997, Page(s):279 - 282
- [49] Parvin, B.; Callahan, D.E.; Johnston, W.; Maestre, M., “Visual Servoing for Micro-Manipulation”, *Proceedings of the 13th International Conference on Pattern Recognition*, Volume 3, 25-29 August 1996, Page(s):341 - 345
- [50] Hill, J.; Park, W.T.; “Real Time Control of a Robot with a Mobile Camera”, *9th International Symposium on Industrial Robots*, March 1979, Page(s):233 - 246
- [51] Corke P.; “Visual Control of Robot Manipulators – A Review”, Technical Report, *CSIRO Division of Manufacturing Technology*, 1992
- [52] Corke, P.; “High - Performance Visual Closed - Loop Robot Control”, *Doctor of Philosophy Dissertation*, University of Melbourne, July 1994
- [53] Hutchinson, S.; Hager, G.D.; Corke, P.I.; “A Tutorial on Visual Servo Control”, *IEEE Transactions on Robotics and Automation*, Volume 12, Issue 5, October 1996, Page(s):651 - 670

- [54] Kragic, D.; Christensen, H.I.; “Survey on Visual Servoing for Manipulation”, Stockholm, Sweden
- [55] Sanderson, A.C.; Weiss, L.E.; “Image-based Visual Servo Control using Relational Graph Error Signals”, *Proceedings of IEEE International Conference on Cybernetics and Society*, 1980, Page(s)=1074 - 1077
- [56] Chaumette, F.; Rives, P.; Espiau, B.; “Positioning of a Robot with Respect to an Object, Tracking it and Estimating its Velocity by Visual Servoing”, *IEEE International Conference on Robotics and Automation Proceedings*, Volume 3, 9-11 April 1991, Page(s):2248 - 2253
- [57] Tsai, R.; “A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology using Off-the-Shelf TV Cameras and Lenses”, *IEEE Journal of Robotics and Automation*, Volume 3, Issue 4, August 1987, Page(s):323 - 344
- [58] Lenz, R.K.; Tsai, R.Y.; “Techniques for Calibration of the Scale Factor and Image Center for High Accuracy 3-D Machine Vision Metrology”, *Pattern IEEE Transactions on Analysis and Machine Intelligence*, Volume 10, Issue 5, September 1988, Page(s):713 - 720
- [59] Zhang, Z.; “A Flexible New Technique for Camera Calibration”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 22, Issue 11, November 2000, Page(s):1330 - 1334
- [60] Yilmaz, B.; Unel, M.; Sabanovic. A.; “Rigid and Affine Motion Estimation in Vision Using Sliding Mode Observers”, *Proceedings of the IEEE International Symposium on Industrial Electronics*, Volume 1, 20-23 June 2005, Page(s):31 - 36
- [61] Dogan, E.; Yilmaz, B.; Unel, M.; Sabanovic. A.; “A Sliding Mode Approach to Visual Motion Estimation”, *IEEE 9th International Workshop on Advanced Motion Control*, Volume 2, 27-29 March 2006 Page(s):733 - 738

- [62] Barrow, H.G.; Tenebaum, J.M.; “Recovering Intrinsic Scene Characteristics from Images”, In A. Hanson and E. Riseman, editors, *Computer Vision Systems*. Academic Press, 1978
- [63] Tappen, M.; Freeman, W.T.; Adelson, E.H; “Recovering Intrinsic Images from a Single Image”, *IEEE Transactions on Pattern Analysis and Machine Intelligence* , Volume 27, Issue 9, September 2005, Page(s):1459 - 1472
- [64] Weiss, Y.; “Deriving intrinsic images from image sequences”, *Eighth IEEE International Conference on Computer Vision Proceedings* , Volume 2, 7-14 July 2001, Page(s):68 - 75
- [65] Tikhonov A.N.; “The Regularization of Ill-Posed Problems”, *Dokl. Akad. Nauk.(SSR 153)*, No. 1, 1963, Page(s):49 - 52
- [66] Vogel, C.R.; Oman, M.E.; “Fast, Robust Total Variation-Based Reconstruction of Noisy, Blurred Images”, *IEEE Transactions on Image Processing* , Volume 7, Issue 6, June 1998, Page(s):813 - 824
- [67] Dogan, E.; Cetin, M.; “Intrinsic Image Estimation and Image Deblurring for Microscopic Images”, *IEEE 14th Conference on Signal Processing and Communication Applications*, 17-19 April 2006