

# **POSE ESTIMATION FOR ROBOTIC PERCUSSIVE RIVETING**

by

Yu Lin

Master of Applied Science, Mechanical Engineering

Ryerson University, 2008

A dissertation

presented to Ryerson University

in partial fulfillment of the  
requirements for the degree of

Doctor of Philosophy

in the Program of  
Aerospace Engineering

Toronto, Ontario, Canada, 2014

© Yu Lin, 2014

## **AUTHOR'S DECLARATION**

I hereby declare that I am the sole author of this dissertation. This is a true copy of the dissertation, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this dissertation to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this dissertation by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

# ABSTRACT

## Pose Estimation for Robotic Percussive Riveting

Yu Lin

A dissertation for the degree of  
Doctor of Philosophy, 2014

Department of Aerospace Engineering, Ryerson University

Recently, a robotic percussive riveting system has been developed at Ryerson University for an automation of percussive riveting process of aero-structural fastening assembly. The system consists of a robot holding a percussive riveting gun equipped with a rivet feeder, a gantry holding a working panel of aero-structure, and a position visual sensor. Prior to riveting, the robot is required to first position and then insert a rivet precisely into a hole on the panel without engaging the panel to prevent potential damage. The underlying challenges to precise insertion are various sources of system uncertainties, mainly including alignment errors among coordinate systems of the robot, panel and sensor, and relatively poor absolute positioning accuracy of the robot due to mechanical deflection, assembly clearance, and machining tolerance. For this reason, the research of relative pose estimation between the robot and panel has been carried out pertaining to these challenges.

Essentially, pose estimation is proposed for robotic percussive riveting, which estimates the relative pose between two rigid bodies based on noisy visual measurements of point features on rigid bodies. Three categories can be classified, namely, static, dynamic, and robust pose estimation. Firstly, static pose estimation is the parameter estimation of static relative pose transformations among a number of frames, which solves the issue of alignment errors. Direct solutions of static relative pose estimation are derived based on least-square methods. Secondly, to tackle the issue of poor absolute positioning accuracy of the robot, dynamic relative pose estimation is proposed addressing a state estimation of relative poses during motion. Iterative extended Kalman filter method is adapted for the state estimation. Thirdly, for robustness against outliers of

point measurements, robust pose estimation is proposed based on an outlier diagnosis using the technique of relaxation of rigid body constraints. Indeed, outlier diagnosis is a pre-processing of point measurements, in which outliers are detected and removed prior to the relative pose estimation. Further, a decorrelation method is proposed for measurement calibration using multivariate statistical analysis to find an optimal sensor-to-target configuration. As a result, each coordinate measurement is close to uncorrelated and it allows for a simple calibration.

# ACKNOWLEDGEMENTS

I am deeply indebted to my supervisor Dr. Fengfeng (Jeff) Xi, to whom I owe a debt of gratitude for the invaluable support, guidance and kindly encouragement throughout the past years of study. With his strong academic background and innovative ideas, he has been really helpful to me for solving many difficulties whenever I needed help. Thanks also go to my co-supervisor Dr. Vincent Chan, for his great support and sincere help during my graduate studies. Also, I would like to thank Post-Doctoral fellow Dr. Yimin Zhao for his time and all the discussion we had.

Gratitude also goes to Ryerson University, National Science and Engineering Research Council (NSERC), FedDev Ontario, and Aerospace Manufacturing Technology Centre (AMTC) of National Research Council Canada (NRC) for providing adequate research funding and opportunities for my graduate studies.

I would like to sincerely thank my family, especially my beloved fiancée, Christine Xiaofang Weng, for her encouragement, support and understanding over the school years.

Last but not least, thank you, Lord, Heavenly Father, for bringing me through all the difficulties of this research. Thank you for listening to my prayers and brightening my world with "the way, the true, and the life" (John 14:6). Thanks also go to my fellow brothers and sisters in Christ, for their continual encouragement.

# TABLE OF CONTENTS

<b>AUTHOR'S DECLARATION .....</b>	<b>ii</b>
<b>ABSTRACT.....</b>	<b>iii</b>
<b>ACKNOWLEDGEMENTS .....</b>	<b>v</b>
<b>TABLE OF CONTENTS .....</b>	<b>vi</b>
<b>LIST OF TABLES .....</b>	<b>xi</b>
<b>LIST OF FIGURES .....</b>	<b>xii</b>
<b>NOMENCLATURE.....</b>	<b>xv</b>
<b>CHAPTER 1 INTRODUCTION.....</b>	<b>1</b>
<b>1.1 Background.....</b>	<b>1</b>
<b>1.2 Robotic Percussive Riveting .....</b>	<b>4</b>
<b>1.3 Problem Formulation.....</b>	<b>7</b>
<b>1.4 Research Objective.....</b>	<b>8</b>
<b>1.5 Outline of Thesis.....</b>	<b>11</b>
<b>CHAPTER 2 LITERATURE REVIEW .....</b>	<b>13</b>
<b>2.1 Pose Estimation.....</b>	<b>13</b>
2.2.1 Static Pose Estimation.....	14
2.2.2 Dynamic Pose Estimation .....	15
2.2.3 Robust Pose Estimation .....	16
<b>2.2 Position-Based Visual Servoing.....</b>	<b>18</b>
<b>CHAPTER 3 STATIC POSE ESTIMATION.....</b>	<b>22</b>
<b>3.1 Formulation of Pose Estimation.....</b>	<b>22</b>
3.1.1 Cartesian Frame Formulation.....	23

3.1.2 Three-Point Method .....	23
3.1.3 Normalized Directional Vectors .....	24
3.1.4 Covariance Matrix Method .....	25
<b>3.2 Least Squares Methods .....</b>	<b>26</b>
<b>3.3 Direct Relative Pose Estimation between Two Rigid Bodies.....</b>	<b>27</b>
3.3.1 Motivation of Direct Relative Pose Estimation .....	28
3.3.2 Problem Statement .....	29
3.3.3 LS solution .....	32
3.3.4 CTLS solution .....	36
3.3.5 Case Studies .....	38
3.3.6 Summary .....	43
<b>3.4 Pose Estimation for Multiple Frames of a Rigid-Body System.....</b>	<b>45</b>
3.4.1 Description of Localization.....	45
3.4.3 Localization of Robotic Riveting System .....	46
3.4.4 Verification of Localization .....	55
3.4.5 Summary .....	56
<b>CHAPTER 4 DYNAMIC POSE ESTIMATION.....</b>	<b>57</b>
<b>4.1 Description of Dynamic Pose Estimation .....</b>	<b>57</b>
<b>4.2 Problem Statement .....</b>	<b>59</b>
<b>4.3 Relative Pose and Motion Estimation.....</b>	<b>62</b>
4.3.1 State Space Modeling.....	63
4.3.2 Relative Pose and Motion Estimation by IEKF .....	65
<b>4.4 Simulations and Experiments.....</b>	<b>68</b>

4.4.1 Preliminary Transformation of Target 2-TCP and Target 1-Working Panel	70
4.4.2 Simulations.....	71
4.4.3 Experiments.....	81
<b>4.5 Summary .....</b>	<b>84</b>
<b>CHAPTER 5 ROBUST POSE ESTIMATION .....</b>	<b>86</b>
<b>5.1 Problem Statement .....</b>	<b>86</b>
<b>5.2 Outlier Diagnosis .....</b>	<b>87</b>
5.2.1 Rigid Body Constraints.....	87
5.2.2 Outlier Diagnosis by Relaxation.....	89
<b>5.3 Simulations and Experiments.....</b>	<b>92</b>
5.3.1 Case Study of Four Outliers.....	93
5.3.2 Experiment Verification.....	96
<b>5.4 Summary .....</b>	<b>97</b>
<b>CHAPTER 6 DECORRELATION METHOD FOR MEASUREMENT CALIBRATION.....</b>	<b>99</b>
<b>6.1 Description of Measurement Calibration.....</b>	<b>99</b>
<b>6.2 Decorrelation Formulations .....</b>	<b>101</b>
6.2.1 Local Decorrelation.....	102
6.2.2 Global Decorrelation.....	104
<b>6.3 Evaluation and Comparison.....</b>	<b>107</b>
<b>6.4 Experiments and Simulations.....</b>	<b>107</b>
<b>6.5 Summary .....</b>	<b>114</b>
<b>CHAPTER 7 CONCLUSIONS AND FUTURE WORKS .....</b>	<b>115</b>

7.1 Conclusions .....	115
7.2 Contributions .....	116
7.3 Future Works.....	117
<b>APPENDIX A - POSITION AND ORIENTATION.....</b>	<b>118</b>
<b>A.1 Position Vector.....</b>	<b>118</b>
<b>A.2 Rotation Matrix .....</b>	<b>119</b>
<b>A.3 Angle-set Representation of a Rotation.....</b>	<b>121</b>
<b>APPENDIX B - GENERAL MOTION.....</b>	<b>124</b>
<b>B.1 General Motion of a Single Rigid Body .....</b>	<b>124</b>
<b>B.2 General Motion of Multiple Rigid Bodies .....</b>	<b>124</b>
<b>APPENDIX C - MATRIX COMPUTATIONS.....</b>	<b>128</b>
<b>C.1 The Rank of a Matrix.....</b>	<b>128</b>
<b>C.2 Singular Value Decomposition .....</b>	<b>129</b>
<b>C.3 The Moore-Penrose (MP) Inverse.....</b>	<b>129</b>
<b>C.4 The Solution of Linear Equation Systems.....</b>	<b>130</b>
<b>C.5 Least Squares Method and Gauss-Markov Theorem .....</b>	<b>131</b>
<b>C.6 Rotation of Subspaces (Constrained Least-Squares) .....</b>	<b>133</b>
<b>APPENDIX D - LEAST-SQUARE ESTIMATION OF TRANSFORMATION PARAMETERS BETWEEN TWO POINT PATTERNS.....</b>	<b>134</b>
<b>APPENDIX E - GENERAL FORMULATION FOR THE OUTLIER DIAGNOSIS BY RELAXATION.....</b>	<b>136</b>
<b>APPENDIX F - MEASURING SYSTEM MODELING .....</b>	<b>137</b>
<b>APPENDIX G - ROBOTIC PERCUSSIVE RIVETING .....</b>	<b>139</b>

<b>APPENDIX H - ROBUSTNESS EVALUATING EXPERIMENTS FOR NDI DATA</b>	
<b>ACQUISITION .....</b>	<b>146</b>
<b>H.1 Reflection Distraction .....</b>	<b>146</b>
<b>H.2 Illumination Distraction .....</b>	<b>147</b>
<b>H.3 Missing data and outliers.....</b>	<b>150</b>
<b>APPENDIX I - JIG DESIGN FOR PANEL .....</b>	<b>153</b>
<b>APPENDIX J - PUBLICATIONS AND PATENTS .....</b>	<b>154</b>
<b>REFERENCES.....</b>	<b>156</b>

# LIST OF TABLES

<b>Table 1.1:</b> Determination of Transformations .....	8
<b>Table 3.1:</b> Observed Coordinates of Points of Rigid Body Targets w.r.t. the Frame of the Measurement System .....	39
<b>Table 3.2:</b> Local Coordinates of Points of Rigid Body Targets.....	40
<b>Table 3.3:</b> Relative Rotation Matrix in Experiments ( $R_{12}$ ) .....	40
<b>Table 3.4:</b> Simulated Coordinates of Points w.r.t. the Measurement Frame .....	42
<b>Table 3.5:</b> Simulated Local Coordinates of Points w.r.t. the Body-Fixed Frames .....	42
<b>Table 3.6:</b> Relative Rotation Matrix in Simulation ( $R_{12}$ ).....	42
<b>Table 3.7:</b> Combination of point numbers.....	44
<b>Table 3.8:</b> Central coordinates of tooling balls in the sensor and panel frames. ....	49
<b>Table 3.9:</b> Coordinates of the TCP in the sensor and robot base frames. ....	53
<b>Table 3.10:</b> LED coordinates of the target in the sensor and local frames. ....	54
<b>Table 4.1:</b> Local coordinates of LED points of the targets in the body-fixed frames .....	69
<b>Table 4.2:</b> Local coordinates of LED points of the targets w.r.t. TCP and working panel	71
<b>Table 4.3:</b> D-H parameters of ABB 4400/45kg robot manipulator .....	77
<b>Table 5.1:</b> Pose Transformation Before and After the Outlier Contamination .....	94
<b>Table 5.2:</b> Indicators of Compatibility - Replacement of Distance Deviation with Integers .....	95
<b>Table 5.3:</b> Iterations of Merit Scores for Outlier Diagnosis by Relaxation (Four Outliers) .....	95
<b>Table 5.4:</b> Pose Transformation Before and After the Outlier Diagnosis.....	96
<b>Table 5.5:</b> Indicators of Compatibility for 7 Markers .....	97
<b>Table 5.6:</b> Iterations of Merit Scores for Outlier Diagnosis by Relaxation (Two Outliers) .....	97
<b>Table B.1:</b> $R_m$ and $b_m$ of different kinematic pairs [71]. ....	127

# LIST OF FIGURES

<b>Figure 1.1:</b> Principle of Robotic Percussive Riveting. [18] .....	5
<b>Figure 1.2:</b> (a) System configuration setup simulated in ABB RobotStudio. (b) Experimental setup of the robotic riveting system at EPH107, Ryerson University. ....	6
<b>Figure 1.3:</b> System transformational modeling with two loops of transformations. One between the panel, measurement, and robot base; while another one between the robotic tooling tip, end-effector, target, measurement, and robot base. ....	7
<b>Figure 1.4:</b> Structure of research of pose estimation - problems and methods. ....	9
<b>Figure 3.1:</b> Cartesian frame formulation. ....	23
<b>Figure 3.2:</b> Transformation determinations using the three-point position data [28]. ....	24
<b>Figure 3.3:</b> Two rigid body targets observed by a measurement system. ....	29
<b>Figure 3.4:</b> Experimental setup of a robotic riveting system. ....	38
<b>Figure 3.5:</b> Two rigid body targets on the work piece and the robot end-effector. ....	39
<b>Figure 3.6:</b> Three tooling balls. ....	48
<b>Figure 3.7:</b> Wireless handheld probe for digitization. ....	48
<b>Figure 3.8:</b> Panel localization using three tooling balls, and TCP of the rivet gun. ....	49
<b>Figure 3.9:</b> (a) Digitization of tooling balls with a handheld probe; (b) the fitted sphere. ....	49
<b>Figure 3.10:</b> Determination of tool length from TCP to end-effector flange, (a) digitizing the surface of the flange of the end-effector; (b) digitizing the TCP of the rivet gun. ....	51
<b>Figure 3.11:</b> (a) Screenshot of tool definition for the rivet gun in the ABB FlexPendant; (b) Pose of TCP of the rivet gun with respect to the robot base frame ( $Trtcp$ ). ....	51
<b>Figure 3.12:</b> Robot localization by creating a virtual target at multiple poses in workspace. ....	52
<b>Figure 3.13:</b> Target 2. (a) Modeling (b) Four infrared LEDs. (c) The target on the tooling and transformation between frames of target 2 and measurement. ....	55
<b>Figure 3.14:</b> Hole digitization with handheld probe. ....	55
<b>Figure 4.1:</b> Two rigid body targets observed by a position measurement system. ....	60
<b>Figure 4.2:</b> (a) Four LEDs on the rigid body target 1. (b) Two rigid body targets attached on the robot tooling and the working panel. ....	69
<b>Figure 4.3:</b> Relative pose estimation integrated in the position-based visual servoing of the robotic riveting system. ....	72
<b>Figure 4.4:</b> Simulink model of the robot position-based visual servoing using the proposed filter of the relative pose estimation designed in MatLab. ....	73
<b>Figure 4.5:</b> Position-based visual servoing in the Simulink model. ....	74
<b>Figure 4.6:</b> Plant of the robot in Simulink model. ....	75
<b>Figure 4.7:</b> The Simulink model of the optoelectronic CMM to simulate the point measurements. ....	77
<b>Figure 4.8:</b> Schematic diagrams of the robot poses in the simulation. (a) The initial position. (b) The desired position. ....	78
<b>Figure 4.9:</b> Simulation results of dynamic pose estimation. (a) Relative translational estimation results. (b) Relative orientation estimation results. The blue solid	

	lines represent the actual relative poses, and the red dashed lines represent the estimated poses.....	79
<b>Figure 4.10:</b>	Norm of the error of relative pose estimation during simulation. (a) Linear scale of norm of error values. (b) Log scale of norm of error values. ....	80
<b>Figure 4.11:</b>	Initial position of the robot with joint angles of $q^0 = [0\ 0\ 0\ 0\ 10\ 0]$ . ....	81
<b>Figure 4.12:</b>	Final position of the robot with joint angles of $q^d = [15^\circ\ -20^\circ\ 30^\circ\ 0\ 15^\circ]$ . ....	82
<b>Figure 4.13:</b>	Experimental linear trajectory comparison. (a) Relative translational estimation results. (b) Relative orientation estimation results. The blue solid lines are the nominal trajectory and the red dashed lines are the estimated trajectory.....	83
<b>Figure 4.14:</b>	Converging results of norms of the errors of relative pose estimation in the experiment. (a) Linear scale of norm of errors. (b) Log scale of norm of errors. ....	84
<b>Figure 5.1:</b>	Motion of the cubic rigid body. ....	87
<b>Figure 5.2:</b>	Point distances under the rigid body constraints. (a) one, or two, or three distance constraints are not sufficient for unique 3D positioning; (b) the minimum number of points is 5 for unique 3D positioning; (c) four distance constraints for point $p_i$ ; (d) an outlier example.....	89
<b>Figure 5.3:</b>	Experimental system set-up.....	92
<b>Figure 5.4:</b>	(a) 7 Markers in the 3D space; (b) target of the cubic rigid body.....	93
<b>Figure 5.5:</b>	Two poses of the cubic rigid body. ....	93
<b>Figure 5.6:</b>	Outliers occurred at markers 1, 2, 6, and 7; (a) Pose transformation determined with four outliers; (b) Pose estimation from the remaining three markers 3-5 after removing the outliers. ....	95
<b>Figure 5.7:</b>	(a) Merit scores during 7 iterations; (b) Signs obtained from normalized merit scores, “-1” indicate outliers (markers 1, 2, 6, and 7), and “+1” indicate the reliable data (markers 3, 4, and 5).....	96
<b>Figure 5.8:</b>	Markers for simulating outliers and the original markers. ....	97
<b>Figure 6.1:</b>	500 repetitive readings of marker_1s' constant 3D position at pose_1. ....	108
<b>Figure 6.2:</b>	Standard deviation (STD) of marker's positions. ....	109
<b>Figure 6.3:</b>	Mean absolute correlation coefficients of 5 poses for 7 markers. ....	109
<b>Figure 6.4:</b>	Comparison of ellipsoids of $1\sigma$ , $2\sigma$ , $3\sigma$ of the original (a) and the decorrelated data (b).....	110
<b>Figure 6.5:</b>	(a) Local optimal decorrelated configurations of the NDI Optotrak system (in red) with respect to the PDF ellipsoids of 7 markers, the original NDI configuration is in cyan; (b) mean and STD of measurements before and after the local decorrelation.....	111
<b>Figure 6.6:</b>	The global decorrelated configuration of NDI for the entire set of markers (in red), when the marker data $X_i$ are mutually uncorrelated. ....	112
<b>Figure 6.7:</b>	Results when the data of the 7 markers are mutually correlated, with a larger size of ellipsoid due to larger eigenvalues. ....	112
<b>Figure 6.8:</b>	Local and global data decorrelation comparisons, with various noisy levels from $\pm 10\%$ to $\pm 30\%$ . “Local_1” to “Local_7” mean the seven local data decorrelation methods at each markers. “Global_UC” represents the global data decorrelation when mutually uncorrelated; “Global_C” indicates the	

global data decorrelation when mutually correlated. ....	113
<b>Figure A.1:</b> Multiple modules system for a serial robot manipulator. ....	118
<b>Figure A.2:</b> Position vector $p$ [71]. ....	119
<b>Figure B.1:</b> General motion of a single module [71]. ....	124
<b>Figure B.2:</b> Vector method for a multi-module system [71] ....	125
<b>Figure F.1:</b> Principle of Optoelectronic Coordinate Measurement Machines [88]. ....	137
<b>Figure G.1:</b> Rivet guns and C-frame riveter. ....	140
<b>Figure G.2:</b> Robotic Percussive Riveting System [3]. ....	142
<b>Figure G.3:</b> Robot Riveted Samples [3]. ....	142
<b>Figure G.4:</b> Flowchart of Process of Robotic Percussive Riveting System. ....	143
<b>Figure G.5:</b> Graphical user interface development for robotic riveting system. ....	144
<b>Figure G.6:</b> Interfaces for acquisition and localization. ....	145
<b>Figure H.1:</b> IR marker's reflection on an aluminum board and a painted metal part. ...	147
<b>Figure H.2:</b> (a) The NDI Optotrak system observes the IR LED lights of the remote instead, when the markers are not observable due to away from the camera sensing direction. (b) Demonstration of Infrared LED remote's distraction during the data acquisition: the remote was pressed down and placed beside a marker (marker 7). ....	148
<b>Figure H.3:</b> The positions of 7 markers under the distraction of the remote: missing data at marker 7; disturbances at markers 3-6; no effect at markers 1 and 2. The blue, green and red lines represent the x, y and z coordinates. ....	149
<b>Figure H.4:</b> (a) Original set-ups with 7 markers attached to the cubic target; (b) a way of creating an outlying marker by detaching the marker from the mounted case. ....	151
<b>Figure H.5:</b> (a) With four outlying markers, the pose estimation failed even when there are three remaining markers; (b) once four outliers were manually removed, the pose estimation of the rigid body was restored immediately. ....	152

# NOMENCLATURE

Symbol	Definition	Units
$x, y, z$	x, y or z coordinate position	m
$\alpha, \beta, \gamma$	pitch, roll, and yaw angles	rad (or degree if specified)
$p$	position vectors	m
$R$	rotation matrix	-
$t$	translational vector	m
$T$	homogeneous transformation matrix	-
$U, \Sigma, V$	singular value decomposition	-
$v$	normalized directional vector	-
$D$	observation matrix formed by $v$	-
$n$	number of points	-
$\mathcal{F}$	frame - Cartesian coordinate system	-
$b$	body vectors	m
$b'$	local body vectors	m
$q$	motion parameters ( $s$ and $\theta$ )	-
$\theta$	rotation angles	rad (or degree if specified)
$I$	identity matrix	-
$J$	Jacobian matrix	-
$g$	geometry of a robot ( $\mathbf{b}$ s and $\mathbf{R}$ s)	m and rad
$X$	state of dynamic pose estimation	m and rad
$A$	point coordinates of target 1	m
$B$	point coordinates of target 2	m
$c$	compatibility of relaxation	-
$\varepsilon$	residual	-
$s$	merit scores	-

Subscript	Definition
$i$	$i^{\text{th}}$ point
$n$	total number of points
$tcp$	tool central point
$r$	robot base
$m$	measurement
$p$	working panel
$tip$	the tip of robot
$t$	target
$min$	minimum
$max$	maximum

# CHAPTER 1 INTRODUCTION

## 1.1 Background

In aerospace industry, high precision is necessary due to manufacturing of airplanes for their improved characteristics about safety, fuel consumption, noise pollution reduction, and comfort for passengers. In aerospace applications ranging from material handling to fastening and assembly, precise positioning plays a crucial role. Robots offer the prospect of removing much of the tedious, unattractive, and in some cases dangerous manufacturing tasks, greatly enhancing the working environment. But, most of today's industrial robots do not have good three dimensional (3D) absolute positioning capacity, though their repeatability following taught positions is high enough. As in most of aerospace manufacturing environment, the complex working parts may have several thousands of positions and trajectories, at which a robot stops or passes. It is impossible to teach the robot at each position. As well, the varieties and the forms of parts may not be always rigid. For flexible fabrication with high precision, adaptive metrology-guidance robotic systems are increasingly demanded in aerospace manufacturing.

Riveting is one of the major joining methods used in aircraft assembly. Manual riveting operations are tedious, repetitious, prone to error, and likely causing health and ergonomic problems [1]. Currently, automated riveting systems are being developed to replace manual riveting operations, such as using dual robots to provide panel holding and feeding functions for conventional squeezing riveting machines [2], as shown in Figure 1.1. More recently, a novel robotic percussive riveting system has been developed in Ryerson University for percussive riveting automation [3]. The system adopts a robot

to hold and move a percussive riveting gun equipped with a rivet feeder, and a gantry to hold a working panel and to move a bucking bar. There are two main separating steps in the robotic percussive riveting process. One is called hole-drillings in which all the holes are first drilled based on a planned riveting path and another one is called rivet-in-hole insertion or called peg-in-hole insertion along the same path. Prior to the riveting process, rivet-in-hole insertion requires a robot to first position and then insert a rivet precisely into a hole on the working panel without engaging the panel to prevent potential damage.

Practically, there always exist undesired errors, namely system uncertainties which significantly influence the positioning accuracy and the stability of the robotic riveting system. The uncertainties can be briefly grouped into two parts, which are the intrinsic parameters such as robot positioning errors due to mechanical deflection, assembly clearance, machining tolerance and thermal expansion, and the extrinsic parameters such as alignment errors, tooling calibration errors, and external disturbances including vibration, drill bit wear, rivet feeder positioning error, etc. In general, the tooling calibration errors can be reduced through well calibration process, but robot positioning errors always exist in a robotic riveting system. Generally, the rivet holes are drilled with a tolerance of approximately 0.2 mm [4]. The performance of the adopted ABB 4400/45 robot can only assure unidirectional pose repeatability of 0.07 mm, linear path accuracy of 0.8 - 1.3 mm and linear path repeatability of 0.25 - 0.4 mm at rated load and 1m/s velocity on a inclined ISO test plane according to ISO 9283 [5]. Apparently, the robot path repeatability would not be able to guarantee a successful rivet insertion from the tip of the rivet gun to the inside of the hole. As a result, in the hole-drilling step of the robotic riveting process, even though the holes are drilled according to a planned path,

they are still deviated due to the presence of the system uncertainties. Unsuccessful rivet insertion would cause damage to sheet metal skins or to the tooling system.

A great deal of research has been done on peg-in-hole insertion [6-11]. In [6], the geometric analysis and force/moment analysis of three-dimensional multiple-peg insertion are presented. Because of the difficulty and complexity in analyzing three-dimensional peg-in-hole insertion, most analyses of this class of assembly tasks were done by simplifying the problems into two dimensions [7, 8]. Surges [9] analyzed the wedging in three-dimensional rectangular peg insertion. A hybrid force-position strategies using active and passive mechanism were designed in three-dimensional insertion [10]. In [11], a compliant mechanism was designed for axis symmetric part insertion to passively compensate for position and orientation uncertainties during an assembly process. However, the approaches on three-dimensional peg-in-hole insertions in assembly tasks depend on the geometric features of assembly objects or the interaction forces between pegs and holes, thus it cannot be used in the rivet-in-hole insertions of the robotic riveting process. Therefore, a robust and efficient control scheme should be explored for rivet-in-hole insertions in robotic riveting system.

One feasible and cost-effective way is to use robot visual servoing (RVS) method to achieve high precision positioning. RVS systems constitute a class of opto-mechatronic systems in which measurement data are integrated into the robot control system to enhance the robot control performance [12]. RVS systems can be classified into two main categories [13]: position-based visual servoing (PBVS) [14] and image-based visual servoing (IBVS) [15]. IBVS is only stable locally and may suffer from Jacobian singularity and local minima [16]. It usually induces the unpredictable and sometimes

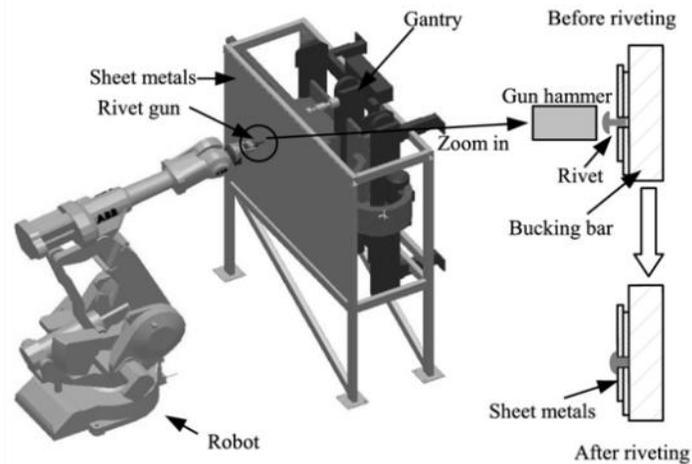
undesired 3D trajectories. In contrast, PBVS is a model-based type of visual servoing and known to be global stable. It utilizes the pose (position and orientation) of object of interest with respect to the camera, therefore, sometimes referred to pose-based visual servoing. However, since it requires a 3D model of a target, the main drawback is reliance upon calibration and usually it has no control of the image plane and may easily lose the visual features that are used to estimate a pose [17]. In PBVS, first the estimation of the pose of the target with respect to the camera is performed, and this information is used to produce the appropriate control signal to track the target.

To do so, pose estimation serves as an essential step of estimating the pose of object of interest with respect to the camera based on visual measurements. However, when multiple objects of interest are tracked, relative poses between these objects are demanded in the visual servoing. In this research, unlike the conventional way of estimating the pose of each object of interest with respect to the camera, relative pose estimation is proposed to directly estimate the relative pose between rigid body objects. Specifically for the purpose of robot precise positioning and rivet insertion control, the online state estimations of the relative pose and motion are desired between the tool center point (TCP) of the robot tooling and the working panel. For this reason, a metrology system of a position visual sensor is introduced to guide the robot and the research of relative pose estimation between the robot and panel has been carried out pertaining to these challenges.

## **1.2 Robotic Percussive Riveting**

The robotic percussive riveting system is a new application for aircraft assembly

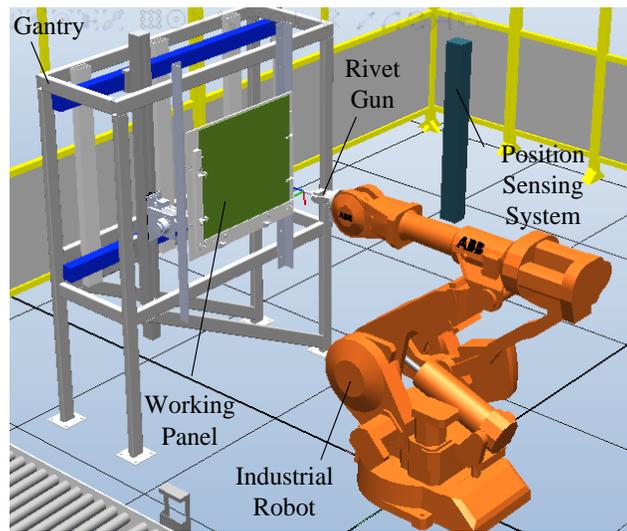
automation. The prototype of the system is located at EPH 107, Ryerson University. Figure 1.1 shows the principle of robotic percussive riveting. It includes a 6-DOF industrial robot that replaces the first worker for holding/moving a percussive rivet gun; a 5-axis computer numerical control (CNC) gantry system that replaces the second worker for holding/moving a bucking bar [3]. A working panel of sheet metals is loaded on the jig of the gantry for riveting operations. The tooling of riveting gun is attached on the end-effector of the robot for riveting assembly of aero-structural parts on the working panel. A rivet is located at the TCP of the robot tooling. The details of the hardware and software development are provided in Appendix G.



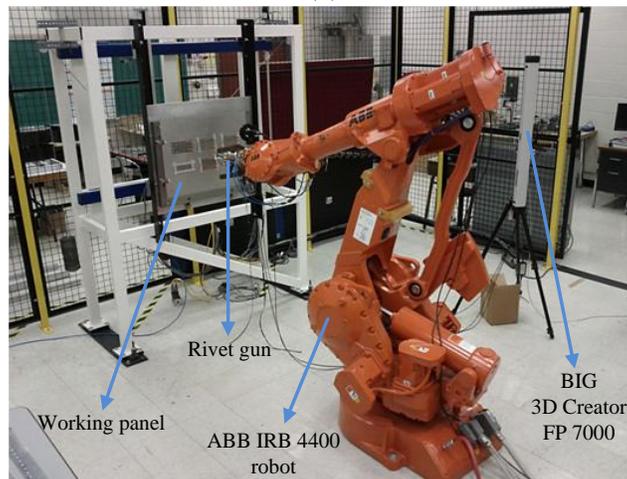
**Figure 1.1:** Principle of Robotic Percussive Riveting. [18]

For the need of precise positioning via metrology guidance, a position sensing system was adopted for position-based visual servoing in the robotic riveting system, as shown in Figure 1.2. The chosen position sensing system is a portable optical coordinate measurement machine (CMM), whose working principle is basically a camera-based triangulation system with 3 linear charge-coupled device (CCD) cameras, resulting in 1

horizontal and 2 vertical angle measurements [19]. The optical CMM is employed to measure the 3D coordinates of multiple feature points on the robotic tooling of riveting gun and the working panel on the gantry. Based on the measured point positions, research of relative pose estimation is to be carried out. Figure 1.2 (b) demonstrates the experimental setup of the robotic riveting system, including a robot of ABB IRB4400/45 and an optical CMM of 3D Creator FP7000 from Boulder Innovation Group (BIG).



(a)



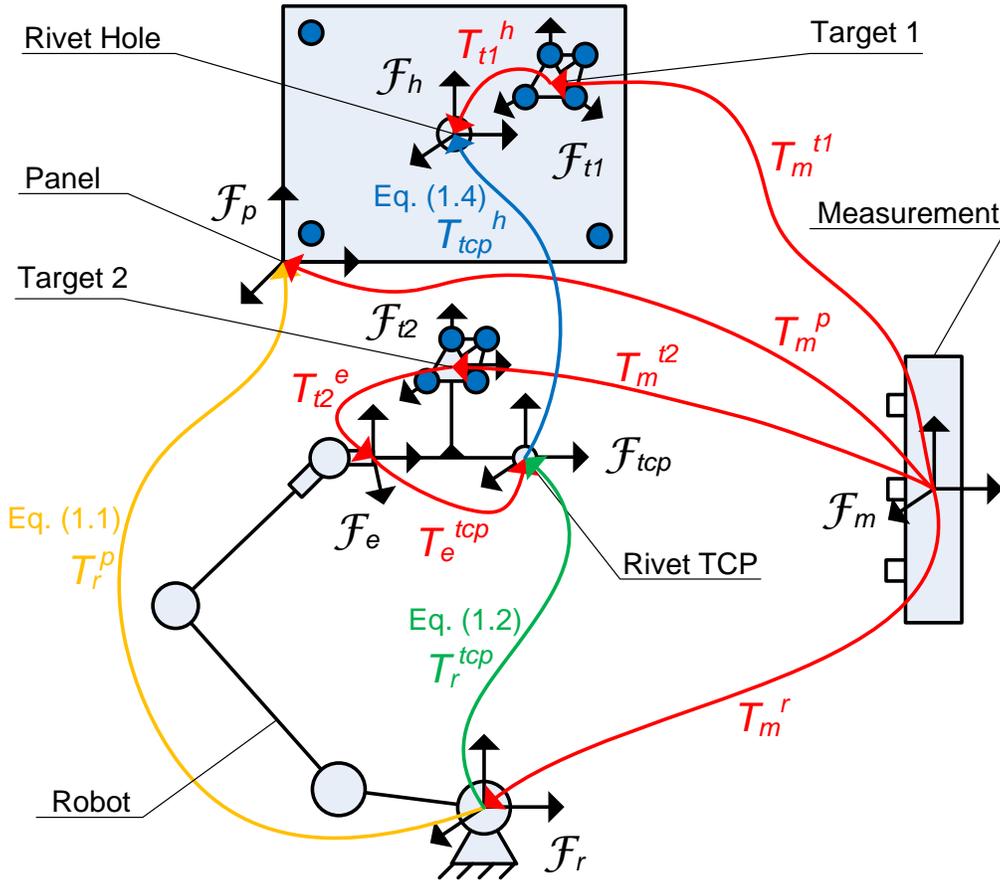
(b)

**Figure 1.2:** (a) System configuration setup simulated in ABB RobotStudio. (b) Experimental setup of the robotic riveting system at EPH107, Ryerson University.

### 1.3 Problem Formulation

The complete system transformational modeling is provided by assigning a number of Cartesian coordinate frames to the robotic riveting system, as shown in Figure 1.3. The frames are defined as follows:

- |  |  |
|--|--|
| $\mathcal{F}_p$ : Frame of working panel | $\mathcal{F}_{tcp}$ : Frame of TCP at the tip of a rivet |
| $\mathcal{F}_m$ : Frame of measurement   | $\mathcal{F}_e$ : Frame of robot end-effector            |
| $\mathcal{F}_r$ : Frame of robot base    | $\mathcal{F}_{t1}$ : Frame of target 1 on the panel      |
| $\mathcal{F}_h$ : Frame of rivet hole    | $\mathcal{F}_{t2}$ : Frame of target 2 on the tooling    |



**Figure 1.3:** System transformational modeling with two loops of transformations. One between the panel, measurement, and robot base; while another one between the robotic tooling tip, end-effector, target, measurement, and robot base.

Figure 1.3 shows two loops of transformations. The first one describes the transformation between frames of the panel, measurement, and robot base, expressed in Eq. (1.1); the second one describes the transformations between frames of the robotic TCP, end-effector, target 2, measurement, and robot base, expressed in Eq. (1.2).

$$T_r^p = (T_m^r)^{-1}T_m^p \quad (1.1)$$

$$T_r^{tcp} = (T_m^r)^{-1}T_m^{t2}T_{t2}^eT_e^{tcp} \quad (1.2)$$

where  $T_y^x$  represents a homogeneous transformation matrix from a  $x$  frame to a  $y$  frame; subscripts  $p, m, r, tcp, e,$  and  $t2$  stand for the panel frame, measurement frame, robot base frame, TCP frame, end-effector frame, and target 2 frame, respectively. Table 1.1 shows the determination of these transformations in Eqs. (1.1) and (1.2).  $T_m^p, T_m^r, T_{t2}^e,$  and  $T_{t1}^h$  are determined by localization;  $T_e^{tcp}$  is obtained by tooling calibration.  $T_m^{t1}$  and  $T_m^{t2}$  are solved by state estimation based on point measurements on targets 1 and 2.

**Table 1.1:** Determination of Transformations

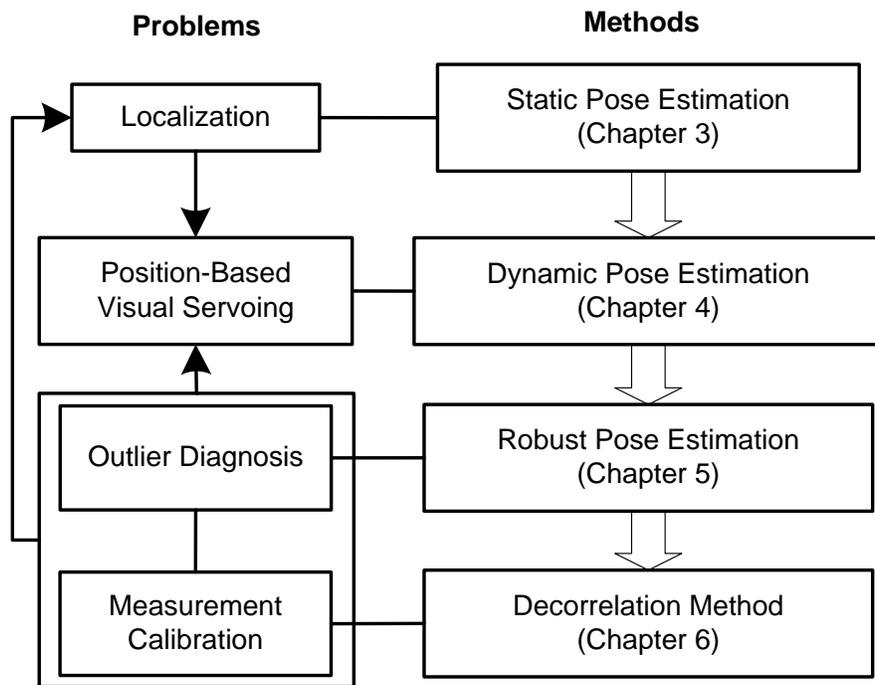
Transformation	Localization	Calibration	State Estimation
$T_m^p$	√		
$T_m^r$	√		
$T_e^{tcp}$		√	
$T_{t2}^e$	√		
$T_m^{t2}$			√
$T_{t1}^h$	√		
$T_m^{t1}$			√

## 1.4 Research Objective

The main research objective is to improve the positioning accuracy of rivet-in-hole insertion for the robotic percussive riveting system. The future successful promotion of the robotic percussive riveting system depends on the precise positioning of rivet-in-hole

insertion. The research approach is to first analyze the sources of error or uncertainty, and then develop methodologies pertaining to these challenges.

In general, there are three main sources of uncertainty, namely, localization, position-based visual servoing, and measurement, as illustrated in Figure 1.4. Localization is to transfer the coordinates of the rivet spots to those in the robot frame after proper alignment, which leads to the research of static pose estimation in Chapter 3. Position-based visual servoing is to control the robotic tooling tip to reach each rivet spot precisely based on state estimation of tip pose from visual sensing, which leads to the research of dynamic pose estimation in Chapter 4. Measurement accuracy can be improved through outlier diagnosis and measurement calibration, which leads to the research of robust pose estimation in Chapter 5, and a decorrelation method for measurement calibration in Chapter 6.



**Figure 1.4:** Structure of research of pose estimation - problems and methods.

Clearly, localization is a prior step to provide a desired tip pose for the position-based visual servoing. Outlier diagnosis and measurement calibration both contribute to provide reliable and accurate point measurements for localization and position-based visual servoing.

For localization, the key is to calculate the static relative pose between the panel and robot using Eq. (1.1). Multiple points attached on the working panel can be measured to estimate the pose transformation of the panel with respect to the measurement frame, denoted by  $T_m^p$ . Likewise, the method of static pose estimation can be applied to find the pose of robot base with respect to the measurement, denoted by  $T_m^r$ . Then,  $T_r^p$  can be calculated conventionally by Eq. (1.1). However, the research reported in Chapter 3 aims at a direct solution of  $T_r^p$  from these point measurements.

Upon the determination of  $T_r^p$ , the local coordinates of the rivet spots in the panel frame, denoted by  $p'$ , can be transferred to those in the robot base frame by Eq. (1.3), based on which the robot can be controlled to reach these desired spots using position-based visual servoing.

$$p = T_r^p p' \quad (1.3)$$

For position-based visual servoing, the key is the state estimation of tip pose from visual sensing using Eq. (1.2). To do so, a target is attached on the robotic tooling as a tracking reference for dynamic pose estimation of robotic tooling tip. Note that a target is a rigid body with multiple observable point features, in particular, artificial markers. Clearly, we have to dynamically estimate the pose of target with respect to the measurement frame based on point measurements, represented by  $T_m^{t2}$  in Eq. (1.2).  $T_e^{tcp}$  is the pose from the

tooling TCP to the robot end-effector, which can be determined from tooling calibration.  $T_{t2}^e$  is the pose from the robot end-effector to the target 2, which can be calculated by the static pose estimation.

However, for flexible manufacturing, panel frame is assumed to be mobile. In other words,  $T_m^p$  may vary during robot operations. Thus, direct state estimation is demanded for relative pose between rivet TCP frame on the tooling and rivet hole frame on the panel during motion, denoted as  $T_{tcp}^h$ . Conventionally,  $T_{tcp}^h$  can be calculated by Eq. (1.4) with provided state estimation of  $T_m^{t1}$  and  $T_m^{t2}$ . In Chapter 4, the reported research of dynamic pose estimation aims at direct state estimation of  $T_{tcp}^h$  using point measurements of targets 1 and 2 on the tooling and panel.

$$T_{tcp}^h = (T_m^{tcp})^{-1} T_m^h = (T_m^{t2} T_{t2}^e T_e^{tcp})^{-1} T_m^{t1} T_{t1}^h \quad (1.4)$$

## 1.5 Outline of Thesis

The remaining thesis is organized as follows.

Chapter 2 provides literature review on previous researches on the position-based visual servoing and pose estimation, including static, dynamic, and robust pose estimation.

Chapter 3 first summarizes a number of formulation of static pose estimation. Direct solutions of relative pose are derived using least-square methods. Pose estimation is then applied in the localization of the robotic riveting system.

Chapter 4 investigates the method of dynamic pose estimation, which addresses the state estimation of relative poses between two rigid bodies during motion using Iterative Extended Kalman filter method.

Chapter 5 provides the robust pose estimation for robustness against outliers of point measurements. A relaxation method is proposed for outlier diagnosis using constraints of

rigid body.

Chapter 6 proposes a data decorrelation method based on multivariate statistical analysis to preprocess the 3D coordinate measurements and find an optimal measurement configuration for calibration.

Chapter 7 summarizes the conclusions, contributions, and future work of the research .

## **CHAPTER 2 LITERATURE REVIEW**

This chapter describes a literature review on previous researches that have been done on position-based visual servoing and pose estimation, including static, dynamic, and robust pose estimation.

### **2.1 Pose Estimation**

A pose of a rigid body includes a position and an orientation, which describe six degrees of freedom (DOF) in 3D space with respect to a reference frame. Pose estimation for rigid bodies plays an important role in robotics, computer vision, and various positioning applications, such as visual servoing for robotics [14, 21-23], biomechanical motion [24], guidance for aircraft assembly automation [3], spacecraft docking and rendezvous [25], and satellite capture [26, 27], etc. The conventional 3D-to-3D pose estimation deals with one 3D rigid body with respect to an observing system, i.e., the pose transformation after a motion of one body.

Estimation is the process of inferring the value of a quantity of interest from indirect, inaccurate and uncertain observations [20]. In general, the variable that is to be estimated can be classified into two categories, namely, a parameter – a time-invariant quantity of a static system and the state of a dynamic system, which involves in time according to a stochastic equation. Consequently, two classes of estimation can be classified, namely, parameter estimation and state estimation. As for pose estimation, parameter estimation is referred to the static pose estimation, and state estimation is referred to the dynamic pose estimation. Further, robust pose estimation adds robustness to pose estimation, dealing

with pose estimation from outliers contaminated point measurements.

### 2.2.1 Static Pose Estimation

Static pose estimation is the parameter estimation of poses, which is to estimate the static poses based on noisy measurements of point features on rigid bodies. The closed-form solutions for static pose estimation are based on a maximum likelihood criterion or orthogonally constrained total least-square (CTLS) goodness-of-fit criterion when assuming an independent Gaussian noise modeling with zero mean and known variance. There are normally two forms of solutions, namely, batch and recursive LS estimators [20]. The former estimates pose parameters by processing the entire measurement data simultaneously. The latter deals with ongoing measurements and estimation, adopted for sequential rather than batch processing. Nevertheless, they are only suitable for static systems or constant pose estimation, where there is no motion during estimation. Further, due to the nonlinearity, the close-form CTLS solutions of static pose estimation tend to decouple the rotational and translational parameters and normally estimate the rotation first and translation later [28, 29].

There is a very extensive bibliography on the topic of pose estimation from corresponded 3D point sets in many disciplines, such as photogrammetry, computer vision, robotics, and psychometrics. Different terminologies have been defined, namely, absolute orientation [30], registration [31], rigid body motion [32], and orthogonal Procrustes problem [33]. A number of efficient close-form solutions for this CTLS problem have been found and compared; see the surveys by Wrobel [34], Eggert *et al.* [35], and Williams *et al.* [31]. The solutions differ in terms of orientation representations and

mathematical derivation. Some applied Singular Values Decomposition (SVD) or polar decomposition based on rotation matrix representation [36, 37]. While others applied the eigensystem analysis of matrices based on unit or dual quaternion representation of rotation [30, 38, 39]. In particular, the SVD approach had been further investigated and improved [40-44]. Kanatani [49] stated the theoretical equivalent of the polar decomposition and SVD based methods. Eggert et al. [35] compared these four solutions and concluded no difference in robustness against noise data. Essentially, these solutions are optimal in the least-squares sense under the statistical assumptions that the Gaussian noise on the point measurements is isotropic (equal distribution in all directions) and homogeneous (equal for all points in space) [23]. However, due to the nature of 3D sensors, point measurement error distributions are usually anisotropic (direction dependent) and heteroscedastic or inhomogeneous (point dependent) [46]. Ohta and Kanatani [46] and Matei and Meer [47] both tackled the problem under the general noise conditions with renormalization and multivariate errors-in-variances regression, respectively.

### **2.2.2 Dynamic Pose Estimation**

Kalman filter (KF) based state estimation provides a computational tractable solution for dynamic pose estimation [21]. Kalman filtering technique provides a way to estimate sequentially the state of a dynamic system using a sequence of noisy measurements made on the system. It not only considers kinematic motion of system model of rigid bodies but also estimates both rotational and translational parameters together as a state vector.

Kalman filter techniques consist of two models: system transition model and observation

model. Basically, the system transition model describes the evolution of the state with time; while the observation model maps the state estimate to noisy measurements. When the transition or/and observation models are nonlinear, linearization is applied during filtering. Although Kalman filter was originally derived for a linear problem, the extended Kalman filter (EKF) algorithm was introduced to provide a near-optimal solution for these nonlinear problems [26]. In terms of linearization errors, sufficiently high sampling rate enforces the accuracy of the linearization over the sampling period.

However, EKF-based solutions might easily diverge under fast and nonlinear trajectory dynamics, for example, quick changes of the pose, even with a relatively high sampling rate [20]. To solve this issue, iterative extended Kalman filter (IEKF) has been developed to improve the linearization by taking into account the measurements iteratively.

Recently, adaptive extended Kalman filter (AEKF) has been investigated for noise covariance adaptability due to the rising robustness requirements [20, 21]. The requirements are mainly caused by non-constant target moving speeds and accelerations, and the measurement vulnerability of traditional planar CCD cameras when illumination condition changes, etc.

### **2.2.3 Robust Pose Estimation**

Visual features of a rigid body are observed in two ways, namely, passive and active mode. The passive mode means that illumination provides only for the visualization. The active mode means that artificial illumination provides geometrical constraints. The natural features include edges, corners, colors, textures, etc. When natural features are not accurate or reliable, it leads to the employment of artificial illumination, such as infrared

LED markers.

A practical problem arises when measurements are contaminated with gross errors, called outliers. Outliers are anomalous observations lying outside the overall error distribution pattern. In reality, 3D measurements are vulnerable in a cluttered, dynamic, and unpredictable environment. Incorrect correspondences may cause outliers due to distractions from reflections, lighting variation, background colours and textures, and even algorithms of feature matching. Besides, sensing failure or invisibility may cause missing data due to obstruction or field-of-view constraints. Missing data is considered and treated as an extreme case of outliers. Some examples are studied in Appendix H.

The foregoing reviewed closed-form solutions for pose estimation are all based on a LS goodness-of-fit criterion. Unfortunately, LS based minimization has been well recognized as highly sensitive to outliers, since it tends to accommodate all data with sum of squares of residuals. In this respect, robust estimation has been investigated in the fields of statistics and computer vision over the last 35 years [48, 49]. Rousseeuw *et al.* [50] and Hodge *et al.* [51] have reviewed two ways of solution: diagnostics and accommodation. The former detects and removes or replaces outliers prior to further processing. The Mahalanobis distance (MD) approach has been a classical method widely used in multivariate outlier detection [52]. The latter replaces the LS criterion with less sensitive ones to accommodate all data and withstand the effect of outliers, such as  $L_1$  regression [53],  $L_\infty$  norm [54], least median of squares (LMedS), M-estimators, etc. [50]. However, robust estimators are not necessarily the only or even the best techniques that can be used to solve the problems caused by outliers in all contexts, since specialized heuristics for handling occasional outliers appear in specific contexts [48].

With these statistical techniques, researchers have attempted to solve the outlier problem in the field of pose estimation. Siegel *et al.* [55, 56] first considered the 2D and 3D rigid transformation for a robust fit using the repeated median algorithm in biometrics. Zhang [57] and Pennec *et al.* [58] attempted to refine the point matching and registration by thresholding data against a MD criterion. Zhuang and Huang [59] also tackled the robust registration problem in an iterative scheme of M-estimator. Boulanger *et al.* [60] employed an LMedS norm for a robust correspondence estimation between computer-aided design (CAD) models and measured points. Rosin [61] also applied the LMedS approach to determine the best minimum subset of point matches for pose estimation. Kumar and Hanson [62] compared the LMedS and M-estimator approaches in camera pose estimation using iterative techniques. Most recently, Enqvist *et al.* [54] studied the geometric conditions of the  $L_\infty$  norm to handle outliers in camera pose estimation. However, all these techniques worked in their specific contexts and assumptions and the fraction of outliers were usual low, and a robust and efficient solution of outlier diagnosis has not been provided to preprocess measurements prior to pose estimation of rigid body motions.

## **2.2 Position-Based Visual Servoing**

Visual servoing describes a class of closed-loop feedback control algorithms in robotics for which the control error is defined in terms of visual measurements [63]. It has diverse applications, including industrial automation, assistance in medical operations, control of ground vehicles, airborne vehicles, and robotic grasping. A comprehensive tutorial and review can be found in [64], and a review of more recent developments was discussed in

[65]. Basically, the relationship between the robot and camera is often described as moving camera (eye-in-hand) or fixed-camera at the kinematic level. In the former approach, one or more cameras are mounted on the end-effector of a robot and several targets are fixed in the workspace. The goal is to control the view of the camera with respect to targets. In the fixed setup, cameras are fixed and the targets are mounted on the end-effector and work-piece. The camera and end-effector are considered as working at opposing ends of a kinematic chain, and pose of the end-effector is controlled independently of the camera view [63].

In terms of control architecture, two types of techniques were first introduced by [66]: position-based and image-based visual servoing. In position-based visual servoing, the control error is calculated after reconstructing the pose of the end-effector from visual measurements. Joint angles are then driven by the error between the observed and desired pose. Conversely, the control error in image-based visual servoing is formulated directly as the difference between the observed and desired location of features (such as points and edges) on the image plane. Both methods have some disadvantage and advantages. The image-based approach relies on the image Jacobian (or interaction matrix), which transforms an image-space error to a pose or joint-space error. The main drawback is the singularities issue in the image Jacobian and it does not provide an explicit control of pose, which can lead to inefficient and unpredictable trajectories in Cartesian space. On the other hand, position-based visual servoing requires explicit reconstruction of the robot and target pose, and it leads to predictable trajectories and allows simple path planners to be directly incorporated into the controller. The position-based visual servoing is implemented in this thesis.

Visual servoing schemes can be further classified as direct visual servo<sup>1</sup> or dynamic look-and-move, depending on whether the control law directly generates joint-level inputs or Cartesian set-points for robot's joint-level controllers. For several reasons [64], almost all recent practical systems as well as the scheme proposed in this proposal adopt the dynamic look-and-move approach.

One more significant distinction should be made between endpoint open-loop (EOL) and endpoint closed-loop (ECL) [64]. An ECL controller observes both end-effector and targets to determine the control error, while an EOL controller observes only the target. In the latter case, the relative position of the robot is controlled using a known, fixed camera to robot end-effector (hand-eye) transformation. This transformation is established by the hand-eye calibration, which has a significant effect on the accuracy of EOL control; while the hand-eye calibration is not necessary for ECL control, since the pose of the end-effector is measured directly [63]. Moreover, it is always possible to transform the EOL control to ECL control by simply adding the observation of the end-effector. Thus, theoretically, the ECL system appears to be preferable to the EOL system. However, the ECL control requires the visibility of end-effector through the whole process of operation, and it may not be satisfied in the complex working conditions.

There are two levels of control dealing with robots, namely, joint level and workspace level. Joint level control applies methodology of inverse kinematics in the control of joint motion of robot manipulators. For current industrial robot manipulators, such as the ABB robot used in the robotic riveting system, joint-level control is already commercially

---

<sup>1</sup> The terminology was introduced by Hutchinson *et al.* [64] to avoid confusion, since “visual servo”, Sanderson and Weiss used in 1980 [66], has come to be accepted as a generic term for any type of visual control of a robotic system.

available and thus it is not the focus of the research. Therefore, the control is targeted on the higher level above the joint level - the workspace level control. The workspace level control deals with the control of trajectories or more simply paths of robots without concern of time in the workspace that sends the commands of desired poses to the joint level control. The workspace level control is just analogue to the jogging mode of robots, where users use a teach pendant to jog robot incrementally in either joint or workspace level, such as linear or oriented modes in an ABB robot. In some circumstances, the workspace level robotic control is considered as the dynamic trajectory planning due to the fact that there is no direct control of joints involved and it only dynamically plans the trajectories of robots during motion.

## **CHAPTER 3 STATIC POSE ESTIMATION**

This chapter presents the direct solution of static relative pose estimation between two rigid bodies based on four sets of corresponding point measurements using least-square methods. Conventionally, static pose estimation is targeted for an object with respect to a camera. Instead, the goal here is to find the direct static relative pose estimation between two rigid body objects, as described in Eq. (1.1). First, several practiced formulations of solutions are summarized. Second, three levels of least-square methods are discussed for solving the static pose estimation. Furthermore, direct relative static pose estimation is derived for determining the relative pose transformation between two rigid bodies, and compared with the indirect method. As an application, the relative pose estimation is employed in localization of the robotic riveting system.

### **3.1 Formulation of Pose Estimation**

Four commonly used formulations are summarized here for pose estimation, namely, Cartesian frame formulation, three point method, normalized directional vectors, and covariance matrix method. The first and second ones are based on three points. Specifically, first one establishes an orthonormalized frame, and the second one simply formulates non-orthogonal linear systems. The third and fourth ones can be applied to more than three points to enhance estimation accuracy. The adopted formulation is normalized directional vectors method, since normalization reduces the effect of point distribution.

### 3.1.1 Cartesian Frame Formulation

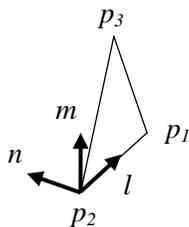
As shown in Fig. 3.1, based on three points ( $p_1$ ,  $p_2$ , and  $p_3$ ), a body-attached Cartesian coordinate system can be formulated using Gram-Schmidt orthonormalization procedure [67], i.e.,

$$R = [l \quad m \quad n] \quad (3.1)$$

where

$$l = \frac{p_1 - p_2}{\|p_1 - p_2\|}, m = \frac{(p_3 - p_2) - \frac{(p_3 - p_2) \cdot l}{l \cdot l} l}{\|(p_3 - p_2) - \frac{(p_3 - p_2) \cdot l}{l \cdot l} l\|}, \text{ and } n = l \times m \quad (3.2)$$

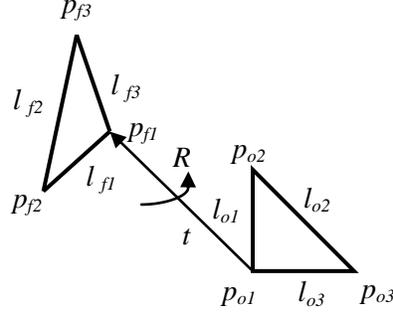
However, an orthogonal coordinate system is not necessary for determining a rotation. Furthermore, the method does not consider the measurement noise associated with the three points. Therefore, the following sections provide a simpler way by establishing non-orthogonal linear systems based on least-square methods.



**Figure 3.1:** Cartesian frame formulation.

### 3.1.2 Three-Point Method

A target with three non-collinear points, whose coordinates can be measured, is utilized to determine the pose transform. The three-point method estimates the pose changes through the initial and final position measurements of three targets, such as light emitting diode (LED) markers. As shown in Figure 3.2,  $p_{o1}$ ,  $p_{o2}$ , and  $p_{o3}$  are the initial positions of the three-point target; while  $p_{f1}$ ,  $p_{f2}$ , and  $p_{f3}$  are the final ones measured by certain devices.



**Figure 3.2:** Transformation determinations using the three-point position data [28].

The initial and final three line vectors formed by the three-point target are

$$[l_{o1} \quad l_{o2} \quad l_{o3}] = [p_{o1} - p_{o2} \quad p_{o2} - p_{o3} \quad p_{o3} - p_{o1}] \quad (3.3)$$

$$[l_{f1} \quad l_{f2} \quad l_{f3}] = [p_{f1} - p_{f2} \quad p_{f2} - p_{f3} \quad p_{f3} - p_{f1}] \quad (3.4)$$

Then, the rotation matrix from the initial pose to the final pose can be formulated by [28]

$$L_f = RL_o \quad (3.5)$$

where

$$L_o = [l_{o1} \quad l_{o2} \quad l_{o1} \times l_{o2}] \quad (3.6)$$

$$L_f = [l_{f1} \quad l_{f2} \quad l_{f1} \times l_{f2}] \quad (3.7)$$

### 3.1.3 Normalized Directional Vectors

In practice, normally, more than three points are employed. The redundancy of point features helps improve the estimation accuracy in the presence of occlusions. When more than three points are utilized for robustness, normalized line vectors can be constructed as directional vectors for each pair of the points  $i$  and  $j$ . Then, the rotation relating two corresponded directional vectors can be expressed as [29]

$$D_f = RD_o \quad (3.8)$$

where

$$D_o = [v'_{12} \quad v'_{13} \quad \cdots \quad v'_{ij} \quad \cdots \quad v'_{(n-1)n}] \in \mathbb{R}^{3 \times [n(n-1)/2]} \quad (3.9)$$

$$D_f = [v_{12} \quad v_{13} \quad \cdots \quad v_{ij} \quad \cdots \quad v_{(n-1)n}] \in \mathbb{R}^{3 \times [n(n-1)/2]} \quad (3.10)$$

where  $n$  is the number of points.  $v'_{ij}$ , and  $v_{ij}$  are the normalized vector from point  $i$  to  $j$ , calculated by

$$v'_{ij} = \frac{p'_j - p'_i}{\|p'_j - p'_i\|} \quad \text{and} \quad v_{ij} = \frac{p_j - p_i}{\|p_j - p_i\|} \quad (3.11)$$

### 3.1.4 Covariance Matrix Method

The covariance method is to adopt statistical characteristics of point patterns to formulate the similarity transformation. The mapping equation system of the rotation matrix can be formulated by

$$C_f = RC_o \quad (3.12)$$

where

$$C_o = [p_{o1} - \mu_o \quad p_{o2} - \mu_o \quad \cdots \quad p_{on} - \mu_o] \in \mathbb{R}^{3 \times n} \quad (3.13a)$$

$$C_f = [p_{f1} - \mu_f \quad p_{f2} - \mu_f \quad \cdots \quad p_{fn} - \mu_f] \in \mathbb{R}^{3 \times n} \quad (3.13b)$$

$\mu_o$  and  $\mu_f$  represent the mean value of two point sets  $p_o$  and  $p_f$ ,

$$\mu_o = \frac{1}{n} \sum_{i=1}^n p_o \quad \text{and} \quad \mu_f = \frac{1}{n} \sum_{i=1}^n p_f \quad (3.14)$$

Thus, the covariance matrix of two point patterns can be formulated as [40]

$$\text{cov}(p_o, p_f) = \frac{1}{n} C_o C_f^T \quad (3.15)$$

The solution based on the singular value decomposition of the above covariance matrix of the point data is given as a theorem in Appendix D.

## 3.2 Least Squares Methods

When considering the measuring noise in the coordinate data of these points, least square methods should be applied to estimate the optimal values for  $R$  in Equations (3.5), (3.8) and (3.12). As a result, three types of least square methods are summarized based on the method of normalized directional vectors in Eq. (3.8):

- Least squares (LS) – Assuming noisy perturbations only on  $D_f$  side.

$$\hat{R}D_o \approx D_f + \Delta D_f, \min \|\Delta D_f\|^2 \quad (3.16)$$

From Appendix C.5, the LS solution of  $R$  is

$$\hat{R} = D_f D_o^T (D_o D_o^T)^{-1} \quad (3.17)$$

where  $D_o^T (D_o D_o^T)^{-1}$  is the pseudo-inverse of  $D_o$ , as explained in Appendix C.3.

- Total least squares (TLS) – Assuming measurement errors exist in both  $D_o$  and  $D_f$ .

$$R(D_o + \Delta D_o) \approx D_f + \Delta D_f, \min(\|\Delta D_o\|_F + \|\Delta D_f\|_F) \quad (3.18)$$

The result is given in [28].

- Constrained total least square methods (CTLS) – Assuming measurement errors exist in both  $D_o$  and  $D_f$  and subject to the orthogonality of  $R$ .

$$R(D_o + \Delta D_o) \approx D_f + \Delta D_f, \min(\|\Delta D_o\|_F + \|\Delta D_f\|_F),$$

$$\text{subject to } \{R \in \mathbb{R}^{3 \times 3} | R^T R = I_3, \det(R) = +1\} \quad (3.19)$$

It is shown in Goryn and Hein [42] that the above problem yields a standard *orthogonal Procrustes problem* in Golub and Van Loan [68] and Appendix C.6, of which the intention is to find an orthogonal matrix  $R$ , which most nearly transforms  $D_o$  to  $D_f$  in the least-squares sense. The solution is

$$\hat{R} = VU^T \quad (3.20)$$

where the orthogonal  $3 \times 3$  matrices  $U$  and  $V$  are given in terms of the SVD of  $D_o D_f^T$ :

$$D_o D_f^T = U \Sigma V^T \quad (3.21)$$

This SVD method for determining  $R$  has the same form as the one presented in Arun *et al.* [36]. To always obtain a rotation matrix (and not a reflection matrix), the modification proposed in Umeyama [40] should be used. As given in Appendix D, the modified solution is

$$\hat{R} = V \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(VU^T) \end{bmatrix} U^T \quad (3.22)$$

Upon the calculation of  $R$ , the translational vector  $t_i$  for each point can be obtained by

$$t_i = p_i - \hat{R} p'_i \quad (3.23)$$

Then, the vector  $t$  for the rigid body can be computed by minimizing

$$\varepsilon(t) = \sum_{i=1}^n \|p_i - \hat{R} p'_i - t\|^2 \quad (3.24)$$

By taking a derivative with respect to  $t$  and setting the derivative to 0, it yields

$$\hat{t} = \sum_{i=1}^n (p_i - \hat{R} p'_i) / n \quad (3.25)$$

Clearly, the optimum solution of  $t$  is the mean of translations of all point vectors.

### 3.3 Direct Relative Pose Estimation between Two Rigid Bodies

This section presents the relative pose estimation of two rigid bodies based on four sets of corresponding point measurements using least-square methods, including LS and CTLS. Two approaches of relative pose estimation are developed and compared, namely, conventional (indirect) and strict (direct) least-square methods. The former is to first obtain individual least-square pose transformation of each rigid body from its two sets of point measurements, based on which the relative pose transformation is then calculated

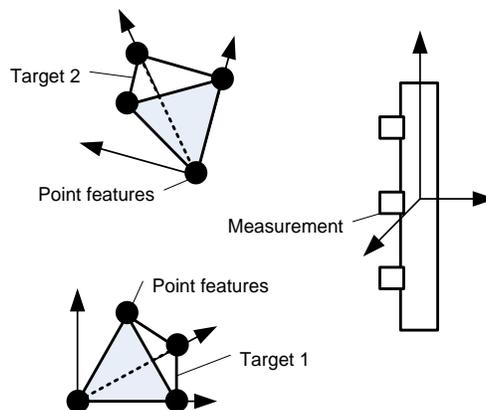
indirectly; while the latter is to formulate a direct transform mapping equation of point measurements for the relative pose transformation, based on which the least-square methods are directly applied to solve the relative pose transformation. The comparison has found and proved the inequality of indirect and direct ordinary LS solutions, and the equivalence of indirect and direct CTLS solutions. Thus, it is safe and acceptable to continue using the conventional indirect CTLS approach of relative pose estimation. Lastly, a case study of a robotic riveting system is provided to validate the result, aiming to estimate the optimal relative pose transformation between a robot end-effector and a work piece.

### **3.3.1 Motivation of Direct Relative Pose Estimation**

The estimation of pose transformation of a rigid body from two sets of corresponding point measurements is a fundamental and well-studied problem, as given in literature study in Section 2.2.1. However, the research of the relative pose transformation between two rigid bodies from four sets of point measurements have not yet received much attention. Conventionally, the relative pose transformation between two rigid bodies is indirectly determined by first estimating the pose transformation of each rigid body from its two sets of corresponding point measurements and then multiplying the inverse of one of them with another one, as given in Eq. (1.1). The arising question is whether the indirect solution from the conventional approach is still optimal or not? Therefore, this section aims to derive a strict (direct) least-square solution of relative pose estimation as oppose to the conventional (indirect) approach and then compare them to verify the differences.

### 3.3.2 Problem Statement

As shown in Figure 3.3, two rigid body targets with a certain number of point features are observed by a measurement system. Based on the observed point data, the underlying problem is to estimate the relative pose transformation between these two rigid bodies.



**Figure 3.3:** Two rigid body targets observed by a measurement system.

Thus, the following problem can be stated:

Let  $X = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{m \times n}$  and  $X' = [x'_1, x'_2, \dots, x'_n] \in \mathbb{R}^{m \times n}$  be two sets of corresponding non-coplanar points for a rigid body (Body 1) in  $m$ -dimensional space;  $Y = [y_1, y_2, \dots, y_k] \in \mathbb{R}^{m \times k}$  and  $Y' = [y'_1, y'_2, \dots, y'_k] \in \mathbb{R}^{m \times k}$  be another two sets of corresponding non-coplanar points for another rigid body (Body 2) in  $m$ -dimensional space. Note that the 3-dimensional cases, i.e.,  $m = 3$ .  $n$  and  $k$  are the number of points on Body 1 and Body 2, respectively.  $X$  and  $Y$  are the observed coordinate data of points with respect to a common measurement frame, denoted as  $\mathcal{F}_0$ ;  $X'$  and  $Y'$  are the local coordinates of points with respect to the body-fixed frames of Body 1 and Body 2, denoted as  $\mathcal{F}_1$  and  $\mathcal{F}_2$ , respectively. The objective is to find the relative pose transformation of frame  $\mathcal{F}_1$  with respect to frame  $\mathcal{F}_2$ , by minimizing the variance of

estimation errors caused by the Gaussian noises of the data of  $X, Y, X'$  and  $Y'$ . Suppose the Gaussian noises of point measurements are isotropic (equal distribution in all directions) and homogeneous (equal for all points in space). The relative pose transformation can be defined as a  $(m + 1) \times (m + 1)$  homogeneous transformation matrix, i.e.,

$$T_{12} = \begin{bmatrix} R_{12} & t_{12} \\ 0 & 1 \end{bmatrix} \quad (3.26)$$

where  $R_{12} \in \mathbb{R}^{m \times m} | R^T R = I_m, \det(R) = +1$  and  $t_{12} \in \mathbb{R}^{m \times 1}$  are the orthonormal rotation matrix and translation vector, respectively. Scaling factor is set to be one and therefore not considered here.

Conventionally, from Eq. (1.1),  $T_{12}$  can be solved by a transform equation as

$$T_{12} = T_{01}^{-1} T_{02} = \begin{bmatrix} R_{01}^{-1} R_{02} & R_{01}^{-1} (t_{02} - t_{01}) \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_{01}^T R_{02} & R_{01}^T (t_{02} - t_{01}) \\ 0 & 1 \end{bmatrix} \quad (3.27)$$

where

$$T_{01} = \begin{bmatrix} R_{01} & t_{01} \\ 0 & 1 \end{bmatrix} \quad (3.28)$$

$$T_{02} = \begin{bmatrix} R_{02} & t_{02} \\ 0 & 1 \end{bmatrix} \quad (3.29)$$

$T_{01}, T_{02} \in \mathbb{R}^{(m+1) \times (m+1)}$  are homogeneous transformations of frames  $\mathcal{F}_1$  and  $\mathcal{F}_2$  with respect to frame  $\mathcal{F}_0$ , respectively;  $R_{01} \in \mathbb{R}^{m \times m}$  and  $t_{01} \in \mathbb{R}^{m \times 1}$  are the orthonormal rotation matrix and translation vector of  $\mathcal{F}_1$  with respect to  $\mathcal{F}_0$ , respectively.  $R_{02} \in \mathbb{R}^{m \times m}$  and  $t_{02} \in \mathbb{R}^{m \times 1}$  are the orthonormal rotation matrix and translation vector of  $\mathcal{F}_2$  with respect to  $\mathcal{F}_0$ , respectively.

Based on  $T_{01}$  and  $T_{02}$ , the transform mappings of  $X'$  and  $Y'$  from frames  $\mathcal{F}_1$  and  $\mathcal{F}_2$  to frame  $\mathcal{F}_0$  can be described in Eqs. (3.30) and (3.31), respectively.

$$\begin{bmatrix} X \\ 1 \end{bmatrix} = \begin{bmatrix} R_{01} & t_{01} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X' \\ 1 \end{bmatrix} \quad (3.30)$$

$$\begin{bmatrix} Y \\ 1 \end{bmatrix} = \begin{bmatrix} R_{02} & t_{02} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Y' \\ 1 \end{bmatrix} \quad (3.31)$$

To solve for  $R_{01}$  and  $R_{02}$ , the rotation mapping equations of body 1 and body 2 can be formulated using the normalized directional vectors method in Eq. (3.8) as

$$D_{f1} = R_{01}D_{o1} \quad (3.32)$$

$$D_{f2} = R_{02}D_{o2} \quad (3.33)$$

where  $D_{o1} \in \mathbb{R}^{m \times n}$  and  $D_{f1} \in \mathbb{R}^{m \times n}$  are based on the corresponding points of body 1, i.e.,  $X'$  and  $X$ , respectively;  $D_{o2} \in \mathbb{R}^{m \times k}$  and  $D_{f2} \in \mathbb{R}^{m \times k}$  are based on the corresponding points of body 2, i.e.,  $Y'$  and  $Y$ , respectively. It is obvious that  $r(D_{o1}) = r(D_{f1}) = r(D_{o2}) = r(D_{f2}) = m$ , provided that the points of rigid bodies are well corresponded and non-coplanar.

However, a direct mapping equation of  $T_{12}$  associated with the observations  $X, Y, X'$  and  $Y'$  is much more appreciated than the conventional solution in Eq. (3.27), in the sense of directly estimating parameters from noisy observations. Therefore, the transform mapping of  $Y'$  from its description in frame  $\mathcal{F}_2$  to a description in frame  $\mathcal{F}_1$  can be described in terms of  $T_{12}$  as

$$T_{10} \begin{bmatrix} Y \\ 1 \end{bmatrix} = T_{12} \begin{bmatrix} Y' \\ 1 \end{bmatrix} \quad (3.34)$$

where  $T_{10}$  is the transform mapping from  $\mathcal{F}_0$  to  $\mathcal{F}_1$ , defined as

$$T_{10} = \begin{bmatrix} R_{10} & t_{10} \\ 0 & 1 \end{bmatrix} \quad (3.35)$$

which can be characterized in the following transform mapping of  $X$  from  $\mathcal{F}_0$  to  $\mathcal{F}_1$ ,

$$\begin{bmatrix} X' \\ 1 \end{bmatrix} = \begin{bmatrix} R_{10} & t_{10} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ 1 \end{bmatrix} \quad (3.36)$$

Theoretically,  $T_{10}$  is the inverse of  $T_{01}$  in Eq. (3.28), i.e.,

$$T_{10} = T_{01}^{-1} = \begin{bmatrix} R_{01}^T & -R_{01}^T t_{01} \\ 0 & 1 \end{bmatrix} \quad (3.37)$$

Thus, the rotation mapping equations for  $R_{12}$  and  $R_{10}$  can be formulated in Eqs. (3.38) and (3.39) by multiplying  $R_{10}$  with both sides of Eq. (3.33) or decoupling  $R_{12}$  and  $R_{10}$  from Eqs. (3.34) and (3.36).

$$D_{12} = R_{12} D_{o2} \quad (3.38)$$

$$D_{o1} = R_{10} D_{f1} \quad (3.39)$$

where

$$D_{12} = R_{10} D_{f2} \quad (3.40)$$

Based on the above conventional and strict formulations of relative pose transformation, the LS and CTLS methods are applied to solve the relative rotation matrix. Also, the conventional and strict solutions of  $R_{12}$  are compared. The objective is to investigate the differences and find out the reasons behind them.

### 3.3.3 LS solution

Without the error assumption of point data and  $n = k = m = r(D_{o1}) = r(D_{o2})$ , the theoretical solutions of  $R_{01}$  and  $R_{02}$  can be determined in Eqs. (3.41) and (3.42) from the aforementioned rotation mapping equations in Eqs. (3.32) and (3.33).

$$R_{01} = D_{f1} D_{o1}^{-1} \quad (3.41)$$

$$R_{02} = D_{f2} D_{o2}^{-1} \quad (3.42)$$

Thus, the theoretical solution of  $R_{12}$  can be derived based on Eq. (3.27) as

$$R_{12} = R_{01}^{-1}R_{02} = (D_{f1}D_{o1}^{-1})^{-1}D_{f2}D_{o2}^{-1} = D_{o1}D_{f1}^{-1}D_{f2}D_{o2}^{-1} \quad (3.43)$$

When considering the independent Gaussian errors in Eqs. (3.32) and (3.33), i.e.,  $D_{f1} = R_{01}D_{o1} + \varepsilon_1$  and  $D_{f2} = R_{02}D_{o2} + \varepsilon_2$ , and the number of points is more than  $m$ , i.e.,  $n > m = r(D_{o1})$  and  $k > m = r(D_{o2})$ , the LS solutions of  $R_{01}$  and  $R_{02}$  are given as

$$\hat{R}_{01} = D_{f1}D_{o1}^+ = D_{f1}D_{o1}^T(D_{o1}D_{o1}^T)^{-1} \quad (3.44)$$

$$\hat{R}_{02} = D_{f2}D_{o2}^+ = D_{f2}D_{o2}^T(D_{o2}D_{o2}^T)^{-1} \quad (3.45)$$

where  $D_{o1}^+$  and  $D_{o2}^+$  are the Moore-Penrose (M-P) inverses of  $D_{o1}$  and  $D_{o2}$ , respectively. Since  $r(D_{o1}) = r(D_{o2}) = m$ , i.e., have full row rank, we have  $D_{o1}^+ = D_{o1}^T(D_{o1}D_{o1}^T)^{-1}$ ,  $D_{o2}^+ = D_{o2}^T(D_{o2}D_{o2}^T)^{-1}$ .

Thus, the conventional (indirect) solution of  $R_{12}$  can be obtained based on  $\hat{R}_{01}$  and  $\hat{R}_{02}$  as

$$\begin{aligned} R_{12} &= \hat{R}_{01}^{-1}\hat{R}_{02} = (D_{f1}D_{o1}^+)^{-1}D_{f2}D_{o2}^+ \\ &= \left(D_{f1}D_{o1}^T(D_{o1}D_{o1}^T)^{-1}\right)^{-1}D_{f2}D_{o2}^T(D_{o2}D_{o2}^T)^{-1} \\ &= D_{o1}D_{o1}^T(D_{f1}D_{o1}^T)^{-1}D_{f2}D_{o2}^T(D_{o2}D_{o2}^T)^{-1}. \end{aligned} \quad (3.46)$$

However, the conventional (indirect) solution might not be the LS solution for  $R_{12}$ . Therefore, the following is to directly derive the strict LS solution of  $R_{12}$  based on its direct rotation mapping equation in Eq. (3.38) and compare it with the conventional solution in Eq. (3.46). Clearly, the strict LS solution of  $R_{12}$  can be solved by

$$\hat{R}_{12} = \operatorname{argmin}\|D_{12} - R_{12}D_{o2}\|_2^2 = D_{12}D_{o2}^+ = R_{10}D_{f2}D_{o2}^+ \quad (3.47)$$

From the rotation mapping equation of  $R_{10}$  in Eq. (3.39), we have

$$\hat{R}_{10} = D_{01}D_{f_1}^+ \quad (3.48)$$

where  $D_{f_1}^+ = D_{f_1}^T(D_{f_1}D_{f_1}^T)^{-1}$  is the M-P inverse of  $D_{f_1}$ .

Substituting Eq. (3.48) into Eq. (3.47), it yields the strict (direct) LS solution of  $R_{12}$

$$\hat{R}_{12} = \hat{R}_{10}\hat{R}_{02} = D_{01}D_{f_1}^+D_{f_2}D_{o_2}^+ = D_{01}D_{f_1}^T(D_{f_1}D_{f_1}^T)^{-1}D_{f_2}D_{o_2}^T(D_{o_2}D_{o_2}^T)^{-1} \quad (3.49)$$

Clearly, it is observed that the strict LS solution in Eq. (3.49) can be quickly achieved by replacing the inverses of  $D_{f_1}$  and  $D_{o_2}$  of the theoretical solution in Eq. (3.43) with their M-P inverses. Thus, Eq. (3.49) provides a general solution of  $R_{12}$ . More importantly, the differences are observed for the conventional (indirect) and strict (direct) LS solutions of  $R_{12}$  in Eqs. (3.46) and (3.49). Specifically, the inequality of the conventional and strict LS solutions, i.e.,  $\hat{R}_{12} \neq R_{12}$ , can be proved by applying the following lemma, i.e.,  $\hat{R}_{10} \neq (\hat{R}_{01})^{-1}$ . This implies that a LS solution of a rotation matrix loses its orthonormal property, i.e.,  $\{R \in \mathbb{R}^{m \times m} | R^T R = I_m, \det(R) = +1\}$ . Therefore, the CTLS solution of relative pose transformation considering the orthonormal constraint is discussed in the next section.

**Lemma 1:** Let  $R_{01} = D_{f_1}D_{o_1}^+$  and  $R_{10} = D_{o_1}D_{f_1}^+$ .  $D_{o_1} \in \mathbb{R}^{m \times n}$ ,  $D_{f_1} \in \mathbb{R}^{m \times n}$ ,  $r(D_{o_1}) = r(D_{f_1}) = m$ .  $D_{o_1}^+ = V_{o_1}\Sigma_{o_1}^+U_{o_1}^T$  and  $D_{f_1}^+ = V_{f_1}\Sigma_{f_1}^+U_{f_1}^T$  are the Moore-Penrose inverses of  $D_{o_1}$  and  $D_{f_1}$  based on their SVD, respectively. An inequality about  $R_{01}$  and  $R_{10}$  can be concluded that  $R_{10} \neq (R_{01})^{-1}$ .

**Proof of Lemma 1:** Suppose  $R_{10} = (R_{01})^{-1}$ , i.e.  $D_{o_1}D_{f_1}^+ = (D_{f_1}D_{o_1}^+)^{-1}$ , then

$$D_{o_1}D_{f_1}^+D_{f_1}D_{o_1}^+ = I. \quad (3.50)$$

Computing the product of  $D_{o1}D_{o1}^+$  and  $D_{f1}^+D_{f1}$  reveals

$$D_{o1}D_{o1}^+ = U_{o1}\Sigma_{o1}V_{o1}^TV_{o1}\Sigma_{o1}^+U_{o1}^T = U_{o1}I_mU_{o1}^T = I_m \quad (3.51)$$

$$D_{f1}^+D_{f1} = V_{f1}\Sigma_{f1}^+U_{f1}^TU_{f1}\Sigma_{f1}V_{f1}^T = [V_1|V_2] \begin{bmatrix} I_m & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} = V_1V_1^T \quad (3.52)$$

where  $\Sigma_{o1} \in \mathbb{R}^{m \times n}$  and  $\Sigma_{f1} \in \mathbb{R}^{m \times n}$  includes the singular values of  $D_{o1}$  and  $D_{f1}$ , i.e.,

$$\Sigma_{o1} = \begin{bmatrix} \sigma_{o1} & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \sigma_{om} & 0 & \cdots & 0 \end{bmatrix} \quad \text{and} \quad \Sigma_{o1}^+ = \begin{bmatrix} \sigma_{o1}^{-1} & 0 & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & \sigma_{om}^{-1} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{n \times m} \quad (3.53)$$

$$\Sigma_{f1} = \begin{bmatrix} \sigma_{f1} & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \sigma_{fm} & 0 & \cdots & 0 \end{bmatrix} \quad \text{and} \quad \Sigma_{f1}^+ = \begin{bmatrix} \sigma_{f1}^{-1} & 0 & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & \sigma_{fm}^{-1} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{n \times m} \quad (3.54)$$

$$V_{f1} = [V_1|V_2] \in \mathbb{R}^{n \times n} \quad (3.55)$$

where  $V_1 \in \mathbb{R}^{n \times m}$  and  $V_2 \in \mathbb{R}^{(n-m) \times n}$  are orthonormal bases for the row space of  $D_{f1}$ ,

denoted by  $R(D_{f1}^T)$  and the nullspace of  $D_{f1}$ , denoted by  $N(D_{f1})$ , respectively.

Hence, Eq. (3.56) can be obtained from Eqs. (3.50) and (3.51).

$$D_{o1}D_{f1}^+D_{f1}D_{o1}^+ = D_{o1}D_{o1}^+ \quad (3.56)$$

yields that

$$D_{f1}^+D_{f1} = I \quad (3.57)$$

which is conflicted with the property of  $D_{f1}^+D_{f1} = V_1V_1^T \neq I$  in Eq. (3.52).

Thus, we prove that

$$R_{10} \neq (R_{01})^{-1} \quad (3.58)$$

This concludes the lemma.

### 3.3.4 CTLS solution

When considering the independent Gaussian errors in both sides of the aforementioned rotation mapping equations in Eqs. (3.32) and (3.33), i.e.,  $D_{f_1} + \varepsilon_{f_1} = R_{01}(D_{o_1} + \varepsilon_{o_1})$  and  $D_{f_2} + \varepsilon_{f_2} = R_{02}(D_{o_2} + \varepsilon_{o_2})$ , and the orthogonality of  $R_{01}$  and  $R_{02}$ , the constrained total least square (CTLS) solutions of  $R_{01}$  and  $R_{02}$  are given by Eq. (3.22) as

$$\hat{R}_{01} = U_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(U_1 V_1^T) \end{bmatrix} V_1^T \quad (3.59)$$

$$\hat{R}_{02} = U_2 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(U_2 V_2^T) \end{bmatrix} V_2^T \quad (3.60)$$

where the orthogonal matrices  $U_1$  and  $V_1$ ,  $U_2$  and  $V_2$  are produced in terms of the SVD of  $D_{f_1} D_{o_1}^T$  and  $D_{f_2} D_{o_2}^T$ , i.e.,

$$D_{f_1} D_{o_1}^T = U_1 \Sigma_1 V_1^T \quad (3.61)$$

$$D_{f_2} D_{o_2}^T = U_2 \Sigma_2 V_2^T \quad (3.62)$$

Thus, the conventional (indirect) CTLS solution of  $R_{12}$  in Eq. (3.27) can be obtained by substituting the  $\hat{R}_{01}$  and  $\hat{R}_{02}$  in Eqs. (3.59) and (3.60) as

$$\begin{aligned} R_{12} &= \hat{R}_{01}^{-1} \hat{R}_{02} = \hat{R}_{01}^T \hat{R}_{02} \\ &= \left( U_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(U_1 V_1^T) \end{bmatrix} V_1^T \right)^T U_2 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(U_2 V_2^T) \end{bmatrix} V_2^T \\ &= V_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(V_1 U_1^T) \end{bmatrix} U_1^T U_2 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(U_2 V_2^T) \end{bmatrix} V_2^T \end{aligned} \quad (3.63)$$

Next is to directly derive the strict CTLS solution from the rotation mapping equation of  $R_{12}$  in Eq. (3.38). When considering errors in both  $D_{12}$  and  $D_{02}$  and the constraint that the

rotation matrix of  $R_{12}$  is orthonormal, the problem can be described as follows:

$$(D_{12} + \Delta D_{12}) \approx R_{12}(D_{02} + \Delta D_{02}), \min \|\Delta D_{12}\|_F^2 + \|\Delta D_{02}\|_F^2, \quad (3.64)$$

subject to  $\{R \in \mathbb{R}^{3 \times 3} | R^T R = I_3, \det(R) = +1\}$ ,

From the mapping equation of  $R_{10}$  in Eq. (3.39), the CTLS solution of  $R_{10}$  is given as

$$\hat{R}_{10} = V_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(V_1 U_1^T) \end{bmatrix} U_1^T \quad (3.65)$$

where the SVD of  $D_{01} D_{f1}^T$  can be obtained based on Eq. (3.61) as

$$D_{01} D_{f1}^T = (D_{f1} D_{01}^T)^T = (U_1 \Sigma_1 V_1^T)^T = V_1 \Sigma_1 U_1^T \quad (3.66)$$

Thus, the strict (direct) CTLS solution of  $R_{12}$  can be determined by

$$\begin{aligned} \hat{R}_{12} &= U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(UV_2^T) \end{bmatrix} V_2^T \\ &= V_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(V_1 U_1^T) \end{bmatrix} U_1^T U_2 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(U_2 V_2^T) \end{bmatrix} V_2^T = \hat{R}_{10} \hat{R}_{02} \end{aligned} \quad (3.67)$$

based on the SVD of  $D_{12} D_{02}^T$  by substituting Eqs. (3.62) and (3.65), i.e.,

$$D_{12} D_{02}^T = \hat{R}_{10} D_{f2} D_{02}^T = U \Sigma_2 V_2^T \quad (3.68)$$

where

$$U = V_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(V_1 U_1^T) \end{bmatrix} U_1^T U_2 \quad (3.69)$$

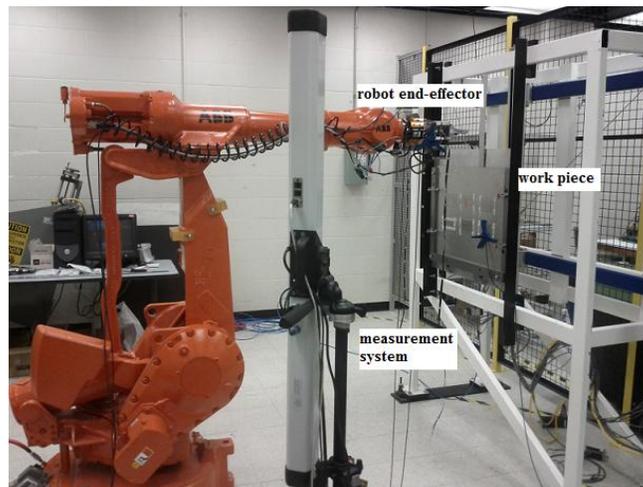
$$\det(UV_2^T) = (\det(V_1 U_1^T))^2 \det(U_2 V_2^T) = \det(U_2 V_2^T) \quad (3.70)$$

Comparing the conventional (indirect) and strict (direct) CTLS solutions in Eq. (3.63) and (3.67), the equivalence of them can be clearly found due to the fact that the CTLS solutions of rotation matrix maintains the orthogonality, i.e.,  $\hat{R}_{01}^{-1} = \hat{R}_{01}^T = \hat{R}_{10}$ .

Hence, it is safe and acceptable to employ the conventional CTLS approach of relative pose estimation. It indicates that any approach that enforces all required constraints and that is based on the same cost function, must lead to the same result. For verification, a case study of a robotic riveting system is presented in the following section.

### 3.3.5 Case Studies

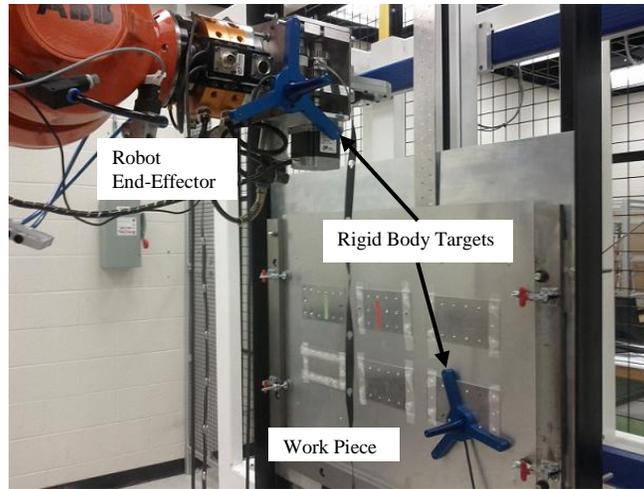
As shown in Figure 3.4, a robotic riveting system consists of a robot end-effector, a work piece, and a measurement system. In this case,  $T_{12}$  represents the relative pose transformation between the robot end-effector and the work piece to be estimated for the purpose of precise rivet insertion.  $T_{01}$  represents the pose transformation of the robot end-effector with respect to the measurement system;  $T_{02}$  represents the pose transformation of the work piece with respect to the common measurement system. The underlying problem becomes finding the optimal relative pose transformation of the work piece with respect to the robot end-effector based on the measurements of points on both bodies.



**Figure 3.4:** Experimental setup of a robotic riveting system.

To represent the points on rigid bodies, rigid body targets are attached and fixed on the robot end-effector and the work piece at predefined locations, as shown in Figure 3.5.

Each target has four observable points at the ends of four frames. The observed coordinates of points of rigid body targets ( $X$  and  $Y$ ) with respect to the frame of the measurement system are given in Table 3.1. The local coordinates of the points ( $X'$  and  $Y'$ ) with respect to the body-fixed frames were pre-measured and collected in Table 3.2. Therefore, the problem of finding the relative pose transformation between the robot end-effector and the work piece becomes obtaining the optimal relative pose transformation between these two rigid body targets.



**Figure 3.5:** Two rigid body targets on the work piece and the robot end-effector.

**Table 3.1:** Observed Coordinates of Points of Rigid Body Targets w.r.t. the Frame of the Measurement System

Bodies	Observed Coordinates of Points (unit: $m$ )				
		Point 1	Point 2	Point 3	Point 4
Robot end-effector ( $X$ )	x	-0.298104	-0.254750	-0.378223	-0.261440
	y	0.106871	0.123163	0.070411	0.008391
	z	-1.678396	-1.774297	-1.738408	-1.696042
Work piece ( $Y$ )	x	0.176478	0.101090	0.238220	0.188659
	y	0.267341	0.308396	0.296363	0.370121
	z	-1.911722	-1.974734	-1.993495	-1.886204

**Table 3.2: Local Coordinates of Points of Rigid Body Targets**

Bodies	Local Coordinates of Points on Bodies (unit: $m$ )				
		Point 1	Point 2	Point 3	Point 4
Robot end-effector ( $X'$ )	x	-0.000005	0.000236	-0.069540	0.069309
	y	0.001074	0.079929	-0.040248	-0.040756
	z	0.052548	-0.019033	-0.016784	-0.016731
Work piece ( $Y'$ )	x	-0.000208	0.000279	-0.069691	0.069620
	y	0.000818	0.079873	-0.040136	-0.040555
	z	0.052497	-0.018844	-0.017044	-0.016610

Based on the observed and local coordinates of points in Tables 3.1 and 3.2, the relative rotation matrix can be determined based on the conventional and strict LS solutions of  $R_{12}$  in Eqs. (3.46) and (3.49), as well as the conventional and strict CTLS solutions in Eqs. (3.63) and (3.67). As shown in Table 3.3, the difference can be noticed between the conventional and strict LS solutions and no difference for CTLS solutions can be observed. The difference of conventional and strict LS solutions can be calculated as

$$\delta_R = R_{12}(\hat{R}_{12})^{-1} - I_3 = \begin{bmatrix} -0.2319 & 0.0813 & -0.0963 \\ 0.0914 & -0.0931 & 0.0272 \\ -0.1696 & 0.0550 & -0.0754 \end{bmatrix} e^{-7}. \quad (3.71)$$

**Table 3.3: Relative Rotation Matrix in Experiments ( $R_{12}$ )**

Methods	Relative Rotation Matrix					
	Conventional (Indirect) Solutions ( $R_{12}$ )			Strict (Direct) Solutions ( $\hat{R}_{12}$ )		
LS	-0.30576275	-0.77385220	0.55468233	-0.30576276	-0.77385219	0.55468232
	-0.16915305	-0.52902878	-0.83143468	-0.16915303	-0.52902878	-0.83143467
	0.93696618	-0.34821710	0.03047248	0.93696616	-0.34821709	0.03047247
CTLS	-0.30575458	-0.77388833	0.55462689	-0.30575458	-0.77388833	0.55462689
	-0.16937745	-0.52902079	-0.83153369	-0.16937745	-0.52902079	-0.83153369
	0.93692338	-0.34818652	0.03067129	0.93692338	-0.34818652	0.03067129

Also, the LS solutions lost the orthogonality of rotation matrix, since the determinant of  $R_{12}$  is about 0.999 and the error of orthogonality is obtained as

$$\varepsilon_R = R_{12}(R_{12})^T - I_3 = \begin{bmatrix} 0.0106 & -0.0713 & -0.1182 \\ -0.0713 & -0.2322 & 0.3903 \\ -0.1182 & 0.3903 & 0.0893 \end{bmatrix} e^{-3}. \quad (3.72)$$

In order to check the accuracy of the LS and CTLS solutions of  $R_{12}$ , a simulation based on the experimental data is carried out by adding Gaussian noises to the nominal coordinates of points assuming the calculated CTLS solutions of pose transformation are nominal. The nominal pose transformations of  $T_{01}$ ,  $T_{02}$  and  $T_{12}$  are given by the CTLS solutions of  $\hat{T}_{01}$ ,  $\hat{T}_{02}$ , and  $\hat{T}_{12}$  as

$$\tilde{T}_{01} = \hat{T}_{01} = \begin{bmatrix} 0.8430 & 0.5377 & -0.0104 & -0.2981 \\ -0.4443 & 0.7072 & 0.5500 & 0.0772 \\ 0.3031 & -0.4590 & 0.8351 & -1.7218 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.73)$$

$$\tilde{T}_{02} = \hat{T}_{02} = \begin{bmatrix} -0.3586 & -0.9333 & 0.0201 & 0.1761 \\ 0.5313 & -0.2218 & -0.8176 & 0.3106 \\ 0.7675 & -0.2825 & 0.5754 & -1.9415 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.74)$$

$$\tilde{T}_{12} = \hat{T}_{12} = \begin{bmatrix} -0.3058 & -0.7739 & 0.5546 & 0.2295 \\ -0.1694 & -0.5290 & -0.8315 & 0.5209 \\ 0.9369 & -0.3482 & 0.0307 & -0.0601 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.75)$$

Thus, the contaminated coordinates of points with respect to the measurement system can be simulated by Eqs. (3.76) and (3.77). Tables 3.4 and 3.5 demonstrates the simulated coordinates of points with Gaussian noises, as opposed to the experimental data in Tables 3.1 and 3.2.

$$\begin{bmatrix} \hat{X} \\ 1 \end{bmatrix} = \hat{T}_{01} \begin{bmatrix} X' + e_{x'} \\ 1 \end{bmatrix} + \begin{bmatrix} e_x \\ 0 \end{bmatrix} \quad (3.76)$$

$$\begin{bmatrix} \hat{Y} \\ 1 \end{bmatrix} = \hat{T}_{02} \begin{bmatrix} Y' + e_{y'} \\ 1 \end{bmatrix} + \begin{bmatrix} e_y \\ 0 \end{bmatrix} \quad (3.77)$$

where the Gaussian noises of point measurements are assumed to be isotropic and homogeneous and subject to

$$e_{x'}, e_x, e_{y'}, \text{ and } e_y \sim \mathcal{N}(0, 1e^{-3}). \quad (3.78)$$

**Table 3.4:** Simulated Coordinates of Points w.r.t. the Measurement Frame

Bodies	Simulated Coordinates of Points (unit: $m$ )				
		Point 1	Point 2	Point 3	Point 4
Robot end-effector ( $\hat{X}$ )	x	-0.299487	-0.255640	-0.379557	-0.261125
	y	0.105938	0.122051	0.070346	0.008379
	z	-1.679709	-1.776592	-1.741395	-1.696640
Work piece ( $\hat{Y}$ )	x	0.178792	0.098922	0.240034	0.189335
	y	0.266327	0.306103	0.295649	0.371958
	z	-1.912906	-1.973721	-1.992492	-1.886938

**Table 3.5:** Simulated Local Coordinates of Points w.r.t. the Body-Fixed Frames

Bodies	Simulated Local Coordinates of Points (unit: $m$ )				
		Point 1	Point 2	Point 3	Point 4
Robot end-effector ( $\hat{X}'$ )	x	-0.001320	0.000193	-0.069476	0.069657
	y	0.000658	0.080512	-0.039648	-0.040938
	z	0.053773	-0.020040	-0.018145	-0.017671
Work piece ( $\hat{Y}'$ )	x	-0.000114	0.000236	-0.070122	0.069996
	y	0.000440	0.080834	-0.041764	-0.040782
	z	0.051014	-0.017106	-0.016877	-0.017759

Based on the simulated observed coordinates of  $\hat{X}$  and  $\hat{Y}$  in Table 3.4 and the simulated local coordinates of  $\hat{X}'$  and  $\hat{Y}'$  in Table 3.5, the relative rotation matrices ( $R_{12}$ ) can be determined in Table 3.6.

**Table 3.6:** Relative Rotation Matrix in Simulation ( $R_{12}$ )

Methods	Relative Rotation Matrix					
	Conventional (Indirect) Solutions ( $R_{12}$ )			Strict (Direct) Solutions ( $\hat{R}_{12}$ )		
LS	-0.31229	-0.76069	0.56700	-0.31216	-0.76065	0.56688
	-0.16128	-0.54042	-0.83075	-0.16132	-0.54043	-0.83069
	0.94755	-0.34769	0.00274	0.94746	-0.34769	0.00280
CTLS	-0.30035	-0.77032	0.56249	-0.30035	-0.77032	0.56249
	-0.17525	-0.53511	-0.82640	-0.17525	-0.53511	-0.82640
	0.93759	-0.34679	0.02572	0.93759	-0.34679	0.02572

Compared with the nominal rotation matrix ( $\tilde{R}_{12}$ ) in  $\tilde{T}_{12}$ , the accuracy of LS and CTLS

solutions of  $R_{12}$  in Table 3.6 can be evaluated by

$$\delta_R = \left\| R_{12}(\tilde{R}_{12})^{-1} - I_3 \right\|_2 \quad (3.79)$$

where

$$\tilde{R}_{12} = \begin{bmatrix} -0.3058 & -0.7739 & 0.5546 \\ -0.1694 & -0.5290 & -0.8315 \\ 0.9369 & -0.3482 & 0.0307 \end{bmatrix}.$$

As a result, the errors of the conventional (indirect) and strict (direct) LS solutions are different, i.e., 0.0330 and 0.0328, respectively. The errors of the conventional (indirect) and strict (direct) CTLS solutions are equivalent, which is 0.0107. Apparently, the CTLS solutions has the best accuracy among these solutions besides the fact that it also maintains the orthogonality of a rotation matrix. Therefore, both the simulation and experiments have demonstrated the inequality of conventional and strict LS solutions, and the equivalence of conventional and strict CTLS solutions. It indicates that the relative pose transformation should be solved by either the conventional or the strict CTLS solutions.

### 3.3.6 Summary

In this section, the conventional (indirect) and strict (direct) closed-form solutions of the least-square relative pose estimation were presented. To find the relative pose transformation between two rigid bodies from four sets of corresponding point measurements, two approaches have been developed and compared, namely, conventional (indirect) and strict (direct) approaches. For each approach, the ordinary least-square (LS) and constrained total least square (CTLS) have been applied to solve the optimal solutions of the relative pose transformation. From the comparison, the

inequality of conventional and strict LS solutions, and the equivalence of conventional and strict CTLS solutions have been theoretically proved and verified by both simulations and experiments. The comparison indicates that the direct solution with constraints lead to identical results of the conventional solution. However, without constraints, the direct solution yields better accuracy compared with the conventional solution. The reason is that CTLS method takes into account the orthogonality of a rotation matrix, i.e., the inverse equals to its transpose. Hence, the relative pose transformation between two rigid bodies should be solved by either the conventional or the strict CTLS solutions. It is safe and acceptable to employ the conventional CTLS approach of relative pose estimation.

Moreover, the number of observed points on the end-effector and the work piece might have different influences on the estimation accuracy of  $R_{12}$ . For example, it might be true that points on the robot end-effector have more influence than those on the work piece. Therefore, an analysis of influence of number of points on the relative pose estimation accuracy may need to be investigated. Also, a more interesting question would be what is the optimal combination of point numbers to have the most accurate result and yet still cost-effective? Table 3.7 demonstrates an example of the combination of point numbers. The objective is to find an optimal combination of numbers of points if existing. Eggert *et. al.* [35] stated that as the number of points grows, the error in the computed pose transformation approaches a value dependent on the noise level.

**Table 3.7:** Combination of point numbers

$n/k$	3	4	5
3	(3,3)	(3,4)	(3,5)
4	(4,3)	(4,4)	(4,5)
5	(5,3)	(5,4)	(5,5)

### **3.4 Pose Estimation for Multiple Frames of a Rigid-Body System**

This section presents pose estimation for multiple frames of a rigid-body system. For localization of a robotic riveting system, pose transformations need to be estimated between a number of coordinate frames. Such a robotic system consists of both single and articulated rigid bodies. The large numbers and different types of frames add complexity into the procedure of localization. For this reason, the approach is designed to generate corresponding point measurements with respect to each pair of two frames, based on which pose transformation between these two frames can be estimated based on a constrained total least-square method. Lastly, experiments are provided to validate the effective implementation of pose estimation among various frames of a robotic riveting system.

#### **3.4.1 Description of Localization**

The localization is to find the location of an object in an environment, specifically pose transformation of a body-fixed frame with respect to a reference frame, including a rotation matrix and a translational vector. When a sensing system is utilized for localization, the underlying problem is to find the pose transformation with respect to the sensor's frame. The parameters of pose transformation can be estimated with a CTLS criterion from redundant measurements. More specifically for a position measuring system, redundant measurements are spatial coordinates of multiple observable point features, which are known as targets rigidly attached on the bodies. More specifically for a position sensing system, measurements are spatial coordinates of multiple points, which are known as targets rigidly attached on the bodies.

In this section, the methodology of pose estimation is implemented in the localization of a robotic percussive riveting system. The localization needs to be performed for the robot holding a rivet gun and a working panel on the gantry, i.e.,  $T_m^r$  and  $T_m^p$  in Eq. (1.1). By employing a position measuring system, the goal is to estimate the pose transformations between frames of the robot base, TCP of the rivet gun, the working panel, and the measuring system based on point measurements. Once established, the online measurements of the target on the robot tooling and the working panel could be transformed into the robot's frame using Eq. (1.2). Also, the coordinates of points of interests on the work piece could be transformed into the robot's frame for manipulation purpose using Eq. (1.3).

It is obvious that the robotic riveting system consists of single and articulated rigid bodies. Single rigid bodies include the working panel, the position measuring system, and rigid body targets; while the articulated rigid body indicates the robot manipulator. Thus, two categories of frames are classified, namely, frames of single rigid bodies with multiple observable point features, and frames of articulated rigid bodies with only one point, i.e., the TCP of the robot. The large numbers and different categories of frames add complexity into the procedure of localization. Basically, the approach is to generate several corresponding point measurements with respect to each pair of two frames, based on which pose transformation between these two frames can be estimated based on the CTLS method using the SVD solution.

### **3.4.3 Localization of Robotic Riveting System**

Prior to the pose estimation, two sets of corresponding points need to be generated for

frames of both single and articulated rigid-bodies. For a single rigid body with multiple points, the generating approach is straightforward. One set is the local coordinates of points with respect to the body-fixed frame, while another set is the measured coordinates with respect to the sensing frame. For an articulated rigid-body system with only one point, the approach is to move the robot to multiple poses and then measure the coordinate of the TCP. The purpose is to create multiple point measurements at these locations as one set of points, while the corresponding local coordinates of the point are also recorded as another set. In this sense, a virtual target with multiple points is created. Thus, based on the methodology of point generation and the relative pose estimation in the previous section, the details of the implementation in the robotic riveting system are developed as follows, including the working panel, robot, and target localization.

#### **3.4.3.1 Working Panel Localization**

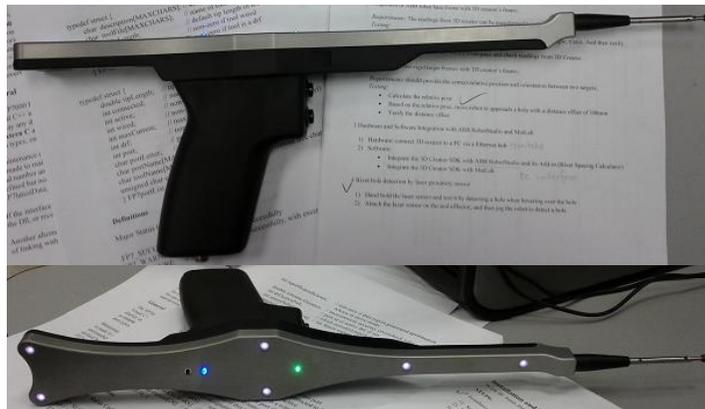
The working panel localization is to determine the pose transformation of the frame of the working panel with respect to the frame of the position measuring system, denoted by  $T_m^p$ . Three tooling balls with known geometry ( $1/2" \pm 0.0002"$ ) are installed at three predefined locations of the working panel, as shown in Figure 3.6. This is a case of a single rigid body with multiple points. Thus, spherical surface digitization of the tooling balls can be conducted with a wireless handheld probe and the position measuring system, as shown in Figure 3.7. There are multiple LEDs on the probe, which provide a reference to calculate the 3D coordinate of tip of the probe. Then, the techniques of sphere fitting can be applied to estimate the central coordinates of the tooling balls with respect to the position measuring system. Based on these two sets of point coordinates of tooling balls and transformation equation in Eq. (3.80), the aforementioned CTLS

solution of the pose estimation can be applied to find the optimal pose transformation between the frame of the working panel and the frame of the position measurement system.

$$\begin{bmatrix} p_i \\ 1 \end{bmatrix} = T_m^p \begin{bmatrix} p'_i \\ 1 \end{bmatrix} \quad (3.80)$$

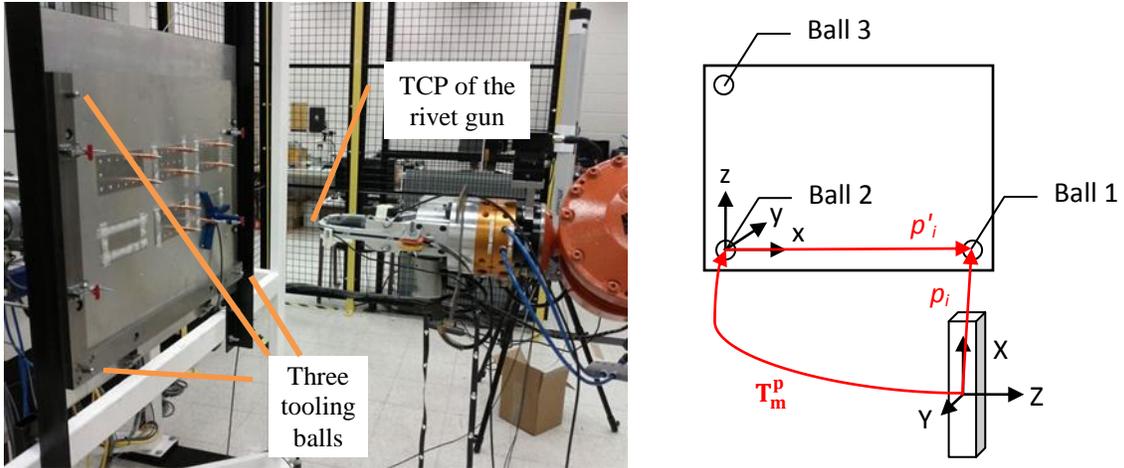


**Figure 3.6:** Three tooling balls.

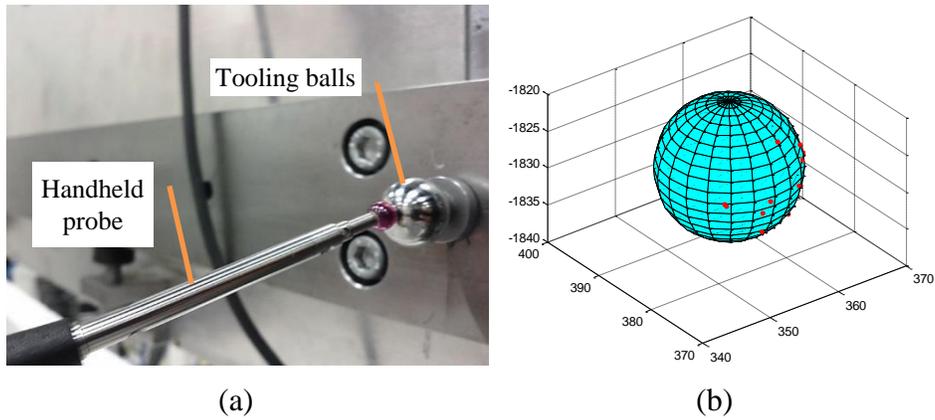


**Figure 3.7:** Wireless handheld probe for digitization.

Figure 3.8 shows the three tooling balls on the corners of the working panel and the TCP of the rivet gun. The surface digitization of tooling balls has been done with a handheld probe by touching the multiple locations of the surface, as shown in Figure 3.9(a). Sphere fitting was then employed to estimate the central coordinates of the tooling balls, as shown in Figure 3.9(b).



**Figure 3.8:** Panel localization using three tooling balls, and TCP of the rivet gun.



**Figure 3.9:** (a) Digitization of tooling balls with a handheld probe; (b) the fitted sphere.

**Table 3.8:** Central coordinates of tooling balls in the sensor and panel frames.

	$p_i$ - Measured coordinates in the sensor frame (mm)			$p'_i$ - Coordinates in the working panel frame (mm)		
	X	Y	Z	X	Y	Z
Tooling ball 1	357.97	388.437	-1830.798	670	0	0
Tooling ball 2	368.958	3.683	-2379.214	0	0	0
Tooling ball 3	-120.915	-5.52	-2382.621	0	0	490

By comparing the actual dimensions, the diameter errors of the fitted spheres for three tooling balls were  $[0.0494 \ 0.0023 \ 0.0488]^T$  mm. The errors of the horizontal and vertical distance between three tooling balls were 0.0129 mm and 0.0290 mm. It proves the precision of the sensing system. Therefore, the fitted central coordinates and the local

coordinates of tooling balls were gathered and shown in Table 3.8. Based on these two sets of coordinates, the pose transformation from the measuring system frame to the working panel frame can be calculated by the CTLS techniques,

$$T_p^m = \begin{bmatrix} -0.016448 & 0.5742488 & 0.8185156 & 1951.378 \\ -0.011381 & 0.8184657 & -0.5744426 & -1365.537 \\ -0.999800 & -0.0187638 & -0.0069269 & 352.493 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

### 3.4.3.2 Robot Localization

For robot localization, the goal is to find the relative pose transformation between the robot base and the working panel, based on which the local coordinates of rivet holes on the working panel from the rivet pattern planning can be transformed to the robot base frame for robotic path planning. To do so, the pose transformation from the robot to the measuring system has to be determined first, denoted by  $T_m^r$ . Then, the relative pose transformation from the panel to the robot, denoted by  $T_r^p$ , can be calculated by Eq. (1.1).

As the first step of estimating  $T_m^r$ , tooling geometric parameters have to be determined for robotic TCP calculation, as shown in Figure 3.10. Specifically, the least-square techniques of plane fitting of end-effector flange and calculation of Euclidean distance from the TCP to the plane can be applied to find the tooling geometric parameter, i.e., the distance from the TCP to the flange surface of the robot end-effector by assuming no coaxiality error of the TCP with respect to the end-effector. Without the assumption, the coaxial errors can be found based on cylinder surface fitting. More specifically, principle component analysis (PCA) can be adopted as a solution of the plane fitting when finding the normal vector of the plane. The result was 369.0433 mm. Note that there was an offset of 2.5 mm considering the radius of the probe ball.



(a)

(b)

**Figure 3.10:** Determination of tool length from TCP to end-effector flange, (a) digitizing the surface of the flange of the end-effector; (b) digitizing the TCP of the rivet gun.

Later, the length value was inputted into ABB robot's tool database for tool length compensation, as shown in Figure 3.11(a). In our case, the rivet gun tooling length of  $369.043\text{mm}$  was along the  $z$  axis of the end-effector frame and assumed that the tooling was coaxial with the robot end-effector. The assumption was reasonable based on the design of the tooling.



(a)

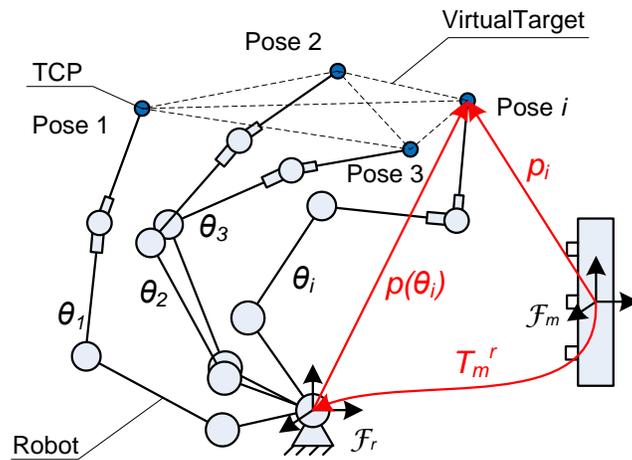
(b)

**Figure 3.11:** (a) Screenshot of tool definition for the rivet gun in the ABB FlexPendant; (b) Pose of TCP of the rivet gun with respect to the robot base frame ( $T_r^{tcp}$ ).

The second step is to create a virtual target at the robot TCP of multiple poses in the workspace, as shown in Figure 3.12. Therefore, two sets of corresponding point sets can be generated for this articulated rigid-body system. The approach is to measure the

coordinates of the TCP by the handheld probe and the position measuring system at various locations when the robot is jogged in the workspace, denoted as  $p_i$ . In the meantime, the coordinates of the TCP with respect to the frame of the robot base are recorded at these poses from the robot controller or the ABB FlexPendant, as shown in Figure 3.11(b), denoted as  $p(\theta_i)$ .  $\theta_i$  denotes various joint angles of the robot.

$$\begin{bmatrix} p_i \\ 1 \end{bmatrix} = T_m^r \begin{bmatrix} p(\theta_i) \\ 1 \end{bmatrix} \quad (3.81)$$



**Figure 3.12:** Robot localization by creating a virtual target at multiple poses in workspace.

Similarly, the aforementioned CTLS solution of the pose transformation can be applied to solve for an optimal estimation of pose transformation of  $T_m^r$  from Eq. (3.81). It should be noted that at least three locations are required and the more numbers of locations spanning in the workspace the less bias of estimation. The selection of locations should be carefully considered in terms of well representing the working volume, such as the riveting operating volume with respect to the working panel. These two sets of data are shown in Table 3.9. Based on them, the pose transformation from the measuring system frame to the robot base frame can be determined by the CTLS techniques as,

$$T_r^m = \begin{bmatrix} 0.004821 & 0.823075 & -0.567912 & 288.109 \\ 0.018174 & -0.567897 & -0.822899 & -1536.275 \\ -0.999823 & -0.006354 & -0.017696 & 1338.163 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Finally, the pose transformation matrix from the panel to the robot can be calculated as

$$T_r^p = \begin{bmatrix} 0.007726 & 0.999837 & -0.01633 & 1644.103 \\ -0.999969 & 0.007698 & -0.001814 & 426.193 \\ -0.001688 & 0.016344 & 0.999865 & 1011.329 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

**Table 3.9:** Coordinates of the TCP in the sensor and robot base frames.

	$p_i$ - Measured TCP coordinates in the sensor frame (mm)			$p(\theta_i)$ - TCP in the robot base frame (mm)		
	X	Y	Z	X	Y	Z
Pose 1	69.825	-316.292	-2059.047	1197.4	339.2	1307.2
Pose 2	-87.23	-334.008	-2146.817	1232.1	418.6	1465.7
Pose 3	214.587	126.688	-1706.057	1362.3	-200.4	1152.6
Pose 4	337.598	7.229	-1973.685	1417.1	90.3	1035
Pose 5	-8.611	-84.016	-1964.318	1334.6	127.8	1382.3
Pose 6	105.71	-103.454	-2023.266	1352.4	189	1269.2
Pose 7	251.214	-163.125	-2138.181	1369.9	319.9	1125.6
Pose 8	83.427	118.864	-1903.497	1467.3	-36.1	1287.6
Pose 9	48.965	-105.784	-2110.403	1399.6	261.6	1327.4
Pose 10	193.107	-48.501	-2070.071	1424.5	197.9	1182.2
Pose 11	158.138	114.124	-1912.339	1468.7	-24.2	1213.1
Pose 12	127.778	-50.006	-2208.214	1501.2	311.5	1249.6
Pose 13	25.521	-114.313	-1998.223	1328.9	173.5	1348.8

### 3.4.3.3 Target Localization

A rigid body target with observable point features is rigidly attached on the tooling of the robot to provide a tracking reference. Target localization is to find the location of the target 2 with respect to the end-effector of the robot. The target is usually equipped with multiple passive or active markers, such as reflective tapes and LEDs. This is also a case of a single rigid body with multiple points. Similarly, from these two sets of local coordinates and the measured coordinates of the points of the target 2, a pose transformation from the target to the measuring system, denoted by  $T_m^{t2}$ , can be estimated

based on the aforementioned CTLS technique of the pose estimation based on

$$\begin{bmatrix} p_{t2,i} \\ 1 \end{bmatrix} = T_m^{t2} \begin{bmatrix} p'_{t2,i} \\ 1 \end{bmatrix}. \quad (3.82)$$

Thus, the pose of the target 2 with respect to the end-effector can be derived by

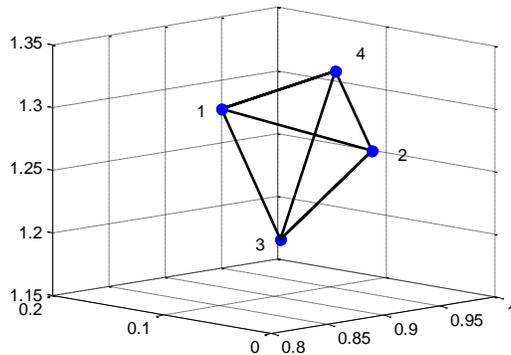
$$T_e^{t2} = (T_r^e)^{-1} * (T_m^r)^{-1} * T_m^{t2}. \quad (3.83)$$

For target localization, a target with four infrared LEDs called a dynamic rigid frame (DRF) is rigidly attached on the tooling, as shown in Figure 3.13. The local coordinates and the measured coordinates of the four LEDs of the target are shown in Table 3.10. Similarly, the pose transformation from the DRF target to the end-effector of the robot can be calculated by the CTLS techniques,

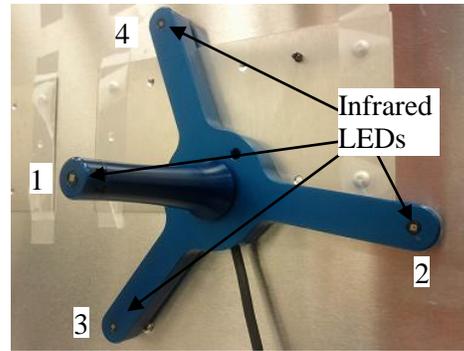
$$T_e^{t2} = \begin{bmatrix} 0.8395696 & 0.5432025 & 0.0073472 & 56.0120374 \\ 0.0159585 & -0.0111422 & -0.9998106 & -164.2331801 \\ -0.5430177 & 0.8395278 & -0.0180233 & 105.6985580 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

**Table 3.10:** LED coordinates of the target in the sensor and local frames.

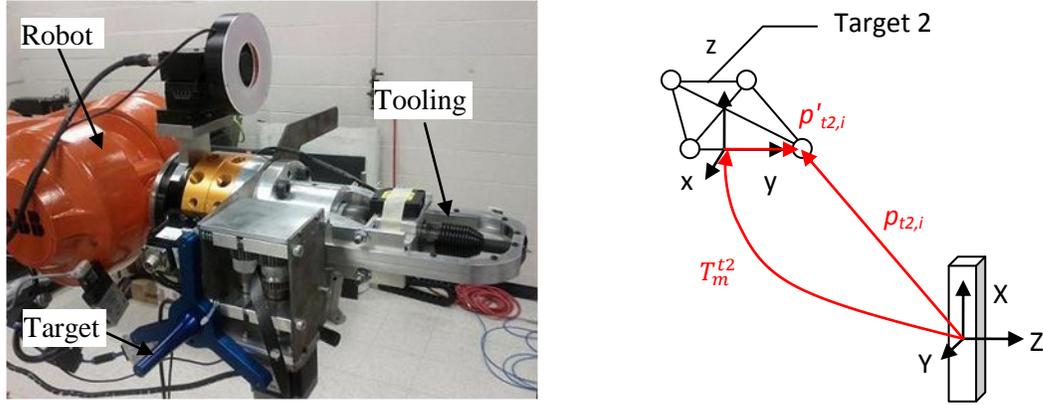
	$p_{t2,i}$ - Measured coordinates in the sensor frame (mm)			$p'_{t2,i}$ - Coordinates in the target frame (mm)		
	X	Y	Z	X	Y	Z
LED 1	40.38	-243.43	-1423.66	-0.0051	1.0743	52.5481
LED 2	84.73	-228.00	-1519.27	0.2362	79.9293	-19.0332
LED 3	-39.63	-279.13	-1484.19	-69.54.4	-40.2479	-16.7839
LED 4	76.02	-342.36	-1440.63	69.3093	-40.7557	-16.7310



(a)



(b)

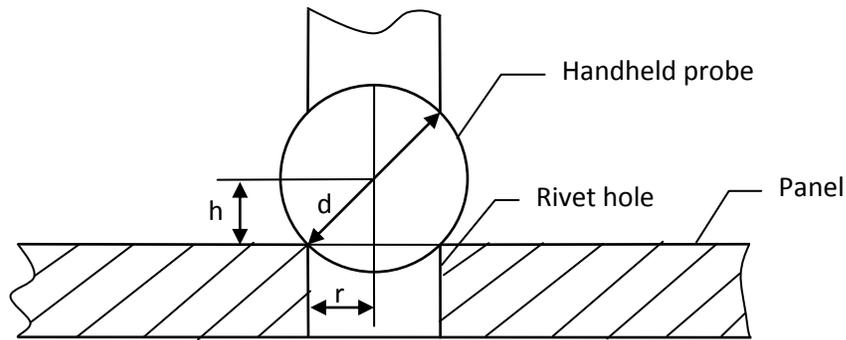


(c)

**Figure 3.13:** Target 2. (a) Modeling (b) Four infrared LEDs. (c) The target on the tooling and transformation between frames of target 2 and measurement.

### 3.4.4 Verification of Localization

For verification of the accuracy of the localization, a procedure was designed as follows. First, a random rivet spot was chosen on the working panel and then digitized the spot with the handheld probe, as depicted in Fig. 3.14. The coordinates were denoted as  $p'_m$ .



**Figure 3.14:** Hole digitization with handheld probe.

Second, the digitized coordinates of the spot were transformed into the robot base frame by

$$p_r = T_r^m * p'_m. \quad (3.84)$$

Third, the transformed coordinate was defined as a robot target and programmed in a

motion command in the ABB robot. Lastly, the command was executed and checked if the robot TCP was actually positioned at the rivet spot. Then, this procedure was repeated for several random spots on the working panel to ensure the results.

It was found that the robot was capable of reaching the digitized rivet spots after the proper localization with an accuracy of 1~2 *mm*, approximately, which depends on the robot absolute positioning accuracy and localization accuracy. Also, the techniques of the localization with the position sensing system along with the digitization using a probe can be integrated as a robot programming-by-demonstration to replace the conventional programming-by-teaching. The work in this area is ongoing to further analyze the accuracy of estimation and the influence of the number of points on the accuracy.

### **3.4.5 Summary**

The pose estimation for multiple frames of a rigid-body system is presented in this section. Two categories of frames are classified, namely, frames of single rigid bodies with multiple observable point features, and frames of articulated rigid bodies with one point. The approach is to generate several corresponding point measurements with respect to each pair of two frames, based on which pose transformation between these two frames can be estimated based on a CTLS method using the SVD. Localization using a virtual target with multiple points is proposed for articulated body systems. The methodology has been implemented for the localization of the robotic riveting system. Pose transformations were estimated between frames at robot base, TCP, working panel, rigid body target, and a position measuring system. Clearly, the method can be extended to other rigid-body systems with single rigid bodies and articulated rigid bodies.

## **CHAPTER 4 DYNAMIC POSE ESTIMATION**

This chapter addresses a novel method of dynamically estimating relative pose between two rigid bodies during motion for position-based visual servoing. The conventional way is to estimate pose of each body individually relative to measurement. The proposed approach is to dynamically estimate relative pose of two bodies simultaneously. For robotic riveting, the objective is for dynamic pose estimation between the robot TCP and working panel, as given in Eq. (1.4). For this reason, when directly assigning the relative pose and motion as a state estimate, the observation model is formulated in such a way that the state estimate is mapped to the observed point sets of two bodies. Since this observation model is nonlinear, an analytical expression of linearization is derived and state estimation by iterative extended Kalman filters (IEKF) is adapted to reduce the linearization errors. The proposed approach is implemented in simulations of position-based visual servoing and validated experimentally on a robotic riveting system for aircraft automated assembly.

### **4.1 Description of Dynamic Pose Estimation**

A relative pose transformation is defined between two rigid body's frames. Instead of tracking each body individually, the work presented in this section addresses the state estimation of relative poses between two rigid bodies simultaneously during motion based on noisy geometric measurements. For measurements, geometric features of rigid bodies can be observed such as points, lines, planes, etc. Among them, point features are the most effective due to easier implementation for processing. In order to measure 3D coordinates of point features in high frame rates for real-time control, a portable optical

CMM is utilized by tracking targets of synchronized active LED point markers attached on rigid bodies [19, 69]. It avoids conventional time-consuming processing of computer vision in planar CCD cameras, such as image formation, processing, feature extraction, matching, and registration, etc.

In visual servoing, the existing relevant research [21, 22] mainly focused on pose estimation of a rigid body relative to a camera, regardless of configurations of cameras. When a camera is attached to an end-effector of a robot, they provide estimation relative to the camera or the end-effector [14]. When cameras are fixed as standalone devices, they provide estimation relative to cameras. Due to the large size of an optical CMM and the requirement of good robotic tooling accessibility [3], the CMM needs to be fixed outside of the robot's operational space. Specifically for aircraft robotic structural assembly, when one target of point markers is fixed on the robotic tooling and another one on the working aero-structural panel, the underlying problem becomes to estimate the state of relative pose and motion between the robotic tooling and the working panel simultaneously based on point measurements of targets.

Therefore, Kalman filtering technique is applied to estimate sequentially the state of a dynamic system using a sequence of noisy point measurements from the optical CMM. It captures both rotational and translational parameters together as a state vector. When directly assigning the relative pose and motion as a state estimate, the observation model is formulated in such a way that the state estimate is implicitly mapped to the observed 3D coordinates of points on the bodies. Since this observation model is nonlinear, an analytical expression of linearization is derived. As mentioned in Section 2.2.2, the recent development of AEKF mainly targets noise covariance adaptability due to changing

velocities and accelerations of tracked objects, and the measurement vulnerability of traditional planar CCD cameras when illumination condition changes, etc [21, 22]. However, there is no such need for an optical CMM using active targets and synchronized acquisition, which has the advantage of being not sensitive to external light conditions [70]. Furthermore, the velocities and accelerations of industrial robot are normally kept to be constant. Therefore, the paper adopts IEKF for the state estimation of relative pose from point measurements provided by an optical CMM.

In this section, a method of relative pose estimation from four sets of corresponding points based on IEKF is presented during the motion of two rigid bodies. The objective is to simultaneously estimate the relative pose between two rigid bodies using point measurements provided by an optical CMM. By formulating the observation model to map the state of relative pose to the measurements of points, this method applies IEKF as a state estimator. The effectiveness of this method is demonstrated through a case study of a robotic riveting system for aircraft automated assembly.

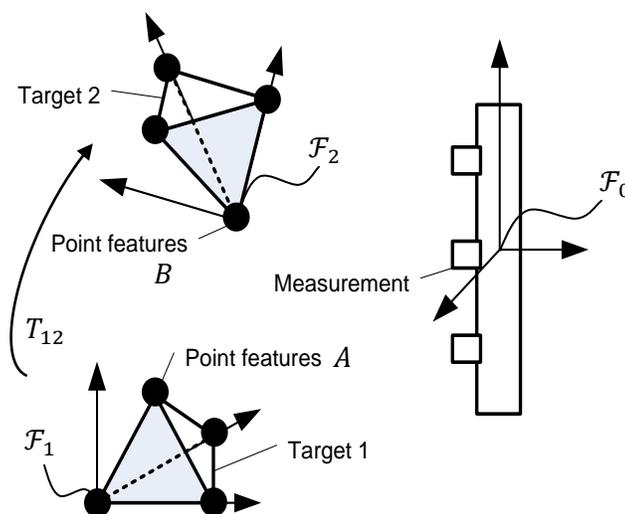
## 4.2 Problem Statement

As shown in Figure 4.1, two rigid body targets with certain numbers of point features are observed by a position measurement system of optoelectronic CMM during motion. Based on the observed points, the underlying problem is to dynamically estimate the relative pose and motion between these two rigid bodies.

Thus, the problem can be stated mathematically as follows:

Let  $A(k) = [a_1(k), a_2(k), \dots, a_n(k)] \in \mathbb{R}^{m \times n}$  and  $A' = [a_1', a_2', \dots, a_n'] \in \mathbb{R}^{m \times n}$  be two

sets of corresponding non-coplanar points for a rigid body (Target 1) in  $m$ -dimensional space at a discrete time step of  $k$ ;  $B(k) = [b_1(k), b_2(k), \dots, b_l(k)] \in \mathbb{R}^{m \times l}$  and  $B' = [b_1', b_2', \dots, b_l'] \in \mathbb{R}^{m \times l}$  be another two sets of corresponding non-coplanar points for another rigid body (Target 2) in  $m$ -dimensional space at discrete time of  $k$ . Note that we focus on 3-dimensional cases, i.e.,  $m = 3$ .  $n$  and  $l$  are the numbers of points on Targets 1 and 2, respectively.  $A(k)$  and  $B(k)$  are the observed sample  $k$  of the coordinates of points with respect to a common measurement frame, denoted as  $\mathcal{F}_0$ ;  $A'$  and  $B'$  are the local coordinates of points with respect to the body-fixed frames of Targets 1 and 2, denoted as  $\mathcal{F}_1$  and  $\mathcal{F}_2$ , respectively.



**Figure 4.1:** Two rigid body targets observed by a position measurement system.

The objective is to find the state estimate of relative pose transformation of frame  $\mathcal{F}_2$  with respect to frame  $\mathcal{F}_1$  and its derivative with respect to time during motion, by minimizing the variance of estimation errors caused by the Gaussian noises of the point data of  $A(k)$  and  $B(k)$ . Suppose the Gaussian noises of point measurements are isotropic (equal distribution in all directions) and homogeneous (equal for all points in space). The relative pose transformation can be defined as a homogeneous transformation matrix, i.e.

$$T_{12} = \begin{bmatrix} R_{12} & t_{12} \\ 0 & 1 \end{bmatrix}, \quad (4.1)$$

where  $T_{12} \in \mathbb{R}^{(m+1) \times (m+1)}$ ,  $R_{12} \in \mathbb{R}^{m \times m} | R^T R = I_m, \det(R) = +1$  and  $t_{12} \in \mathbb{R}^{m \times 1}$  are orthonormal rotation matrix and translation vector, respectively. As for the orientation representation, Euler angles with a rotation sequence of Z-Y-X axes are adapted instead of the rotation matrix. Thus, the relative pose transformation in Eq. (4.1) can be represented as a relative pose vector,

$$X_1 = (\phi_1^T \quad \phi_2^T)^T = (x \quad y \quad z \quad \alpha \quad \beta \quad \gamma)^T, \quad (4.2)$$

where  $X_1 \in \mathbb{R}^{2m \times 1}$ ,  $\phi_1 = (x \quad y \quad z)^T = t_{12}$ ;  $\phi_2 = (\alpha \quad \beta \quad \gamma)^T$  are the Euler angles for  $R_{12}$ , then let  $R_{12}: \phi_2 \rightarrow \mathbb{R}^{m \times m}$  be a matrix function defined on a set of  $\phi_2$  in  $\mathbb{R}^{m \times 1}$ , i.e.,

$$R_{12} = \rho(\phi_2) = Rot_z(\alpha)Rot_y(\beta)Rot_x(\gamma), \quad (4.3)$$

where

$$Rot_z(\alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.4)$$

$$Rot_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \quad (4.5)$$

$$Rot_x(\gamma) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma) & -\sin(\gamma) \\ 0 & \sin(\gamma) & \cos(\gamma) \end{bmatrix}. \quad (4.6)$$

Thus, from Eq. (4.1) and (4.2), let  $T_{12}: X_1 \rightarrow \mathbb{R}^{(m+1) \times (m+1)}$  be a matrix function defined on a set of  $X_1$  in  $\mathbb{R}^{2m \times 1}$ , i.e.,

$$T_{12} = f(X_1). \quad (4.7)$$

To directly map the relative pose transformation of  $T_{12}$  to the measured points ( $B$ ) of Target 2, the transform mapping of local points of  $B'$  from its description in frame  $\mathcal{F}_2$  to a

description in frame  $\mathcal{F}_1$  can be described in terms of  $T_{12}$  as

$$T_{10} \begin{bmatrix} B \\ 1 \end{bmatrix} = T_{12} \begin{bmatrix} B' \\ 1 \end{bmatrix}, \quad (4.8)$$

where  $T_{10}$  is the transform mapping from  $\mathcal{F}_0$  to  $\mathcal{F}_1$ , defined as

$$T_{10} = \begin{bmatrix} R_{10} & t_{10} \\ 0 & 1 \end{bmatrix}, \quad (4.9)$$

which can be characterized in the following transform mapping of the measured points ( $A$ ) of Target 1 from  $\mathcal{F}_0$  to  $\mathcal{F}_1$ ,

$$\begin{bmatrix} A' \\ 1 \end{bmatrix} = \begin{bmatrix} R_{10} & t_{10} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} A \\ 1 \end{bmatrix}. \quad (4.10)$$

Based on the problem statement, the following is to investigate the methodology of dynamically estimating the relative pose and motion from these four corresponding point sets during motion.

### 4.3 Relative Pose and Motion Estimation

First, the problem is further modeled as discrete time-varying non-linear systems in state space domain, and the relative pose vector and its time derivative are both captured as a state vector. Essentially, the nearly constant velocity model is assumed for the state transition during motion. Moreover, the observation model is formulated to directly map the state to the point measurements. Second, the IEKF is applied to find the suboptimal estimation of the state based on the non-linear systems. An analytical expression of linearization is derived for the non-linear observation model and the iterative scheme of IEKF minimizes the linearization errors.

### 4.3.1 State Space Modeling

When the relative pose vector in Eq. (4.2) and its derivative with respect to time are chosen as the state variables, the problem can be described as discrete time-varying non-linear stochastic systems in state space domain, i.e.,

$$X(k+1) = FX(k) + G(k)u(k) + \Gamma v \quad (4.11)$$

$$Y(k) = h(X(k)) + Mw, \quad (4.12)$$

where  $X(k) \in \mathbb{R}^{4m \times 1}$  is the state variables, defined as

$$X(k) = [X_1(k) \quad X_2(k)]^T = [X_1(k) \quad \dot{X}_1(k)]^T, \quad (4.13)$$

and  $\dot{X}_1(k) \in \mathbb{R}^{2m \times 1}$  is the time derivative of  $X_1(k)$  in Eq. (4.2),  $F \in \mathbb{R}^{4m \times 4m}$  is a state transition matrix.  $u(k) \in \mathbb{R}^{2m \times 1}$  is a control input with a control gain of  $G(k) \in \mathbb{R}^{4m \times 2m}$ , detailed in the following section of implementation.  $Y(k) \in \mathbb{R}^{(m \times l) \times 1}$  is a measurement vector associated with measured points of  $B$ , and  $h(X(k))$  is a non-linear observation mapping model from the state of  $X(k)$  to the measurement of  $Y(k)$ . The unpredictable disturbances of  $v \in \mathbb{R}^{2m \times 1}$  and  $w \in \mathbb{R}^{(m \times l) \times 1}$  are assumed to be zero-mean white Gaussian noises and mutually independent. Furthermore,  $\Gamma \in \mathbb{R}^{4m \times 2m}$  and  $M \in \mathbb{R}^{(m \times l) \times (m \times l)}$  are the noise gains for the disturbances of  $v$  and  $w$ , respectively.

As for the state transition model in Eq. (4.11), a nearly constant velocity model is assumed during motion. Thus, the state transition matrix can be defined as

$$F = \begin{bmatrix} I_{2m \times 2m} & \Delta t I_{2m \times 2m} \\ 0 & I_{2m \times 2m} \end{bmatrix}, \quad (4.14)$$

where  $\Delta t$  is the sampling time interval. As mentioned, the acceleration is modeled as a zero-mean white Gaussian noise, i.e.,  $v = (v_1 \cdots v_{2m})^T$ , where  $v_i \sim \mathcal{N}(0, \sigma_v^2)$ ,

$E(v_i v_j') = 0$ , when  $i \neq j$ . Thus, the corresponding noise gain is given by [20]

$$\Gamma = \begin{bmatrix} \frac{1}{2} \Delta t^2 I_{2m \times 2m} \\ \Delta t I_{2m \times 2m} \end{bmatrix}, \quad (4.15)$$

and the covariance of  $\Gamma v$  can be calculated as

$$Q_v = E(\Gamma v_i, \Gamma v_j') = \Gamma E(v_i v_j') \Gamma' = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \sigma_v^2, \quad (4.16)$$

where  $Q_v \in \mathbb{R}^{4m \times 4m}$ , and  $Q_{11}, Q_{12}, Q_{21}, Q_{22} \in \mathbb{R}^{2m \times 2m}$ , given by

$$Q_{11} = \frac{1}{4} \Delta t^4 I_{2m \times 2m}, \quad Q_{12} = Q_{21} = \frac{1}{2} \Delta t^3 I_{2m \times 2m}, \quad Q_{22} = \Delta t^2 I_{2m \times 2m}. \quad (4.17)$$

For the observation model in Eq. (4.12), a nonlinear mapping from the state of the relative pose and motion to the measurements of  $B$  can be formulated in Eq. (4.18) and (4.19) based on the transform mapping equation in Eq. (4.7) and (4.8).

$$Y(k) = \begin{bmatrix} R_{10} b_1(k) + t_{10} \\ \vdots \\ R_{10} b_l(k) + t_{10} \end{bmatrix} \quad (4.18)$$

$$h(X(k)) = \begin{bmatrix} h_1(X_1(k), b'_1) \\ \vdots \\ h_l(X_1(k), b'_l) \end{bmatrix} = \begin{bmatrix} R_{12}(k) b'_1 + t_{12}(k) \\ \vdots \\ R_{12}(k) b'_l + t_{12}(k) \end{bmatrix}. \quad (4.19)$$

Further, the noise of the measurement is  $w \sim \mathcal{N}(0, \sigma_w^2)$ , i.e.,  $E(w_i w_j') = \sigma_w^2$ , depending on the precision of the position measuring system. Based on the assumption of isotropic and homogeneous noise, the covariance of  $Mw$  is denoted as

$$Q_w = E(Mw(i), Mw(j)') = ME(w_i w_j') M' = I_{(m \times l) \times (m \times l)} \sigma_w^2, \quad (4.20)$$

where the noise gain is given by

$$M = I_{(m \times l) \times (m \times l)}. \quad (4.21)$$

### 4.3.2 Relative Pose and Motion Estimation by IEKF

Based on the state space model in Eq. (4.11) and (4.12), the IEKF is applied to iteratively compute the mean of the posterior estimate of the state in Eq. (4.22) and then determine its corresponding covariance in Eq. (4.23) upon the convergence of the mean. Eq. (4.22)-(4.28) are iterative until successive state estimates  $\hat{X}^i(k, k)$  converge according to suitable criteria [20]. The number of iteration depends on the termination criteria. The final results of  $\hat{X}^i(k, k)$  is taken as the updated state estimate of  $\hat{X}(k, k)$ . Essentially, the scheme of the IEKF removes the effects of observation model nonlinearity.

$$\hat{X}^i(k, k) = \hat{X}(k, k-1) + K^{i-1}(k) \left[ e^{i-1}(k) - H^{i-1}(k) \left( \hat{X}(k, k-1) - \hat{X}^{i-1}(k, k) \right) \right] \quad (4.22)$$

$$P(k, k) = \left( I - K^i(k)H^i(k) \right) P(k, k-1), \quad (4.23)$$

where  $\hat{X}(k, k-1)$  and  $P(k, k-1)$  are the mean and covariance of the prior estimate of the state, predicted as

$$\hat{X}(k, k-1) = F\hat{X}(k-1, k-1) + G(k)u(k) + \Gamma v \quad (4.24)$$

$$P(k, k-1) = FP(k-1, k-1)F^T + Q_v. \quad (4.25)$$

In Eq. (4.22) and (4.23),  $K^{i-1}(k)$  is the *Kalman gain* using minimum mean-square error as the performance criterion, of which one form is given by

$$K^i(k) = P(k, k-1)H^i(k)^T \left[ H^i(k)P(k, k-1)H^i(k)^T + Q_w \right]^{-1}, \quad (4.26)$$

$e^i(k)$  is the measurement residual calculated from Eq. (4.18) and (4.19), i.e.

$$e^i(k) = Y(k) - h \left( \hat{X}^i(k, k) \right), \quad (4.27)$$

and  $H^i(k)$  is the linear approximation of  $h(X(k))$  in Eq. (4.19), derived as

$$\begin{aligned}
H^i(k) &= \begin{bmatrix} \left. \frac{\partial h_1(X, b'_1)}{\partial X} \right|_{X=\hat{X}^i(k,k)} \\ \vdots \\ \left. \frac{\partial h_l(X, b'_l)}{\partial X} \right|_{X=\hat{X}^i(k,k)} \end{bmatrix} = \begin{bmatrix} \frac{\partial h_1(\hat{X}^i(k,k), b'_1)}{\partial X_1} & \frac{\partial h_1(\hat{X}^i(k,k), b'_1)}{\partial X_2} \\ \vdots & \vdots \\ \frac{\partial h_l(\hat{X}^i(k,k), b'_l)}{\partial X_1} & \frac{\partial h_l(\hat{X}^i(k,k), b'_l)}{\partial X_2} \end{bmatrix} \\
&= \begin{bmatrix} \frac{\partial h_1(\hat{X}^i(k,k), b'_1)}{\partial X_1} & 0 \\ \vdots & \vdots \\ \frac{\partial h_l(\hat{X}^i(k,k), b'_l)}{\partial X_1} & 0 \end{bmatrix}. \tag{4.28}
\end{aligned}$$

Specifically, from the definition of  $X_1$  in Eq. (4.2) and the derivative of the pose transformation matrix in Lemma 2, it yields

$$\begin{aligned}
\frac{\partial h_j(\hat{X}^i(k,k), b'_j)}{\partial X_1} &= \begin{bmatrix} \frac{\partial h_j(\hat{X}^i(k,k), b'_j)}{\partial \phi_1} & \frac{\partial h_j(\hat{X}^i(k,k), b'_j)}{\partial \phi_2} \end{bmatrix} = [I_{m \times m} \quad \Psi \times R_{12}(k)b'_j] \\
&= [I_{m \times m} \quad (\widetilde{R_{12}(k)b'_j})^T \Psi], \quad \forall j \in [1, l], \tag{4.29}
\end{aligned}$$

where

$$\Psi \times R_{12}b' = -R_{12}b' \times \Psi = -\widetilde{R_{12}b'}\Psi = (\widetilde{R_{12}b'})^T \Psi. \tag{4.30}$$

**Lemma 2:** A homogeneous matrix of pose transformation is defined as  $T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$ . Let

$T: X \rightarrow \mathbb{R}^{(m+1) \times (m+1)}$  be a matrix function defined on a set of  $X$  in  $\mathbb{R}^{2m \times 1}$ , i.e.,  $T =$

$f(X)$ . Then, the partial derivative of  $T$  with respect to

$X = (\phi_1^T \quad \phi_2^T)^T = (x \quad y \quad z \quad \alpha \quad \beta \quad \gamma)^T$  can be derived as

$$\frac{\partial f}{\partial X} = \begin{bmatrix} \frac{\partial f}{\partial \phi_1} & \frac{\partial f}{\partial \phi_2} \end{bmatrix} = \begin{bmatrix} 0 & I_{m \times m} & \Psi \times R & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{4.31}$$

**Proof of lemma 2:** From the definition of  $R$  in Eq. (4.3),  $f(X)$  can be reformulated as

$$f(\phi_1, \phi_2) = \begin{bmatrix} \rho(\phi_2) & \phi_1 \\ 0 & 1 \end{bmatrix}. \quad (4.32)$$

Since  $\dot{R} = \Omega R$ , and  $\Omega$  is the angular velocity tensor defined as

$$\Omega = \tilde{\omega} = \omega \times = \dot{R}R^T, \quad (4.33)$$

where  $\tilde{\omega}$  is the skew-symmetric matrix of  $\omega$ , defined as

$$\tilde{\omega} = \begin{bmatrix} 0 & \omega_3 & -\omega_2 \\ -\omega_3 & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix}, \quad (4.34)$$

and the Jacobian of angular velocity in Lemma 3, the differential of  $R$  can be derived as

$$\dot{R} = \Omega R = \omega \times R = \Psi \dot{\phi}_2 \times R = -R \times \Psi \dot{\phi}_2. \quad (4.35)$$

Then, the derivative of  $R$  with respect to  $\phi_2$  is

$$\frac{d\rho}{d\phi_2} = -R \times \Psi = \Psi \times R. \quad (4.36)$$

Thus, the partial derivative of Eq. (4.32) is derived as

$$\frac{\partial f}{\partial X} = \begin{bmatrix} \frac{\partial f}{\partial \phi_1} & \frac{\partial f}{\partial \phi_2} \end{bmatrix} = \begin{bmatrix} 0 & I_{m \times m} & \Psi \times R & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad (4.37)$$

where

$$\frac{\partial f}{\partial \phi_1} = \begin{bmatrix} \frac{\partial \rho}{\partial \phi_1} & \frac{\partial \phi_1}{\partial \phi_1} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & I_{m \times m} \\ 0 & 0 \end{bmatrix} \quad (4.38)$$

$$\frac{\partial f}{\partial \phi_2} = \begin{bmatrix} \frac{\partial \rho}{\partial \phi_2} & \frac{\partial \phi_1}{\partial \phi_2} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{dR}{d\phi_2} & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \Psi \times R & 0 \\ 0 & 0 \end{bmatrix}. \quad (4.39)$$

**Lemma 3:** For the rotation sequence of Z-Y-X, the angular velocity can be expressed as

[71]

$$\omega = \Psi \dot{\phi}_2, \quad (4.40)$$

where  $\Psi$  is the Jacobian that maps the rate of change of the Euler angles to the angular velocity,

$$\Psi = u(\phi_2) = \begin{bmatrix} 0 & -\sin(\alpha) & \cos(\alpha)\cos(\beta) \\ 0 & \cos(\alpha) & \sin(\alpha)\cos(\beta) \\ 1 & 0 & -\sin(\beta) \end{bmatrix}. \quad (4.41)$$

**Proof of Lemma 3:** If the rotation sequence is Z-Y-X, then the angular velocity can be expressed as a combination of an individual rotation axis multiplied by the time rate of change of the angles about the axis,

$$\omega = [0 \ 0 \ \dot{\alpha}]^T + Rot_z(\alpha)[0 \ \dot{\beta} \ 0]^T + Rot_z(\alpha)Rot_y(\beta)[\dot{\gamma} \ 0 \ 0]^T. \quad (4.42)$$

Thus, from Eq. (4.4)-(4.6), the angular velocity can be reformulated in matrix form as

$$\omega = \Psi \dot{\phi}_2 = \begin{bmatrix} 0 & -\sin(\alpha) & \cos(\alpha)\cos(\beta) \\ 0 & \cos(\alpha) & \sin(\alpha)\cos(\beta) \\ 1 & 0 & -\sin(\beta) \end{bmatrix} \begin{pmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{pmatrix}. \quad (4.43)$$

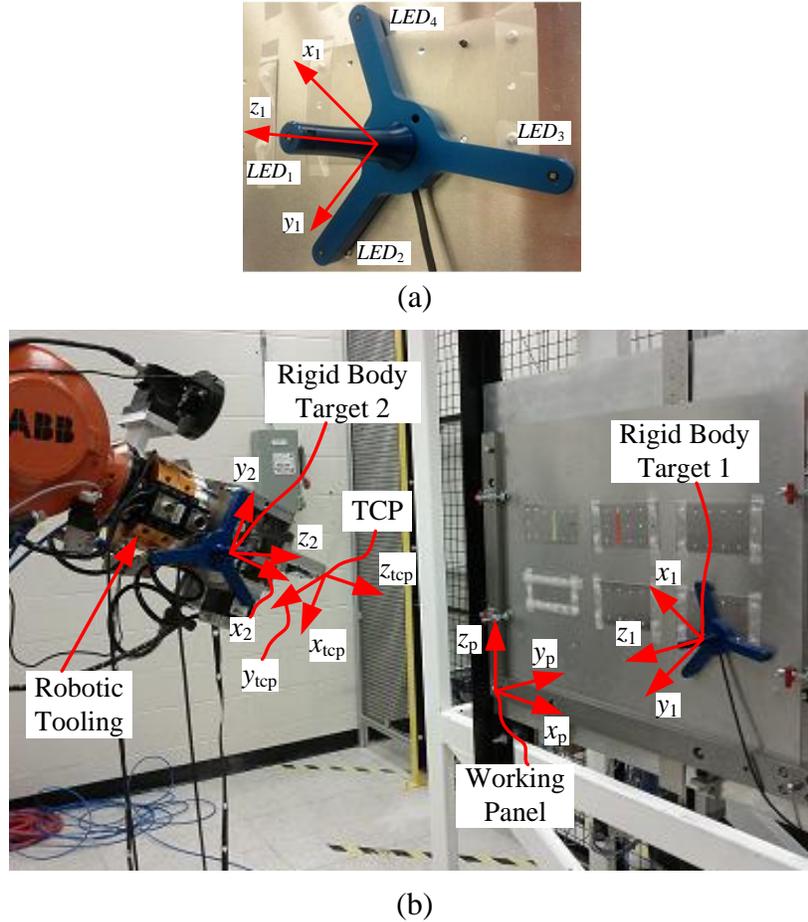
## 4.4 Simulations and Experiments

Based on the aforementioned formulation of the relative pose and motion estimation using IEKF for robot positioning control, a case study of an aircraft robotic riveting system is presented to validate the effectiveness and performance of the methodology.

As mentioned, the purpose is to calculate the state estimations of the relative pose and motion between the TCP of the robot tooling and the working panel. Thus, two rigid body targets with four infrared LEDs are rigidly attached on the robot tooling and the working panel serving as references for relative pose and motion estimation. As shown in Figure 4.2, targets 2 and 1 are attached on the robot tooling and the working panel, respectively.

The local coordinates of the four LED points of the targets with respect to the body-fixed

frames are shown in Table 4.1. Based on the local and measured coordinates of the points of the targets, the relative pose and motion between the TCP of the robot tooling and the working panel are to be estimated dynamically using the aforementioned method.



**Figure 4.2:** (a) Four LEDs on the rigid body target 1. (b) Two rigid body targets attached on the robot tooling and the working panel.

**Table 4.1:** Local coordinates of LED points of the targets in the body-fixed frames

	$p_1$ - Target 1 (mm)			$p_2$ - Target 2 (mm)		
	X	Y	Z	X	Y	Z
LED 1	-0.2075	0.8182	52.4971	-0.0051	1.0743	52.5481
LED 2	0.2794	79.8727	-18.8438	0.2362	79.9293	-19.0332
LED 3	-69.6914	-40.1362	-17.0435	-69.5404	-40.2479	-16.7839
LED 4	69.6195	-40.5548	-16.6097	69.3093	-40.7557	-16.7310

#### 4.4.1 Preliminary Transformation of Target 2-TCP and Target 1-Working Panel

To estimate relative pose of TCP for the rivet insertion control, the local coordinates of target 2 in Table 4.1 need to be transformed to the frame of the TCP. To do so, the pose transformation from the optoelectronic CMM system to the robot base is determined as

$$T_r^m = \begin{bmatrix} 0.0047 & 0.8260 & -0.5636 & 290.84 \\ 0.0178 & -0.5636 & -0.8258 & -1537.75 \\ -0.9998 & -0.0061 & -0.0173 & 1337.83 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

and the pose transformation from the frame of the target 2 to the end-effector of the robot is

$$T_e^{t2} = \begin{bmatrix} 0.8240 & 0.5662 & 0.0195 & 540.41 \\ 0.0249 & -0.0018 & -0.9997 & -165.06 \\ -0.5660 & 0.8242 & -0.0156 & 103.21 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Since the offset of the robot tooling from the end-effector to the TCP is 369.04mm along the z axis of the end-effector, the pose transformation from the frame of target 2 to that of the TCP is

$$T_{tcp}^{t2} = T_{tcp}^e T_e^{t2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -369.04 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.8240 & 0.5662 & 0.0195 & 540.41 \\ 0.0249 & -0.0018 & -0.9997 & -165.06 \\ -0.5660 & 0.8242 & -0.0156 & 103.21 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} 0.8240 & 0.5662 & 0.0195 & 540.41 \\ 0.0249 & -0.0018 & -0.9997 & -165.06 \\ -0.5660 & 0.8242 & -0.0156 & -265.83 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

based on which the local coordinates of points of target 2 with respect to the frame of TCP are calculated by  $b' = T_{tcp}^{t2} p_2$ . The results are listed in Table 4.2.

Similarly, since the target 1 is attached on working panel, the local coordinates of target 1 need to be transformed to the frame of the working panel. First, the pose transformation

from the optoelectronic CMM system to the working panel is obtained as

$$T_p^m = \begin{bmatrix} -0.0160 & 0.5699 & 0.8215 & 1952.98 \\ -0.0111 & 0.8215 & -0.5701 & -1365.19 \\ -0.9998 & -0.0183 & -0.0068 & 351.87 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Then, the pose transformation from the frame of target 1 to that of the working panel is determined as

$$T_p^{t1} = \begin{bmatrix} 0.9392 & -0.3433 & 0.0011 & 537.64 \\ -0.0019 & -0.0086 & -1.000 & 1.94 \\ 0.3433 & 0.9392 & -0.0087 & 184.04 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

based on which the local coordinates of points of target 1 with respect to the frame of the working panel are determined by  $a' = T_p^{t1}p_1$ , as shown in Table 4.2.

In the following, simulations and experiments are presented to verify the convergence and accuracy of the proposed relative pose estimation by IEKF.

**Table 4.2:** Local coordinates of LED points of the targets w.r.t. TCP and working panel

	$a'$ - Target 1 (mm) w.r.t. the panel			$b'$ - Target 2 (mm) w.r.t. TCP		
	X	Y	Z	X	Y	Z
LED 1	537.22	-50.56	184.28	55.67	-217.60	-265.77
LED 2	510.46	20.10	259.32	99.12	-146.18	-199.79
LED 3	485.94	19.46	122.57	-26.38	-149.94	-259.39
LED 4	616.93	18.76	169.99	87.75	-146.54	-338.40

#### 4.4.2 Simulations

A simulation of a robotic position-based visual servoing is carried out for the robotic riveting system using the proposed filter of relative pose estimation, as illustrated in Figure 4.3.

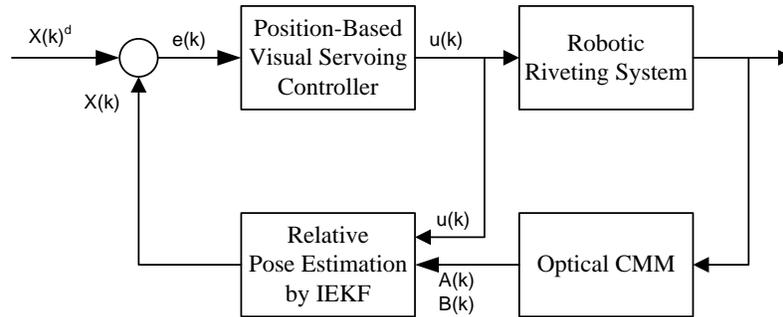
The controller was designed using a simple proportional control law based on the error of

estimated relative pose compared with the desired one, denoted as  $e(k)$ . Note that the estimated Euler angles are changed back into the rotation matrix as outputs of the state estimator, since the calculation of relative pose error in the controller is based on the rotation matrix. In this case, the control input is the velocity of the robot TCP with respect to the frame of end-effector or TCP, denoted by  $u(k)$ . Thus, to map the control input to the state of the relative pose estimation, the control gain in the transition model in Eq. (4.11) can be designed as

$$G(k) = \begin{bmatrix} \Delta t R_p^{tcp} & 0 \\ 0 & \Delta t R_p^{tcp} \\ R_p^{tcp} & 0 \\ 0 & R_p^{tcp} \end{bmatrix} \quad (4.44)$$

$$R_p^{tcp} = R_p^m R_m^r R_r^e(k) R_e^{tcp} \quad (4.45)$$

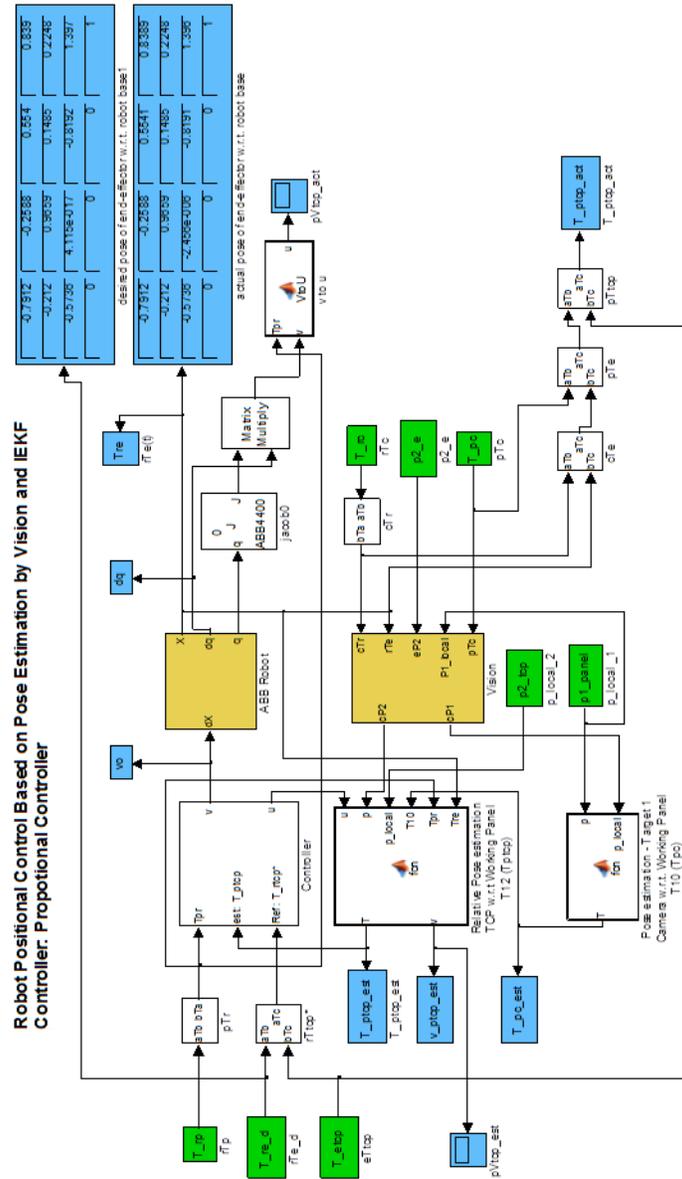
where  $R_p^{tcp}$  is the rotation matrix of the TCP frame with respect to the frame of the working panel.  $R_p^m$  is the rotation matrix from the CMM to the panel.  $R_m^r$  is the rotation matrix from the base of the robot to the CMM.  $R_r^e(k)$  is the rotation matrix of end-effector with respect to the base of the robot at the time step of  $k$ .



**Figure 4.3:** Relative pose estimation integrated in the position-based visual servoing of the robotic riveting system.

The Simulink model of the position-based visual servoing was designed in MatLab 7.12

based on the Robotics Toolbox 9.4 [72], as shown in Figure 4.4. It consists of a controller, a plant of an ABB robot, a vision of optoelectronic CMM, and the relative pose estimation block.



**Figure 4.4:** Simulink model of the robot position-based visual servoing using the proposed filter of the relative pose estimation designed in MatLab.

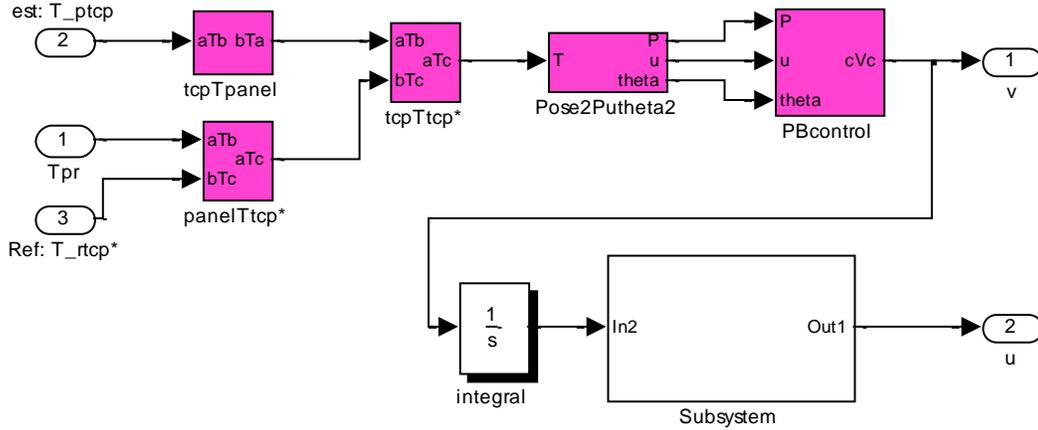
As shown in Figure 4.5, the control error is calculated by comparing the estimated values of relative pose transformation of the TCP with respect to the working panel, denoted as

$T_p^{tcp}$ , and the desired values, denoted as  $T_p^{tcp*}$ , i.e.,

$$T_{tcp}^{tcp*} = \begin{bmatrix} R_{tcp}^{tcp*} & t_{tcp}^{tcp*} \\ 0 & 1 \end{bmatrix} = (T_p^{tcp})^{-1} T_p^{tcp*} \quad (4.46)$$

where  $R_{tcp}^{tcp*}$  and  $t_{tcp}^{tcp*}$  are the orientation and translational errors of the TCP, respectively.

Note that the error is calculated with respect to the frame of the TCP.



**Figure 4.5:** Position-based visual servoing in the Simulink model.

Then, the error of orientation is converted to the representation of the angle-axis parameters by

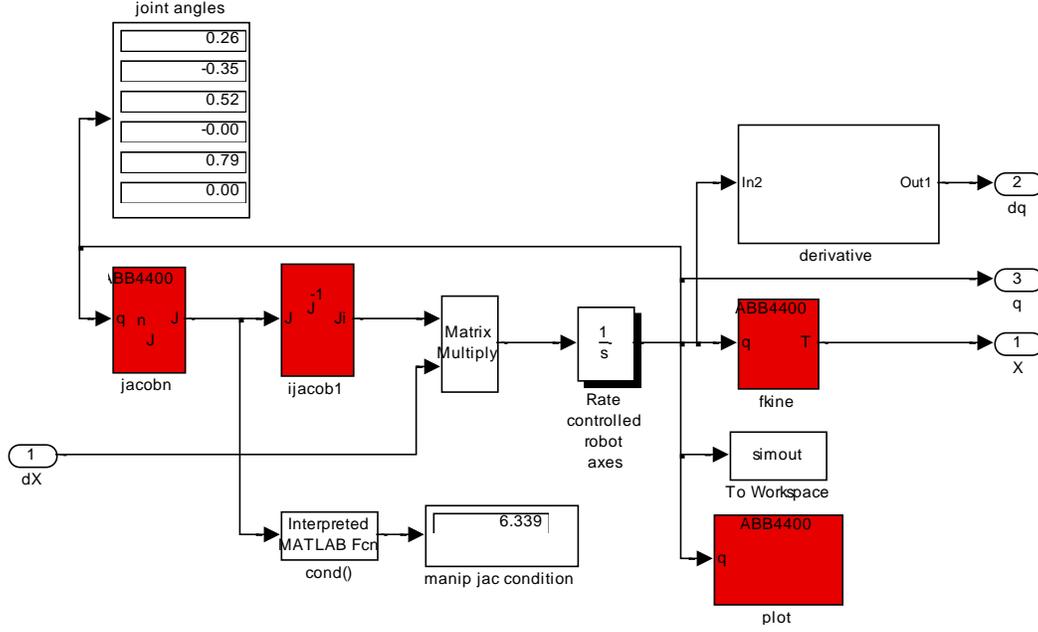
$$Vect(R_{tcp}^{tcp*}) = \vec{a}(k) \sin \theta(k) \quad (4.47)$$

$$Tr(R_{tcp}^{tcp*}) = 1 + 2 \cos \theta(k) \quad (4.48)$$

where  $\vec{a} \in \mathbb{R}^{m \times 1}$  is a unit direction vector representing the equivalent axis of the rotation error of  $R_{tcp}^{tcp*}$ .  $\theta$  is the rotation angle about the axis with the right-hand rule. In doing so, the control law of the proportional gain can be applied using Eq. (4.49) to calculate the velocity of the TCP as an input of the robot plant, as shown in Figure 4.6.

$$v_{tcp}(k) = -K_p e = -K_p (p \quad \vec{a} \theta)^T \quad (4.49)$$

where  $K_p$  is the proportional gain with a tuned value of 0.5.



**Figure 4.6:** Plant of the robot in Simulink model.

In the robot plant in Figure 4.6, the control input of TCP velocity is converted to the robot joint space using the robot Jacobian mapping as

$$\dot{\theta}(k) = (J_n(\theta(k)))^{-1} v_{tcp}(k) \quad (4.50)$$

where  $J_n(\theta(k)): \theta(k) \rightarrow \mathbb{R}^{2m \times \lambda}$  is the Jacobian matrix function defined on a set of joint angles  $\theta(k)$  in  $\mathbb{R}^{\lambda \times 1}$ , and  $\lambda$  is the number of the joints of the robot. Basically,  $J_n(\theta(k))$  maps the robot joint angle rates  $\dot{\theta}(k) \in \mathbb{R}^{\lambda \times 1}$  to the velocity of the TCP, derived by

$$J_n(\theta(k)) = \begin{bmatrix} Z_1(k) \times P_{1(\lambda+1)}(k) & \cdots & Z_i(k) \times P_{i(\lambda+1)}(k) & \cdots & Z_\lambda(k) \times P_{\lambda(\lambda+1)}(k) \\ Z_1(k) & \cdots & Z_i(k) & \cdots & Z_\lambda(k) \end{bmatrix}, \quad (4.51)$$

where  $Z_i(k)$  is the  $z$ -axial vector of the  $i$ th revolute joint of the robot with respect to the frame of TCP, assuming the joint rotation is about  $z$  axis of the joint.  $P_{i(\lambda+1)}(k)$  is the

position vector from the  $i$ th robot joint to the TCP.

Then, the pose of robot TCP can be calculated based on the robot forward kinematics from integration of the joint angle rates,

$$\begin{aligned} T_r^{tcp}(k) = r(\theta(k), g) &= \begin{bmatrix} R_{0(\lambda+1)}(k) & P_{0(\lambda+1)}(k) \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \prod_{i=1}^{\lambda+1} R_{(i-1)i}(k) & \sum_{i=0}^{\lambda} R_{0i}(k)\eta'_i \\ 0 & 1 \end{bmatrix}, \end{aligned} \quad (4.52)$$

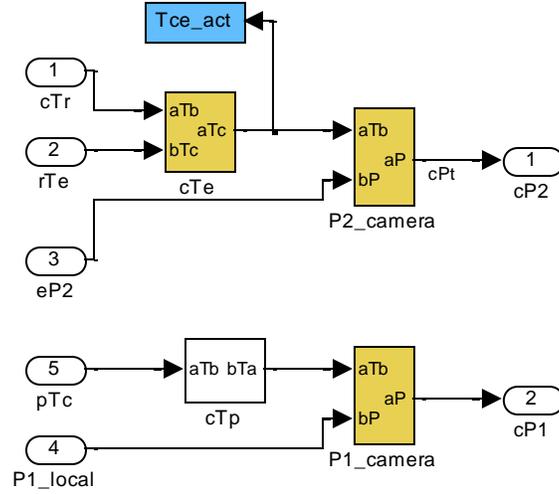
where  $g \in \mathbb{R}^{2m \times \lambda}$  is the robot's geometric or kinematic parameters, including the local body vectors of the links of the robot, denoted as  $\eta'$ . Specifically,  $\eta'_\lambda$  is the body vector from the end-effector to the TCP.  $R_{0(\lambda+1)}(k)$  and  $P_{0(\lambda+1)}(k)$  are the rotation matrix and the position vector of the robot TCP with respect to the robot base.  $R_{(i-1)i}(k)$  is the rotation matrix of the  $i$ th joint with respect to the  $(i-1)$ th joint.  $R_{0i}(k)$  is the rotation matrix of the  $i$ th joint with respect to the robot base.

The Simulink block of the optoelectronic CMM for simulating the point target measurements is illustrated in Figure 4.7. The simulated point measurements are generated by

$$b = T_c^{tcp} b' = T_c^r T_r^{tcp}(k) b' \quad (4.53)$$

$$a = T_c^p a'. \quad (4.54)$$

For simulation of robot positioning control, the initial and desired robot joint angles are set as  $q^0 = (0 \ 0 \ 0 \ 0 \ 10^\circ \ 0)^T$  and  $q^d = (15^\circ \ -20^\circ \ 30^\circ \ 0^\circ \ 15^\circ \ 0^\circ)^T$ , respectively. The schematic diagrams of initial and desired robot positions are illustrated in Figure 4.8. The D-H parameters of the ABB 4400/45kg robot manipulator are listed in Table 4.3.



**Figure 4.7:** The Simulink model of the optoelectronic CMM to simulate the point measurements.

**Table 4.3:** D-H parameters of ABB 4400/45kg robot manipulator

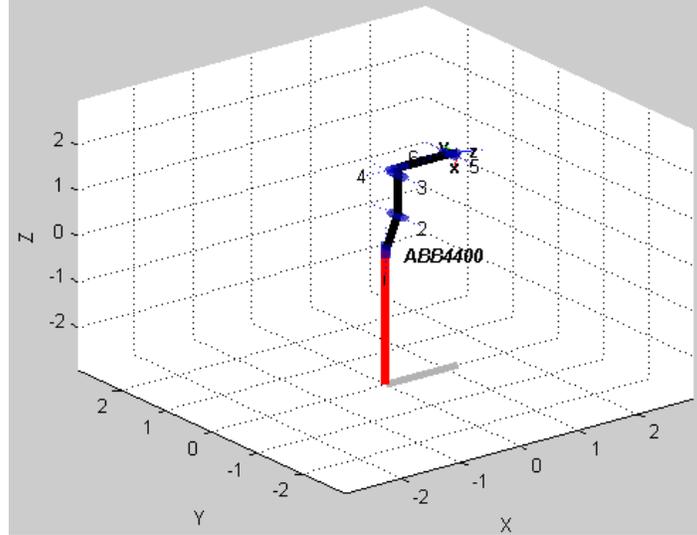
	theta (rad)	d (mm)	a (mm)	alpha (rad)
Link 1	0	680	200	$-\pi/2$
Link 2	$-\pi/2$	0	890	0
Link 3	0	0	150	$-\pi/2$
Link 4	0	880	0	$\pi/2$
Link 5	0	0	0	$-\pi/2$
Link 6	$\pi$	140	0	0

Then, based on the robot forward kinematics, the desired pose of TCP with respect to the robot base is obtained as

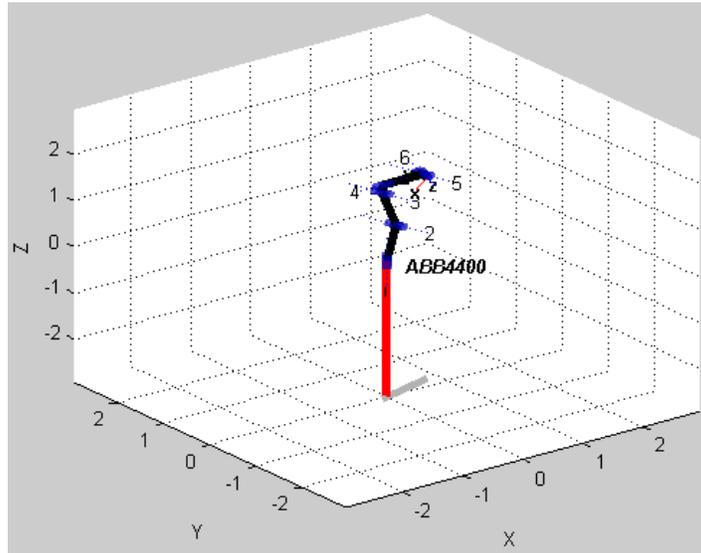
$$T_r^{tcp} = T_r^e T_e^{tcp} = \begin{bmatrix} -0.4082 & -0.2588 & 0.8754 & 1207.04 \\ -0.1094 & 0.9659 & 0.2346 & 323.42 \\ -0.9063 & 0 & -0.4226 & 1296.1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Thus, the desired relative pose of TCP with respect to the working panel can be given as

$$T_p^{tcp} = T_p^r T_r^{tcp} = \begin{bmatrix} 0.1078 & -0.9679 & -0.2271 & 989.18 \\ -0.4234 & -0.2514 & 0.8703 & -435.63 \\ -0.8995 & 0.0023 & -0.4370 & 292.08 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$



(a)

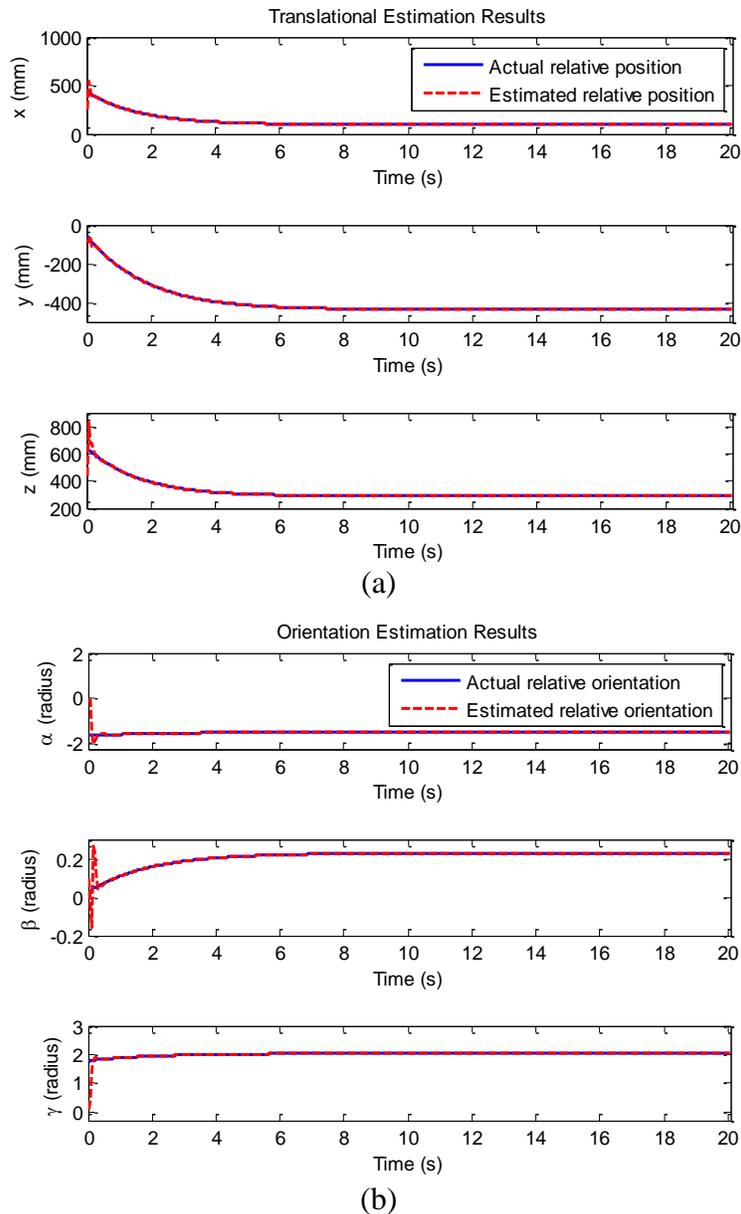


(b)

**Figure 4.8:** Schematic diagrams of the robot poses in the simulation. (a) The initial position. (b) The desired position.

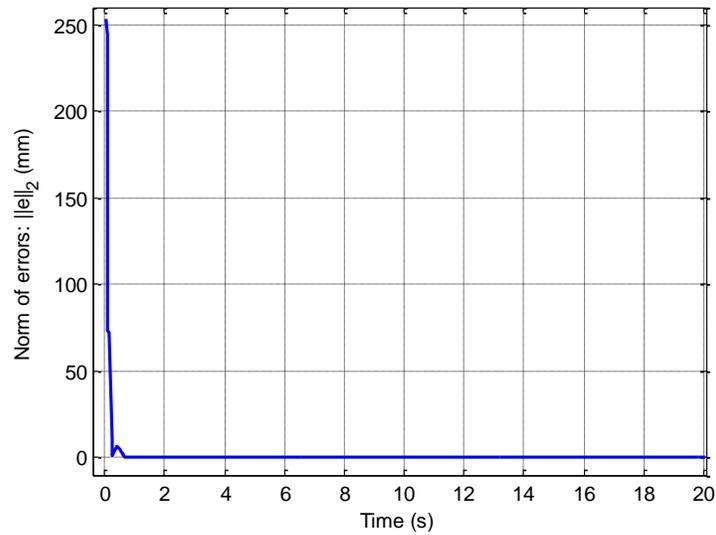
For the filter setup of relative pose estimation, the tuned noise covariance matrices are set based on the noise variances of  $\sigma_v^2 = 4 \times 10^{-4} (m/s^2)^2$  and  $\sigma_w^2 = 1 \times 10^{-9} m^2$ , instead of null values. The process and measurement noise variances are obtained from the acceleration range of the robot motion and offline sampling of the optical CMM device, respectively. For simplicity, the initial state value and covariance are both set to be zero.

To evaluate the estimation accuracy, the estimated relative poses of the TCP with respect to the working panel are compared with the actual ones calculated from the robot forward kinematics. The trajectories of the actual and estimated relative translational and orientation are illustrated in Figure 4.9(a) and (b), respectively.

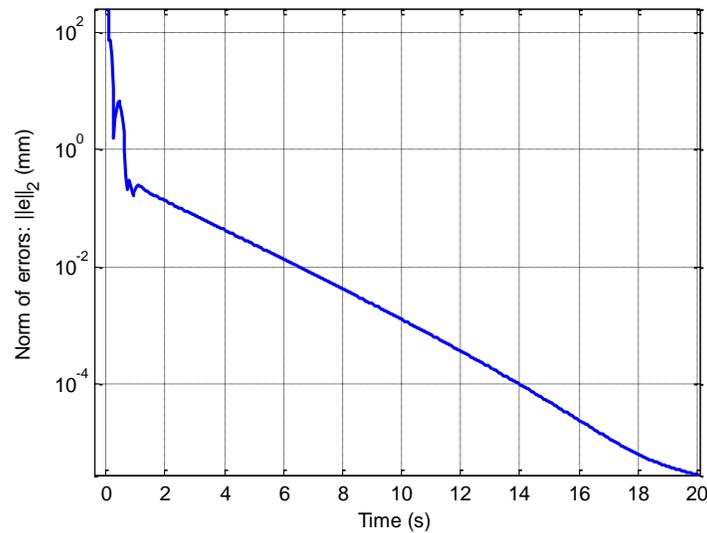


**Figure 4.9:** Simulation results of dynamic pose estimation. (a) Relative translational estimation results. (b) Relative orientation estimation results. The blue solid lines represent the actual relative poses, and the red dashed lines represent the estimated poses.

As shown in Figure 4.10, the norm of error of relative pose estimation converges sharply from over 250 mm to 1 mm at about 0.8 s, and then continues to decline steadily to the required accuracy of 0.1 mm at about 2.5 s. Note that the total simulation time is 20 s and the fix-step size of sampling period is 0.04 s. In the next section, the experiment is provided to further verify the filter with the same system setup and filter initialization.



(a)

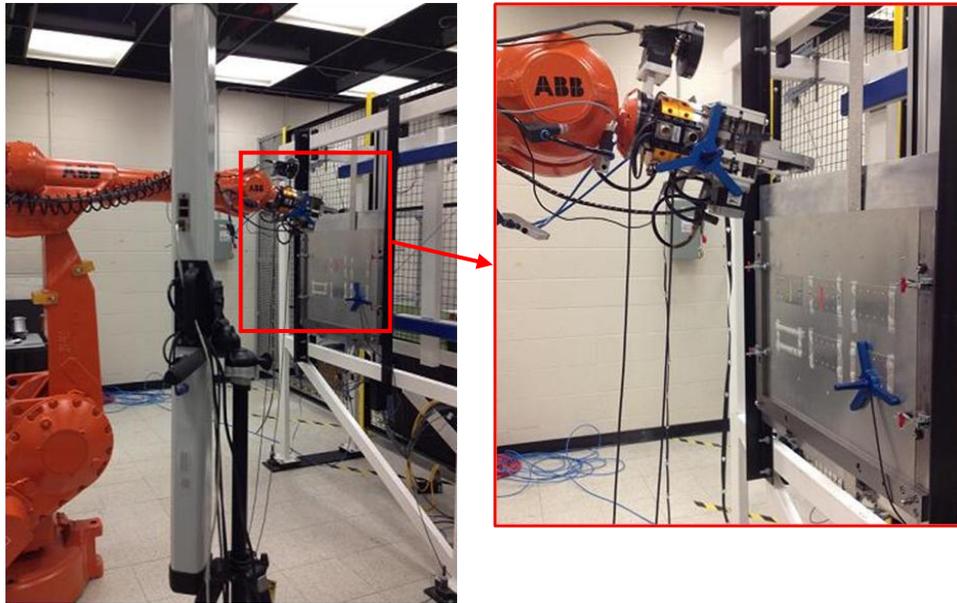


(b)

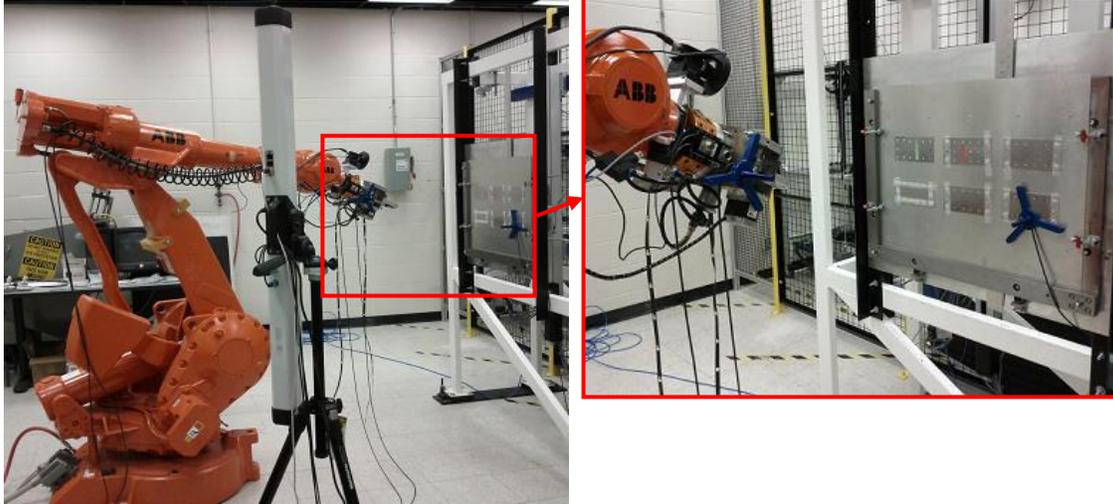
**Figure 4.10:** Norm of the error of relative pose estimation during simulation. (a) Linear scale of norm of error values. (b) Log scale of norm of error values.

### 4.4.3 Experiments

To further verify the performance of the proposed method, experiments are carried out on the robotic riveting system under the equivalent system setup and IEKF initialization. Note that experiments focus on the relative pose estimation alone without the experimental implementation in the robot position-based visual servoing. First, the points of targets are measured and stored while the robot moves along the predefined linear trajectory with average velocities of  $25 \text{ mm/s}$  and  $0.0154 \text{ rad/s}$ . The initial and final positions of the robot with the working panel are depicted in Figures 4.11 and 4.12, respectively. Then, the method is applied offline to process the collected point measurements. Lastly, the estimated and nominal relative poses are compared and the norm of the error is calculated accordingly.



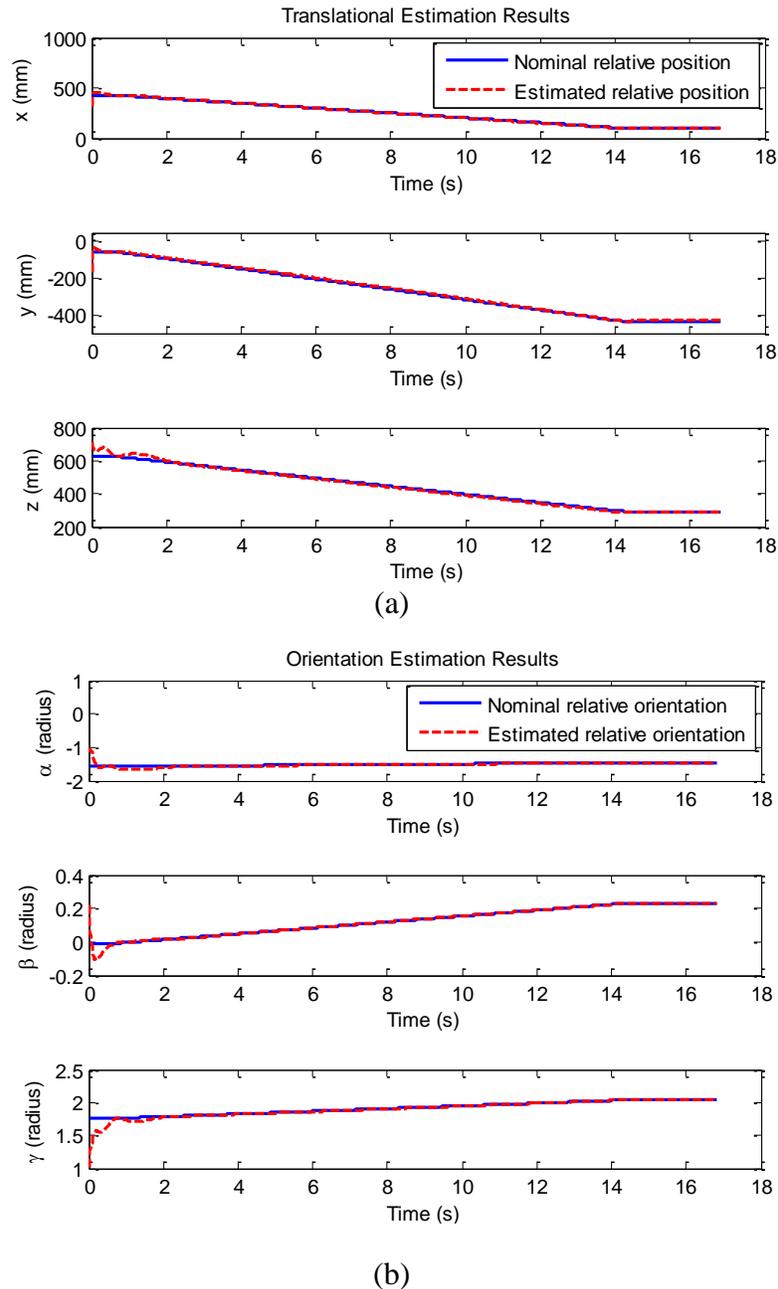
**Figure 4.11:** Initial position of the robot with joint angles of  $q^0 = [0 \ 0 \ 0 \ 0 \ 10 \ 0]$ .



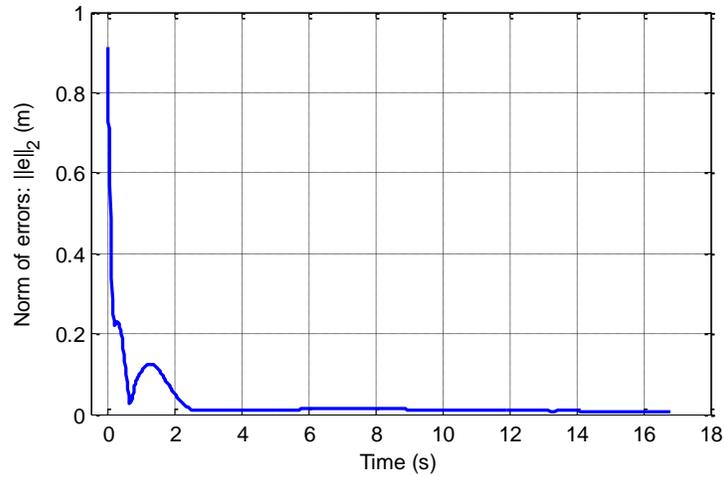
**Figure 4.12:** Final position of the robot with joint angles of  $q^d = [15^\circ -20^\circ 30^\circ 0^\circ 15^\circ 0^\circ]$ .

As illustrated in Figure 4.13, it is observed that the performance of the relative pose estimation ensured the convergence to the nominal trajectories. In Figure 4.14, the norms of errors of the relative pose estimation converge dramatically from approximately 800 *mm* to 25 *mm* at about 0.8 *s*, to 9 *mm* at about 2.5 *s*, and then decline continuously before settling down stably to around 6.9 *mm* after about 14.3 *s* at the end when the robot stops. The bump around 1 *s* is normal as overshooting happens at the beginning of filtering, mainly due to zero initial value of the state. The final converging errors are 1.7 *mm*, 3.6 *mm*, -3.6 *mm*,  $-0.6e-3$  *rad*,  $-3.8e-3$  *rad*, and  $-1.8e-3$  *rad*. The errors are mainly contributed by the poor robot absolute positioning accuracy caused by the systematic errors, such as deflection due to flexibility and loading, assembly errors due to backlash, and manufacturing errors due to machining tolerances, etc. [73]. That explains the necessity of the proposed dynamic pose estimation. The approach is capable of directly estimating the positioning errors of the TCP with respect to the working panel during motion. Based on the approach, further research of calibration-based iterative learning control (ILC) is being developed to reduce the estimated errors [97, 98]. Also, it could be extended for a

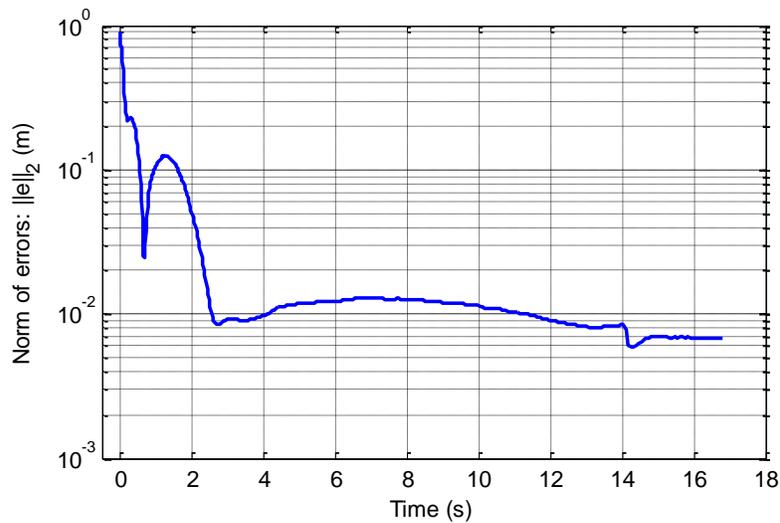
wide range of robotic positioning applications for intelligent manufacturing. Further, no control input was enrolled in the experiment as opposed to the simulation with the control input of the TCP velocity, which obviously benefited better motion prediction.



**Figure 4.13:** Experimental linear trajectory comparison. (a) Relative translational estimation results. (b) Relative orientation estimation results. The blue solid lines are the nominal trajectory and the red dashed lines are the estimated trajectory.



(a)



(b)

**Figure 4.14:** Converging results of norms of the errors of relative pose estimation in the experiment. (a) Linear scale of norm of errors. (b) Log scale of norm of errors.

## 4.5 Summary

In this section, the state estimation of relative pose from point measurements is proposed between two rigid bodies during motion using iterative extended Kalman filter (IEKF). Unlike the conventional way of tracking one rigid body at a time with respect to cameras, the method tracks two rigid bodies simultaneously from point measurements provided by

an optical CMM. To do so, when the relative pose and motion are defined as a state vector, state-space discrete time-varying non-linear stochastic systems are modeled consisting of state transition and observation models. Specifically, the observation model nonlinearly maps the point measurements to the state of relative pose and motion. To tackle the state estimation, IEKF is adopted to minimize the linearization errors. The performance of the proposed method has ensured the convergence of the estimation through the verifications of the simulations and experiments on a robotic riveting system.

Note that the proposed filter for relative pose estimation merely provides estimated feedback of relative pose errors for the control of visual servoing. Thus, the control of visual servoing is not the focus in this research. The simulation with PD control merely demonstrated the performance of the proposed filter. The experiment showed the filter could dynamically estimate the relative pose errors, which potentially could be used as feedback for control loop. With the proposed method of relative pose estimation, further research has been aiming at the control method of calibration based iterative learning control (ILC) for robotic riveting [97, 98]. The method of calibration based ILC is to reduce the systematic errors that were estimated in the experiment, while the method of relative pose estimation in this research just guarantees the estimation accuracy.

## CHAPTER 5 ROBUST POSE ESTIMATION

In this chapter, a novel outlier diagnosis method is proposed for robust pose estimation of rigid body motions from outlier contaminated 3D point measurements. Due to incorrect correspondences in a cluttered measuring environment, observed point data are contaminated by outliers, which are unusual gross errors that lie out of an overall error distribution. Standard least-squares methods for pose estimation are highly sensitive to outliers. For this reason, an outlier diagnosis method is developed to preprocess measured point data prior to pose estimation. This diagnosis method detects and removes outliers based on a relaxation method with rigid body constraints of a rigid body. Simulations and experiments prove the effectiveness and advantages of high breakdown point and ease of implementation.

### 5.1 Problem Statement

As illustrated in Figure 5.1, the cubic structure represents a rigid body and its corners represent attached markers from which 3D coordinate data are measured. Note that  $p'_i$  and  $p_i$  are the 3D coordinates of the  $i$ th marker measured at two different poses. Based on  $p'_i$  and  $p_i$ , the target is to optimally estimate the pose transformation from frame  $\{B'\}$  to  $\{B\}$ , denoted by  $T_B^{B'}$ . The solution of  $T_B^{B'}$  was given in Chapter 1.

However, considering outliers that occur in the measurements  $p'_i$  and  $p_i$ , static pose estimation cannot give a robust solution due to the nature of least square fitting criterion. Therefore, an outlier diagnosis of  $p'_i$  and  $p_i$  is proposed and conducted in order to detect and remove the outliers before pose estimation.

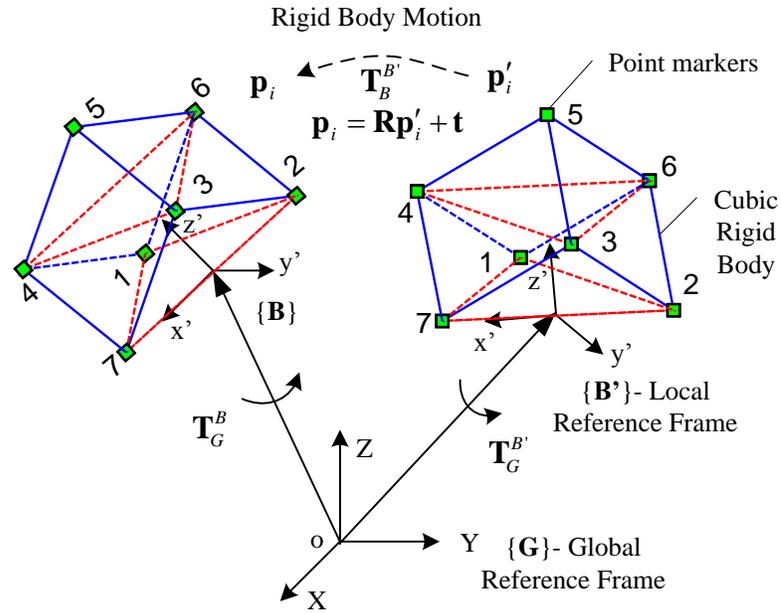


Figure 5.1: Motion of the cubic rigid body.

## 5.2 Outlier Diagnosis

During rigid body motion, point measurements must obey rigid body constraints. Based on these rigid body constraints, an outlier diagnosis is developed by utilizing a relaxation method. The principle is to replace rigid body constraints with values into an objective function of outlier diagnosis so as to exact a penalty on the objective if the constraints are not satisfied.

### 5.2.1 Rigid Body Constraints

The rigid body constraint means that distances between points must not change when the rigid body is moved. Thus, the Euclidean distances calculated between point measurements should remain statistically constant or equivalent to the nominal distances when the model of the rigid body is known. Assuming the model is given, the absolute residuals between the calculated and nominal point distances can be determined by

$$\varepsilon_{ij} = |l_{mij} - l_{oij}| \leq \delta, \text{ for } 1 \leq i, j \leq n, i \neq j \quad (5.1)$$

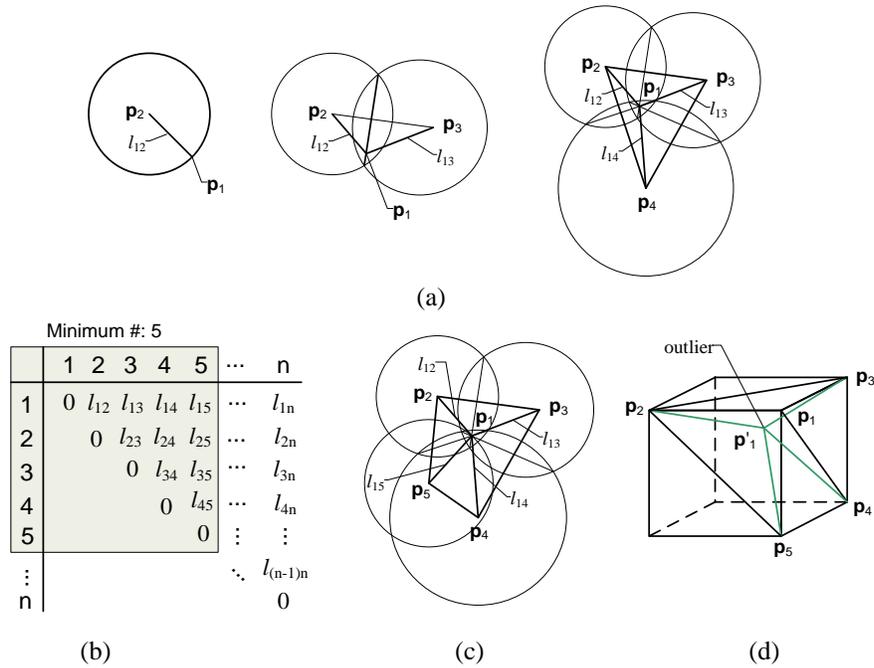
where

$$l_{mij} = \|\mathbf{p}_i - \mathbf{p}_j\| = \left[ (\mathbf{p}_i - \mathbf{p}_j)^\top (\mathbf{p}_i - \mathbf{p}_j) \right]^{1/2} \quad (5.2)$$

where  $p_i$  and  $p_j$  are measurements of the  $i$ th and  $j$ th marker,  $l_{mij}$  and  $l_{oij}$  are the measured and nominal distances between  $p_i$  and  $p_j$ , respectively. The deviation  $\varepsilon_{ij}$  must be within the requirement of  $\delta$  to satisfy the rigid body constraints. The value of  $\delta$  is determined by the error model of the distance calculation due to the sensor accuracy of the measuring system. If the model of the rigid body is unknown, the model-free approach can be developed by building the reference distances from a “teaching-by-showing” process. The process is to show the rigid body to the measuring system such that the point distances can be learned and stored as references. Thus, the deviation of the measured distances from the references can still be monitored by using (5.1).

Based on the distance deviations, we can find out the candidate outliers. Large deviations out of  $\delta$  indicate the violation of rigid body constraints and the existing of outliers. Thresholding of these errors by  $\delta$  is based on the deviation of point measurement distribution. However, the procedure needs to be robust in order to provide a reliable measure for the recognition of outliers. Geometrically, at least four distances from one point to four other noncollinear and non-coplanar points are required to yield a unique solution of the point’s 3D position, as shown in Figure 5.2. Figure 5.2(a) demonstrates that if there are only three or less measured distances identical to the nominal values, it may implicitly embrace outliers. In particular, there may be an outlier of reflection occurred when only three of distance constraints are satisfied. Figure 5.2(b) shows that at least five points are needed since four distance constraints of  $l_{12}$ ,  $l_{13}$ ,  $l_{14}$ ,  $l_{15}$  are capable of

locating the unique 3D position of point  $p_1$  in Figure 5.2(c). An example of a detected outlier is provided in Figure 5.2(d). Thus, though four points targets were used in static and dynamic pose estimation in Chapters 3 and 4, targets with minimum five points are recommended for the concern of robustness against outliers using rigid body constraints.



**Figure 5.2:** Point distances under the rigid body constraints. (a) one, or two, or three distance constraints are not sufficient for unique 3D positioning; (b) the minimum number of points is 5 for unique 3D positioning; (c) four distance constraints for point  $p_1$ ; (d) an outlier example.

### 5.2.2 Outlier Diagnosis by Relaxation

Relaxation techniques iteratively assign values to mutually constrained objects in such a way as to ensure that the values remain consistent [74, 75]. In our case, objects are markers of a rigid body mutually constrained by the rigid body constraints defined in (5.1). The mechanism of outlier diagnosis using relaxation techniques begins by replacing the rigid body constraints with compatibility indicators. For  $1 \leq i, j \leq n$ , a

compatibility indicator, denoted by  $c_{i,j}$ , is defined by

$$c_{i,j} = \begin{cases} 1, & \text{when the rigid body constraint between } \mathbf{p}_i \text{ and } \mathbf{p}_j \text{ is satisfied} \\ 0, & \text{no comment nor judgment made} \\ -1, & \text{when the rigid body constraint between } \mathbf{p}_i \text{ and } \mathbf{p}_j \text{ is not satisfied} \end{cases} \quad (5.3)$$

Essentially,  $c_{i,j}$  indicates the compatibility of the rigid body constraints between markers  $p_i$  and  $p_j$ . In particular, the compatibility for each marker  $p_i$  itself is defined by  $c_{i,i} = 1$ .

A merit score ( $s_i$ ) is defined and assigned to each marker  $p_i$  according to how well all other markers  $p_j$  are compatible with the rigid body constraints between  $p_j$  and  $p_i$  when  $p_i$  is not an outlier. Higher merit scores indicate better compatibility of constraints and less likelihood of outliers, and vice versa. Subsequent iterations use the product of the merit scores  $s_j^{k-1}$  of  $p_j$  and the compatibility of the constraints  $c_{i,j}$  in (5.3) to assign a new merit score  $s_i^k$  to the marker  $p_i$ . Marker  $p_j$  with higher merit score and better compatibility contributes more positive support to the assumption that marker  $p_i$  is not an outlier, and vice versa. The quantity of total support from all other markers after several iterations provides a robust measure for outlier diagnosis. Therefore, the outlier diagnosis by relaxation involves iterative, “parallel” adjustment of merit scores for each marker on the basis of local outlier evidence, expressed by [29]

$$\begin{aligned} s_i^k &= s_i^{k-1} + \sum_{j=1}^{n-1} (s_j^{k-1} c_{i,j}) \\ &= s_i^{k-1} c_{i,i} + \sum_{j=1}^{n-1} (s_j^{k-1} c_{i,j}) \\ &= \sum_{j=1}^n (s_j^{k-1} c_{i,j}) \end{aligned} \quad (5.4)$$

The initial values of merit scores  $s_i^0$  can be assigned in a variety of ways. It was chosen to

use the number of satisfied rigid body constraints as an initialization, since a priori knowledge of outlier is unknown. Essentially, the original problem of finding outliers among marker measurements becomes to iteratively calculate the merit scores in (5.4) starting from the initial values  $s_i^0$ .

Here are the procedures developed for the outlier diagnosis by relaxation [29]:

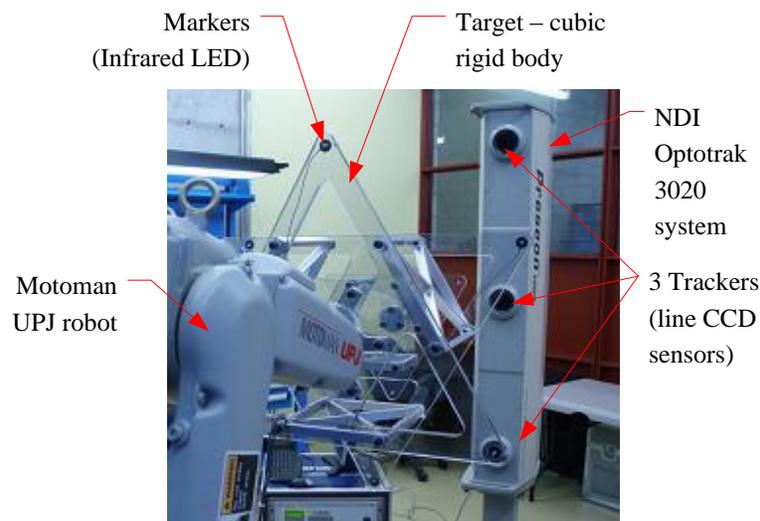
- i. Calculate the compatibility of rigid body constraints  $c_{i,j}$  using (5.1) and (5.3);
- ii. Assign the initial values of merit scores for each point  $s_i^0$ ;
- iii. Iteratively compute the merit scores  $s_i^k$  with  $s_i^{k-1}$  and  $c_{i,j}$  using (5.4);
- iv. Detect and remove the outliers according to the final merit scores.

For quantifying the robustness of an estimator, a breakdown point had been defined to measure the fraction of data that can be contaminated without breaking down the estimator [76]. For example, the LS estimator has a breakdown point of 0% since one single outlier can affect the estimation significantly. Compared with other referred robust estimators, such as M-estimator or LMedS method, the advantage of using the proposed outlier diagnosis by relaxation is higher breakdown point and ease of implementation. This method has a breakdown point of more than 0.5, providing that at least three marker matches remain for pose estimation after removing the outliers. Additionally, this method can be easily implemented as long as the observed markers are rigidly moved. Furthermore, the method can be extended to multiple sets of markers that are attached to a number of different rigid bodies. Outlier diagnosis can be carried out simultaneously for these multiple sets of markers. As a downside of the method, the computational complexity for the compatibility calculation in (5.3) is  $O(n^2)$ . However, it is a one-time job prior to the whole iterative procedure of the outlier diagnosis. For the iterations of the diagnosis, the complexity is linear about the number of markers  $O(n)$ . Nevertheless, the

computational cost is always under control since the number of markers is usually user-defined and limited.

### 5.3 Simulations and Experiments

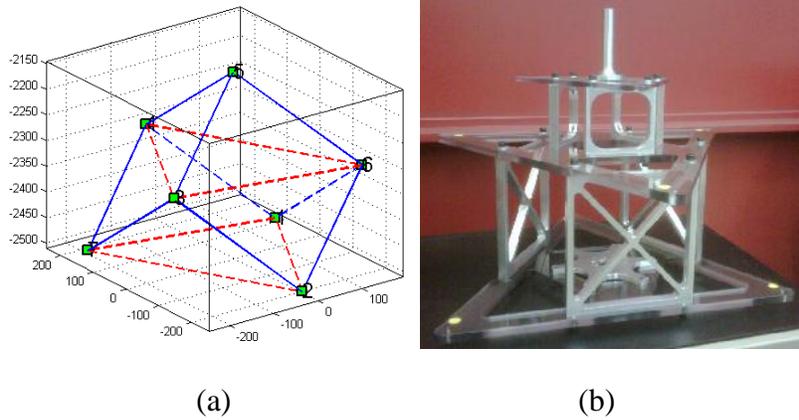
As shown in Figure 5.3, the experimental system located at the Aerospace Manufacturing Technology Centre (AMTC), is consisted of a Motoman UPJ robot, an Optotrak 3020 system from NDI Inc., and a cubic rigid body target. To validate the proposed robust pose estimation, the cubic rigid body target was attached to the end-effector of the robot and moved by the robot within the workspace. 7 markers were attached to the 7 corners of the cubic rigid body, as illustrated in Figure 5.4.



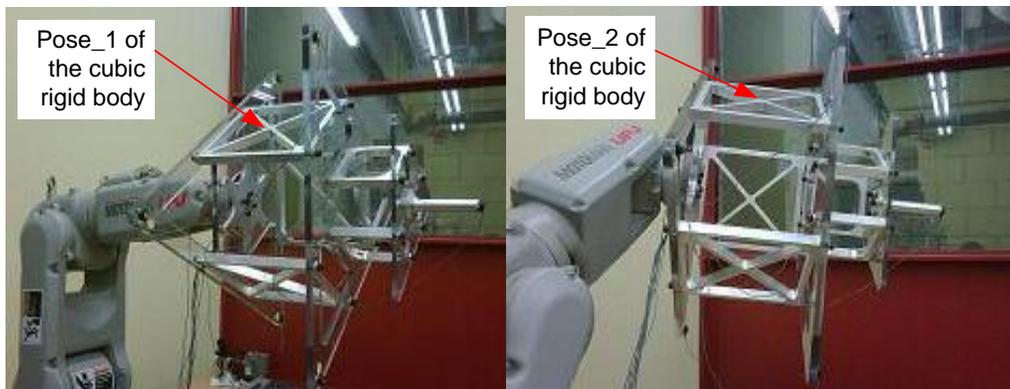
**Figure 5.3:** Experimental system set-up.

The positions of these markers were measured by the Optotrak system, which is also an optical CMM. With synchronized active markers and three line imaging sensors, the optical CMM systems provide efficient triangulation that enables high frame rates of 3D data delivery. The repetitive position measurements of these markers were recorded for 5 seconds at a frame rate of 100Hz at the two different poses of the robot, as shown in

Figure 5.5. Note that marker\_1 represents the number one marker, and pose\_1 represents the first pose of the cubic rigid body.



**Figure 5.4:** (a) 7 Markers in the 3D space; (b) target of the cubic rigid body.



**Figure 5.5:** Two poses of the cubic rigid body.

### 5.3.1 Case Study of Four Outliers

As an ultimate case study, four outliers were added at marker 1, 2, 6, and 7 of pose\_1. Thus, the percentage of outlier is four out of seven, i.e., about 57.1%. These outliers were generated randomly from a mixture error model with Gaussian and uniform noise components, defined as

$$\xi = (1-\eta)N(0, \sigma^2) + \eta U(a, b) \quad (5.5)$$

where  $N(0, \sigma^2)$  is a Gaussian noise, with zero mean and a standard deviation of  $\sigma$ ,  $U(a, b)$

is a uniform noise, with a bound between  $a$  and  $b$ , and  $\eta$  is the proportion of outlier in the total noise. Since the data acquired from the experiments had Gaussian noise already, we chose a uniform noise model alone for outlier simulations. From the outlier contaminated data, the pose transformation from pose\_1 to pose\_2 was estimated, denoted by  $T_{outlier}$  and listed in Table 5.1. To demonstrate the influence of these four outliers, the cube at pose\_1 was transformed to pose\_2 by the calculated  $T_{outlier}$ . As shown in Figure 5.6(a), the deviation of two cubes indicates the error of pose estimation due to the outliers.

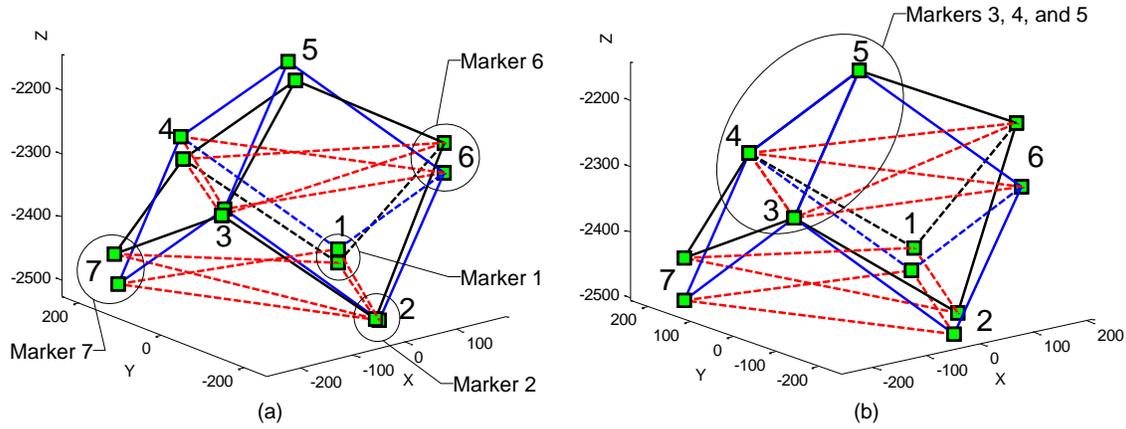
**Table 5.1:** Pose Transformation Before and After the Outlier Contamination

Pose Transformation ( $T$ )							
$T_{actual}$				$T_{outlier}$			
0.9975	-0.0007	-0.0701	-215.554	0.9989	-0.0157	-0.0444	-161.798
0.0006	1	-0.0011	-3.1334	0.0156	0.9999	-0.0007	-1.9748
0.0701	0.001	0.9975	3.6284	0.0444	-0	0.999	6.1705
0	0	0	1	0	0	0	1

As mentioned before, the threshold  $\delta$  in Eq. (5.1) depends on the accuracy of the measuring system. In our case, the standard deviations of the Optotrak system are about  $\sigma = [3e-3, 3e-3, 2e-2]^T$  mm at  $x$ ,  $y$ , and  $z$  axis, respectively. Thus, the three-sigma rule leads to a threshold of Euclidean distance deviation, determined by  $\delta = 2[(3\sigma)^T (3\sigma)]^{1/2} = 0.1227$ mm. Therefore, the compatibility indicators  $c_{ij}$  can be calculated by Eq. (5.1) and (5.3), as listed in Table 5.2. By counting the number of satisfied constraints, the initial merit scores  $s_i^0$  of 7 markers are 1, 1, 3, 3, 3, 1, and 1, respectively. Table 5.3 and Figure 5.7(a) show the merit scores of all markers after relaxation with 7 iterations. As a result, the outliers (markers 1, 2, 6, and 7) can be detected clearly from the negative sign “-1” of the normalized merit scores in Figure 5.7(b).

After removing the outliers from the measurements, the data of markers 3, 4, and 5 are kept for the pose estimation. As listed in Table 5.4,  $T_{removed}$  represents the pose

transformation after removing the outliers. The satisfied results are illustrated in Figure 5.6(b), showing two cubic rigid bodies placed together using  $T_{removed}$ . Essentially, this case study has proved the effectiveness of the method and a breakdown point of more than 0.5, providing that at least three markers remain for the pose estimation after the outlier diagnosis.



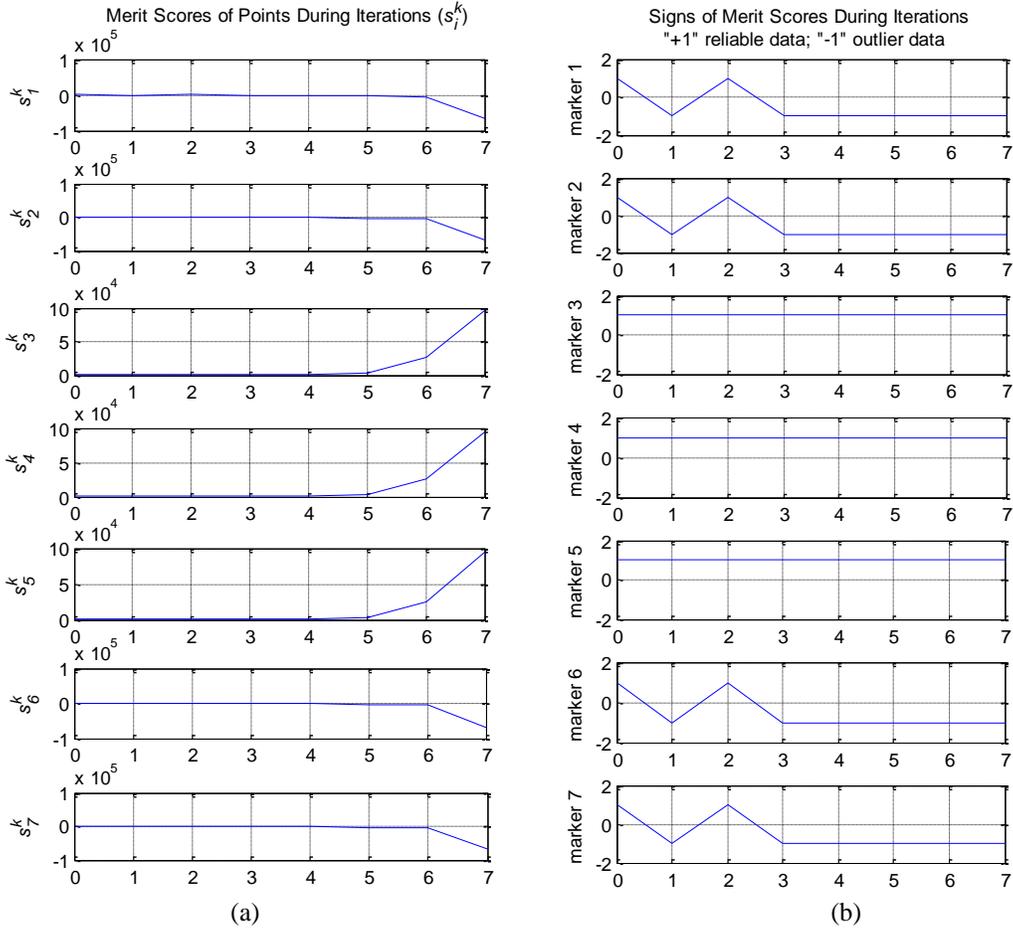
**Figure 5.6:** Outliers occurred at markers 1, 2, 6, and 7; (a) Pose transformation determined with four outliers; (b) Pose estimation from the remaining three markers 3-5 after removing the outliers.

**Table 5.2:** Indicators of Compatibility - Replacement of Distance Deviation with Integers

Indicators of compatibility for rigid body constraints ( $c_{i,j}$ )							
	Point 1	Point 2	Point 3	Point 4	Point 5	Point 6	Point 7
Point 1	1	-1	-1	-1	-1	-1	-1
Point 2	-1	1	-1	-1	-1	-1	-1
Point 3	-1	-1	1	1	1	-1	-1
Point 4	-1	-1	1	1	1	-1	-1
Point 5	-1	-1	1	1	1	-1	-1
Point 6	-1	-1	-1	-1	-1	1	-1
Point 7	-1	-1	-1	-1	-1	-1	1

**Table 5.3:** Iterations of Merit Scores for Outlier Diagnosis by Relaxation (Four Outliers)

Merit scores for point $i$ ( $s_i$ )							
Iterations	Point 1	Point 2	Point 3	Point 4	Point 5	Point 6	Point 7
0 - initial	1	1	3	3	3	1	1
1	-11	-11	5	5	5	-11	-11
2	7	7	59	59	59	7	7
3	-191	-191	149	149	149	-191	-191
4	-65	-65	1211	1211	1211	-65	-65
5	-3503	-3503	3893	3893	3893	-3503	-3503
6	-4673	-4673	25691	25691	25691	-4673	-4673
7	-67727	-67727	95765	95765	95765	-67727	-67727



**Figure 5.7:** (a) Merit scores during 7 iterations; (b) Signs obtained from normalized merit scores, “-1” indicate outliers (markers 1, 2, 6, and 7), and “+1” indicate the reliable data (markers 3, 4, and 5).

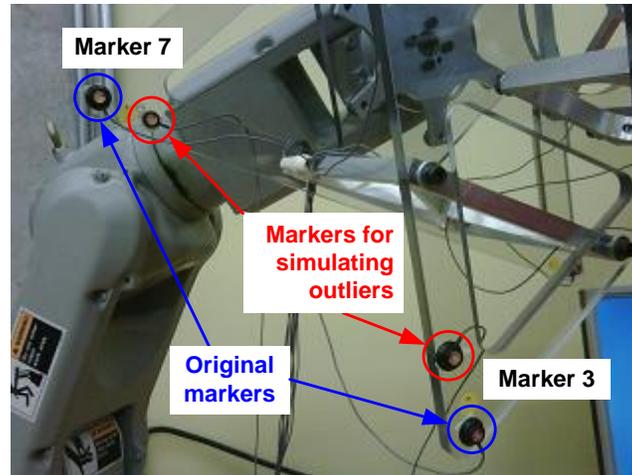
**Table 5.4:** Pose Transformation Before and After the Outlier Diagnosis

$T_{\text{outlier}}$				$T_{\text{removed}}$			
0.9956	0.0335	-0.088	-262.963	0.9975	-0.0007	-0.0701	-215.3592
-0.0284	0.9979	0.058	134.8958	0.0007	1	-0.001	-2.9641
0.0898	-0.0553	0.9944	-25.5177	0.0701	0.0009	0.9975	3.6335
0	0	0	1	0	0	0	1

### 5.3.2 Experiment Verification

Two additional markers were attached adjacent to the original markers (markers 3 & 7) for simulating the occurrence of outliers, as shown in Figure 5.8. The measurements of these markers were treated as those of the original markers. As an experimental

verification, the outlier diagnosis was applied and markers 3 & 7 were identified, as shown in Tables 5.5 and 5.6.



**Figure 5.8:** Markers for simulating outliers and the original markers.

**Table 5.5:** Indicators of Compatibility for 7 Markers

Indicators of compatibility for rigid body constraints ( $c_{i,j}$ )							
	Point 1	Point 2	Point 3	Point 4	Point 5	Point 6	Point 7
Point 1	1	1	-1	1	1	1	-1
Point 2	1	1	-1	1	1	1	-1
Point 3	-1	-1	1	-1	-1	-1	-1
Point 4	1	1	-1	1	1	1	-1
Point 5	1	1	-1	1	1	1	-1
Point 6	1	1	-1	1	1	1	-1
Point 7	-1	-1	-1	-1	-1	-1	1

**Table 5.6:** Iterations of Merit Scores for Outlier Diagnosis by Relaxation (Two Outliers)

Merit scores for point $i$ ( $s_i$ )							
Iterations	Point 1	Point 2	Point 3	Point 4	Point 5	Point 6	Point 7
0 - initial	5	5	1	5	5	5	1
1	23	23	-25	23	23	23	-25
2	165	165	-115	165	165	165	-115
3	1055	1055	-825	1055	1055	1055	-825

## 5.4 Summary

For robust pose estimation from outlier contaminated 3D marker measurements, a novel outlier diagnosis method has been developed using a relaxation of rigid body constraints. Instead of accommodating outliers in a pose estimator, the presented outlier diagnosis

method preprocesses measured data first to detect and remove outliers prior to further pose estimation.

The rigid body constraint indicates that marker distances of a rigid body should remain statistically constant after motion. The constraint is chosen as the compatibility criteria of marker measurements. First, the diagnosis procedure by relaxation begins with a replacement of rigid body constraints with integers. Second, a merit score is defined for each marker to represent the possibility of being outliers. Third, an iterative calculation of merit scores is carried out based on the compatibility criteria. Fourth, outliers are clearly distinguished with negative diverge merit scores after the relaxation. Essentially, relaxation techniques iteratively assign merit scores to mutually constrained markers. It involves iterative, “parallel” adjustment of merit scores for each marker on the basis of local outlier evidence. Finally, a pose transformation is estimated from the remaining marker measurements after removing the detected outliers.

This novel method has advantages of high breakdown point and ease of implementation. Case studies have proved the effectiveness of the method and a breakdown point of more than 0.5, providing that at least three marker measurement matches remain for pose estimation after removing the outliers. Moreover, this method can be easily implemented as long as the observed markers are rigidly moved. Nevertheless, it provides an easy-to-use competitive alternative for robust pose estimation of rigid body motion and offers a simple outlier detection mechanism that can potentially be applied to many other areas.

# **CHAPTER 6 DECORRELATION METHOD FOR MEASUREMENT CALIBRATION**

In this chapter, a novel data decorrelation method is proposed for measurement calibration of a portable optical CMM measuring system. The measured 3D data suffer from correlations due to the principle of the triangulation, as described in Appendix F. This nature causes complexity when a calibration procedure is carried out. For this reason, a data decorrelation method based on multivariate statistical analysis is proposed to preprocess the 3D coordinate measurements and find an optimal configuration of CMM, in such a way that each component of the coordinate can be calibrated individually during calibration process. Consequently, the calibration procedure can be significantly simplified in a less time and memory consuming manner. The proposed method includes the local and global data decorrelation. Evaluation and comparisons of these two approaches have been analyzed with a conclusion on the better performance of the global data decorrelation. The optimal decorrelated configuration based on the results has been identified for the measuring system.

## **6.1 Description of Measurement Calibration**

Photogrammetry is the measurement of three-dimensional object coordinates from stereo photographic pictures [77]. The discipline started in the 1880s, but more recently the advent of digital imaging has dramatically changed the accessibility and application areas of photogrammetry [78]. The nature of digital images makes it well suited for real-time applications. Originally used for the measurement of buildings, and later adopted for aerial mapping until 1970s, now the applications encompass a wide range of engineering,

medicine, geology, archaeology.

Line sensors based photogrammetric measuring systems are recently wide used in optical 3D coordinate measuring machines, especially in modern industry, for its ease of automated integration. With synchronized active targets (also called structured lighting [79]), efficient triangulation with cylindrical lenses and line imaging sensors as well as embedded circuits enables high frame rates of 3D data delivery. However, the 3D data suffer from correlations due to the computation algorithm of triangulation, as explained in Appendix F. Basically, the 3D object coordinates are computed by a spatial intersection of fan beams from the image points through the camera perspective centre into the object space [80]. Thus, each component of the coordinate from the intersection mutually depends on each other's calculation.

For the measuring system, calibration is necessary in order to ensure the accurate 3D measurement. In a calibration procedure, a rigid body target needs to be fabricated as a reference object. A number of point luminous markers, such as infrared LED, are attached to the rigid body. The target needs to be displaced at multiple positions with different depths in the 3D measurement volume. Based on the readings from these markers, calibration could be achieved by either identifying the modeled parameters or building a lookup table. The former method with a physical model has been investigated by many researchers [81], [82], [83], which may undergo linearization errors, correlations, and possibility of unmodeled systematic errors. The latter method using a lookup table is straightforward. It stores the ground truth discrete data and needs interpolation among the consecutive values.

However, the existence of correlation causes complexity when the lookup table is being built. The table needs to be 3D that costs enormous memory throughout the measurement volume, and speed could be sacrificed during the data indexing, interpolation, and transferring. Assuming the data could be decorrelated and each component of the coordinate could be treated individually, one dimensional lookup table would qualify for storing each axis [69]. For this reason, a data decorrelation method is presented based on multivariate statistical analysis to preprocess the 3D measurements, in such a way that the calibration procedure can be significantly simplified in a less time and memory consuming manner.

The network configuration has been defined to represent the geometric arrangement of cameras and targets [79], [84]. A number of literatures have demonstrated the impact of imaging geometry on precision [81]. In this article, the relative configuration of sensors regardless of target is defined as the relative network configuration. For some measuring systems, the relative network configurations are fixed as the whole sensors are packaged into one single unit. In this case, based on the data decorrelation results, the optimal decorrelated configuration of the measuring system with targets can be identified. By setting up with this configuration, the measuring system is capable of providing more accurate data than other configurations. The stability under this configuration is studied as well.

## **6.2 Decorrelation Formulations**

Both local and global data decorrelation are defined and formulated. The two methods employ the eigen-decomposition to explore the structure of the variance-covariance

matrix of the measurement data. The difference between the two is the way how the variance-covariance matrix is formulated. The local method is formulated using each individual marker data, while the global method is formulated using a linear combination of the entire set of marker data. Two approaches are evaluated and compared.

### 6.2.1 Local Decorrelation

Assuming that the 3D position measurement of markers complies with a normal distribution, the probability distribution function (PDF) of this multivariate normal distribution can be expressed as [85]

$$f(X) = (2\pi)^{-\frac{p}{2}} \det(C)^{-\frac{1}{2}} \exp\left(-\frac{(X-\mu)^T C^{-1} (X-\mu)}{2}\right) \quad (6.1)$$

where  $X = (x, y, z)^T$ ;  $p$  is the number of variables, in this case,  $p = 3$ ;  $\mu$  is the  $3 \times 1$  column vector of means; and  $C$  is the  $3 \times 3$  covariance matrix of  $X$ . This distribution can be noted as  $X \sim \mathcal{N}_p(\mu, C)$ , with the mean vector and variance-covariance matrix expressed as

$$\mu = (\mu_x \quad \mu_y \quad \mu_z)^T \quad (6.2)$$

$$C = \begin{bmatrix} \sigma_x^2 & Cov(x, y) & Cov(x, z) \\ Cov(x, y) & \sigma_y^2 & Cov(y, z) \\ Cov(x, z) & Cov(y, z) & \sigma_z^2 \end{bmatrix} = \begin{bmatrix} \sigma_x^2 & \rho_{xy}\sigma_x\sigma_y & \rho_{xz}\sigma_x\sigma_z \\ \rho_{xy}\sigma_x\sigma_y & \sigma_y^2 & \rho_{yz}\sigma_y\sigma_z \\ \rho_{xz}\sigma_x\sigma_z & \rho_{yz}\sigma_y\sigma_z & \sigma_z^2 \end{bmatrix} \quad (6.3)$$

where  $\rho_{xy}$ ,  $\rho_{xz}$ ,  $\rho_{yz}$  are the correlation coefficients among  $x$ ,  $y$ , and  $z$  axes and detailed as

$$\sigma_x^2 = E[(x - \bar{x})^2] = E(x^2) - \bar{x}^2 = \frac{\sum x_i^2}{n} - \frac{(\sum x_i)^2}{n^2} \quad (6.4)$$

$$\begin{aligned} Cov(x, y) &= E[(x - \bar{x})(y - \bar{y})] = E(xy) - \bar{x}\bar{y} \\ &= \frac{\sum (x_i y_i)}{n} - \frac{\sum x_i \sum y_i}{n^2} \end{aligned} \quad (6.5)$$

$$\rho_{x,y} = \frac{Cov(x, y)}{\sigma_x \sigma_y} = \frac{E(xy) - \bar{x}\bar{y}}{\sqrt{[E(x^2) - \bar{x}^2][E(y^2) - \bar{y}^2]}} \quad (6.6)$$

From the eigen-decomposition,  $C$  can be factorized as [10]

$$C = VDV^{-1} \quad (6.7)$$

where the columns of  $V$  are the eigenvectors of  $C$ , i.e.,  $V = (v_1 \ v_2 \ v_3)$ ;  $D$  is a diagonal matrix, with eigenvalues as its diagonal elements, i.e.,  $D = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$ .  $C$  is a symmetric positive definite matrix, which has an orthonormal basis of eigenvectors.

Therefore,  $V$  is an orthogonal matrix, and (6.7) becomes

$$C = VDV^T \quad (6.8)$$

By substituting (6.8) into the exponential part of (6.1), it yields

$$\begin{aligned} (X - \mu)^T C^{-1} (X - \mu) &= (X - \mu)^T (VDV^T)^{-1} (X - \mu) \\ &= (V^T X - V^T \mu)^T D^{-1} (V^T X - V^T \mu) \\ &= (X' - \mu')^T D^{-1} (X' - \mu') \end{aligned} \quad (6.9)$$

where

$$X' = V^T X, \quad \mu' = V^T \mu \quad (6.10)$$

From (6.10),  $X'$  is the new set of coordinate values from converting  $X$  by  $V^T$ . Equation (6.9) can be viewed as an ellipsoid defining the surface of constant density of distribution

$$\begin{aligned} d^2 &= (X - \mu)^T C^{-1} (X - \mu) = (X' - \mu')^T D^{-1} (X' - \mu') \\ &= \frac{(v_1^T x - v_1^T \mu_1)^2}{\lambda_1} + \frac{(v_2^T y - v_2^T \mu_2)^2}{\lambda_2} + \frac{(v_3^T z - v_3^T \mu_3)^2}{\lambda_3} \\ &= \frac{(x' - \mu'_1)^2}{\lambda_1} + \frac{(y' - \mu'_2)^2}{\lambda_2} + \frac{(z' - \mu'_3)^2}{\lambda_3} \end{aligned} \quad (6.11)$$

where  $d$  is a real constant scalar,  $V^T$  is considered as the rotation matrix, rotating the axes of the ellipsoid to be aligned with the principle  $x$ ,  $y$ , and  $z$  axes. Note that no scaling happens during the rotational transformation, since  $V^T$  is orthogonal. Refer to (6.9), the distribution of the obtained data  $X'$  can be noted as  $X' \sim \mathcal{N}_3(\mu', D)$ , whose covariance

matrix is a diagonal matrix without any cross-correlations. Also, the principle axial directions of ellipsoids correspond to the eigenvectors; the principle lengths are proportional to the eigenvalues.

In the light of the relative rotation between sensors and markers, we could transform the coordinate frame of sensors instead of markers' coordinates. The new frame of the measuring system, denoted as  $e'$ , can be described by [69]

$$e'X' = eX \quad (6.12)$$

where

$$e' = (v_1 \quad v_2 \quad v_3), \text{ and } e = (e_x \quad e_y \quad e_z) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

With the obtained eigenvectors or the rotation matrix, the sensors could be placed in such an optimal configuration that the 3D coordinate measurements of markers would not suffer from the correlations. The coordinate frames of sensors are aligned with the principle directions of PDF ellipsoids of markers, and the coordinates can be treated independently. For multiple markers, however, since the normal distributions and variance-covariance matrices for each one of them are different, the eigenvalues and eigenvectors are different accordingly. In other words, the optimal configurations for different markers are different as well.

### 6.2.2 Global Decorrelation

As mentioned before, the optimal decorrelated configurations of the sensors vary with different markers. A common optimal sensor configuration is more preferable for the entire set of markers. Therefore, the entire set of marker data could be combined together

to form a global covariance matrix. Then, the eigen-decomposition in (6.8) can be applied to find the global eigenvectors for the decorrelation in (6.10). This kind of decorrelation is regarded as the global data decorrelation method, while the above method for each individual marker is the local data decorrelation method.

Consider a linear combination of the coordinate vectors of markers [86],

$$Y = a_1 X_1 + a_2 X_2 + \cdots + a_n X_n = a_1 \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + a_2 \begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} + \cdots + a_n \begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix} \quad (6.13)$$

where the  $i$ th marker's vector is  $X_i = (x_i, y_i, z_i)^T$ , with a normal distribution of  $\mathcal{N}_p(\mu_i, C_i)$ ,  $p = 3$ , as detailed in (6.1)–(6.3); and  $n$  is the number of markers.

When combining the vectors as a column vector,  $X = (X_1, \dots, X_n)^T$ , the combination in (6.13) can be expressed as

$$Y_{p \times 1} = AX = (a_1 \mathbf{I} \quad a_2 \mathbf{I} \quad \cdots \quad a_n \mathbf{I})_{p \times np} \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix}_{np \times 1} \quad (6.14)$$

where  $A = (a_1 \mathbf{I}, a_2 \mathbf{I}, \dots, a_n \mathbf{I})$ , and  $\mathbf{I}$  is the  $3 \times 3$  identity matrix. The covariance or correlations of the combination  $Y$  are considered as the global covariance or correlations; while the local covariance or correlations are defined for  $X_i$ . The  $np \times 1$  column vector  $X$  has a multivariate normal distribution of  $\mathcal{N}_{np}(\mu, C)$ , where  $\mu = (\mu_1 \quad \mu_2 \quad \cdots \quad \mu_n)^T$  and

$$C = \begin{bmatrix} C_1 & Cov(X_1, X_2) & \cdots & Cov(X_1, X_n) \\ Cov(X_1, X_2) & C_2 & \cdots & Cov(X_2, X_n) \\ \vdots & \vdots & \ddots & \vdots \\ Cov(X_1, X_n) & Cov(X_2, X_n) & \cdots & C_n \end{bmatrix} \quad (6.15)$$

Then, the distribution of the combination  $Y$  can be determined by  $\mathcal{N}_p(A\mu, ACA^T)$ , i.e.,

$\mathcal{N}_p(\sum_{i=1}^n a_i \mu_i, \sum_{i=1}^n \sum_{j=1}^n (a_i a_j \text{Cov}(X_i, X_j)))$ , where  $\text{Cov}(X_i, X_j) = C_i$ .

Statistically, assuming these vectors  $X_i$  are mutually uncorrelated, i.e.,  $\text{Cov}(X_i, X_j) = 0$ ,

Equation (6.15) becomes

$$C = \begin{bmatrix} C_1 & 0 & \cdots & 0 \\ 0 & C_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C_n \end{bmatrix} \quad (6.16)$$

Then the distribution of  $Y$  becomes  $\mathcal{N}_p(\sum_{i=1}^n a_i \mu_i, \sum_{i=1}^n (a_i^2 C_i))$ . Specifically, when  $a_i = 1/\sqrt{n}$ ,  $i$

$= 1, \dots, n$ , or the vectors are uniformly combined, the distribution becomes  $\mathcal{N}_p(\frac{1}{\sqrt{n}} \sum_{i=1}^n \mu_i,$

$\frac{1}{n} \sum_{i=1}^n C_i)$ . In this case, the components of the global covariance matrix are the mean values

of those of each marker. Substituting with (6.3), it yields

$$\text{Cov}(Y) = \frac{1}{n} \sum_{i=1}^n C_i = \frac{1}{n} \begin{bmatrix} \sum_{i=1}^n \sigma_{xi}^2 & \cdots & \vdots \\ \sum_{i=1}^n \text{Cov}(x_i, y_i) & \sum_{i=1}^n \sigma_{yi}^2 & \vdots \\ \sum_{i=1}^n \text{Cov}(x_i, z_i) & \sum_{i=1}^n \text{Cov}(z_i, y_i) & \sum_{i=1}^n \sigma_{zi}^2 \end{bmatrix} \quad (6.17)$$

where

$$\frac{1}{n} \sum_{i=1}^n \sigma_{xi}^2 = \frac{1}{m \times n} \left( \sum_{i=1}^m (x_{1i} - \bar{x}_1)^2 + \cdots + \sum_{i=1}^m (x_{ni} - \bar{x}_n)^2 \right) \quad (6.18)$$

and  $m$  is the number of measurements for each marker.

Applying the eigen-decomposition, the global covariance matrix in (6.17) can be diagonalized and the eigenvectors  $V$  can be obtained. Also, the global correlation coefficients of  $x$  and  $y$  for the combination can be calculated by

$$\begin{aligned}
\rho(x, y) &= \frac{Cov(x, y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^n Cov(x_i, y_i) / n}{\sqrt{\sum_{i=1}^n \sigma_{x_i}^2 / n} \cdot \sqrt{\sum_{i=1}^n \sigma_{y_i}^2 / n}} = \frac{\sum_{i=1}^n Cov(x_i, y_i)}{\sqrt{\sum_{i=1}^n \sigma_{x_i}^2} \cdot \sqrt{\sum_{i=1}^n \sigma_{y_i}^2}} \\
&= \frac{\sum_{i=1}^n [E(x_i, y_i) - \bar{x}_i \bar{y}_i]}{\sqrt{\sum_{i=1}^n \sigma_{x_i}^2} \cdot \sqrt{\sum_{i=1}^n \sigma_{y_i}^2}} = \frac{\sum_{i=1}^n E(x_i, y_i) - \sum_{i=1}^n \bar{x}_i \bar{y}_i}{\sqrt{\sum_{i=1}^n \sigma_{x_i}^2} \cdot \sqrt{\sum_{i=1}^n \sigma_{y_i}^2}}
\end{aligned} \tag{6.19}$$

From (6.6) & (6.19), when  $\rho(x, y) = 0$ , or  $\sum E(x_i, y_i) = \sum \bar{x}_i \bar{y}_i$ , it does not necessarily mean  $\rho(x_i, y_i) = 0$ , or  $E(x_i, y_i) = \bar{x}_i \bar{y}_i$ ; but not vice versa.

### 6.3 Evaluation and Comparison

For a comparison, the local and global data decorrelation methods are evaluated in terms of the root mean square (RMS) values of the local correlations coefficients of  $X$ , described as

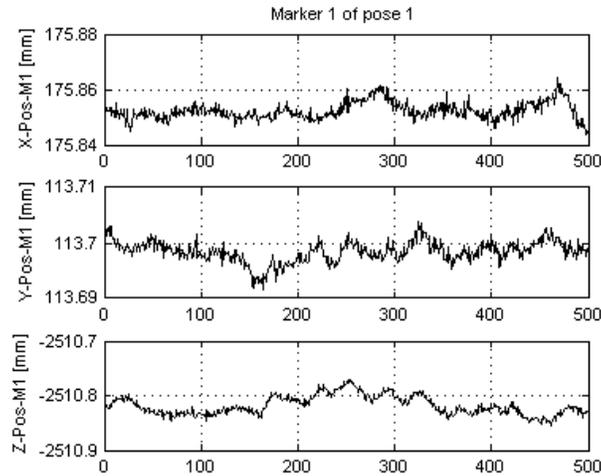
$$CorrCoef\_RMS = \sqrt{\sum_{i=1}^N \sum_{j=1}^N (\rho_{ij}^2)} \tag{6.20}$$

where  $\rho_{ij}$  is the markers' local correlation coefficients, with  $\{i, j \in \mathbb{R}^1 \mid 1 \leq i, j \leq N, i \neq j, N = n \times p\}$ .

### 6.4 Experiments and Simulations

As an experimental validation, the data decorrelation of the 3D position readings from the Optotrak 3020 system located at AMTC was conducted with the proposed method. As shown in Figure 5.3 & 5.4, the experiment system consisted of a Motoman UPJ robot, the Optotrak system, and a target of a cubic rigid body, which was designed to form a cubic structure. In the experiment, 7 markers (infrared LED) were attached to the rigid body.

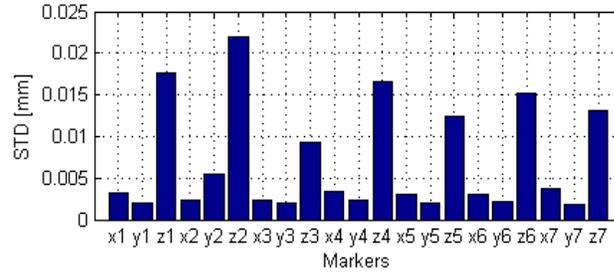
The locations are the 7 corners of the cube. The Optotrak device has three line CCD sensors packaged into one single unit. The relative network configuration is fixed and arranged in a way for the spatial intersection. The maximum frame rate of the system is  $3500/(n+1)$ , about 437 Hz in this case ( $n = 7$ ).



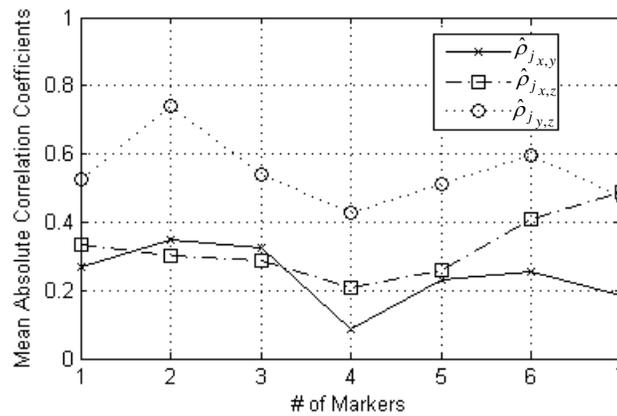
**Figure 6.1:** 500 repetitive readings of marker\_1s' constant 3D position at pose\_1.

The repetitive position measurements of these markers were recorded for 5 seconds at a frame rate of 100Hz at the five different poses of the robot. Figure 6.1 shows the 500 measurements of marker\_1 at pose\_1. From these data, variance and correlation of coordinates have been calculated, with the maximum amounts of variance at  $z$  axis (the viewing or depth direction) and correlation at  $y$  and  $z$  axes. Figure 6.2 shows the standard deviations and the average values on  $x$ ,  $y$ , and  $z$  axes are 0.003, 0.0025, and 0.0151 mm, respectively. Clearly, the value of deviation on  $z$  axis is about 5 - 6 times of those on  $x$  and  $y$  axes, since  $z$  value is calculated upon  $x$  and  $y$  values and the errors are accumulated. The average values of the correlation coefficients of  $xy$ ,  $xz$ , and  $yz$  axes are 0.2431, 0.3274, and 0.5443, respectively. Wing-shaped curve of the correlations has been found as shown in Figure 6.3. All of these indicate that 3D point measurement noises are

usually anisotropic (direction dependent) and heteroscedastic or inhomogeneous (point dependent), due to the nature of 3D sensors.

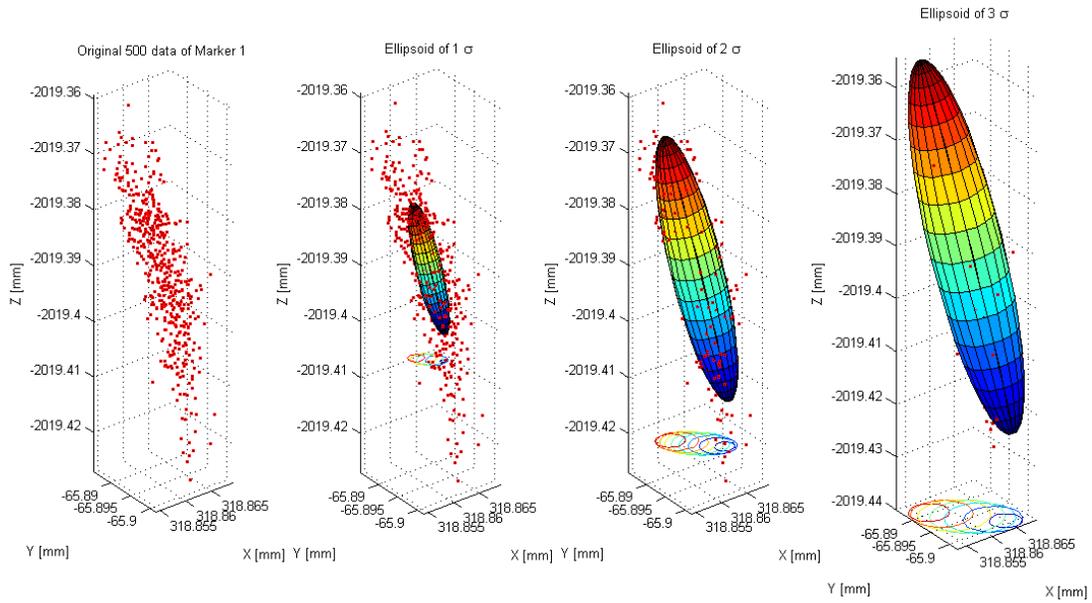


**Figure 6.2:** Standard deviation (STD) of marker's positions.

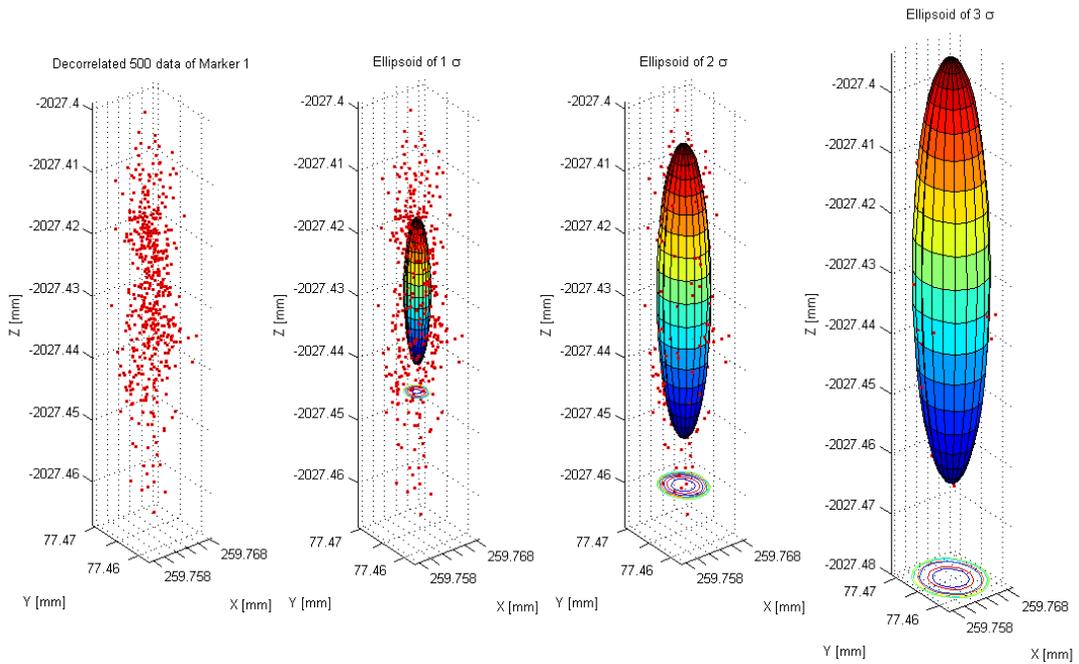


**Figure 6.3:** Mean absolute correlation coefficients of 5 poses for 7 markers.

Figure 6.4(a) and (b) show the comparison of ellipsoids of  $1\sigma$ ,  $2\sigma$ ,  $3\sigma$  of the original and the decorrelated data. As mentioned, the principle axial directions and lengths of ellipsoids are defined by the data's covariance matrix. In Figure 6.5(a), the local decorrelated configurations of NDI sensors are shown in red, with respect to the corresponding ellipsoids of the markers. The original configuration is shown in cyan. Note that there are slightly differences of NDI configurations for each marker according to the different eigenvectors. Figure 6.5(b) shows the mean and standard deviation values before and after the local data decorrelation.

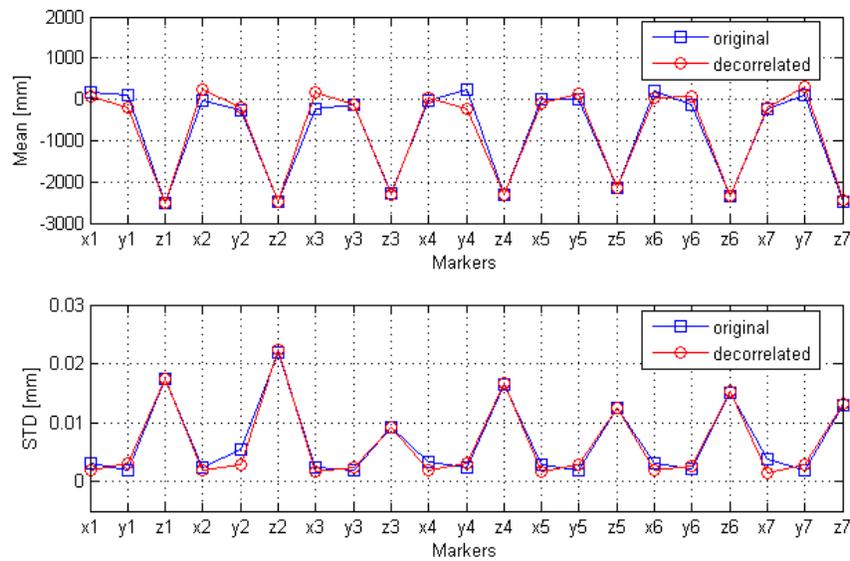
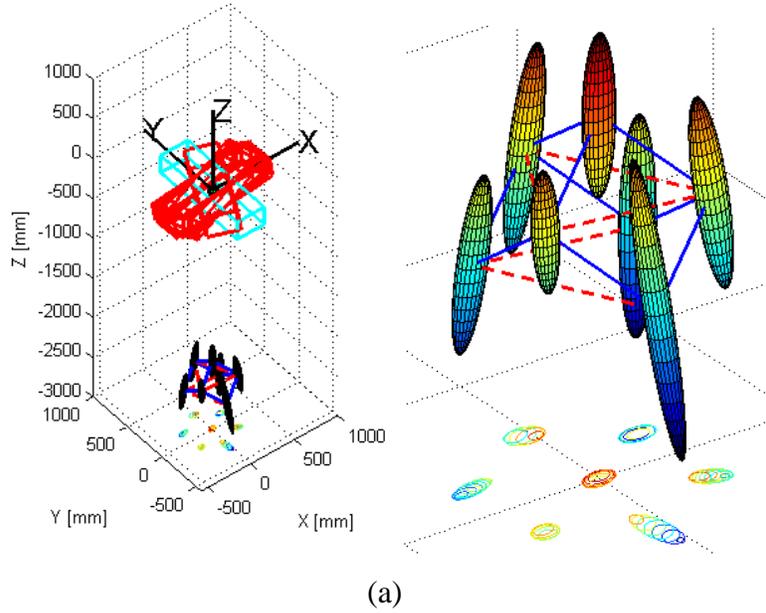


(a)



(b)

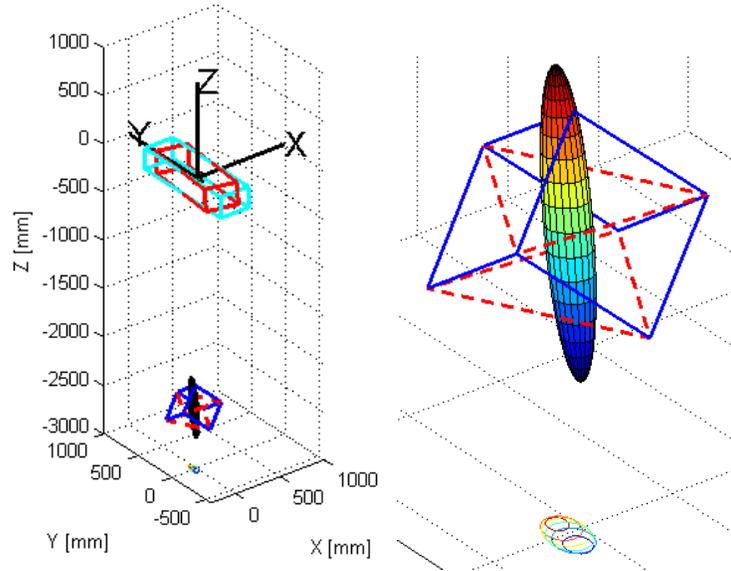
**Figure 6.4:** Comparison of ellipsoids of  $1\sigma$ ,  $2\sigma$ ,  $3\sigma$  of the original (a) and the decorrelated data (b).



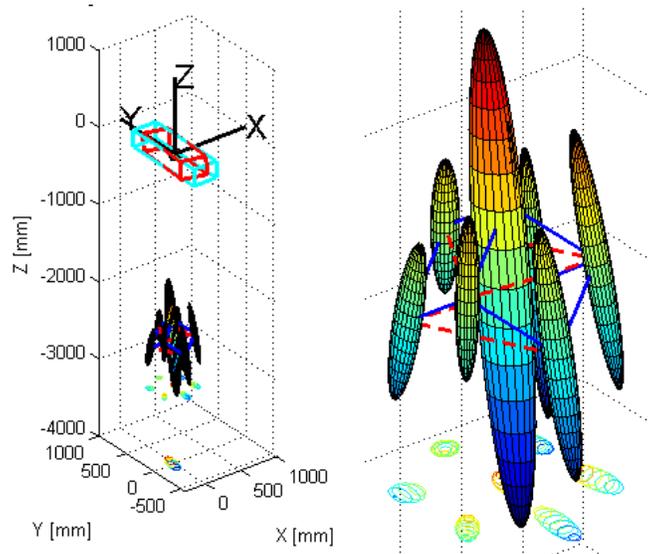
**Figure 6.5:** (a) Local optimal decorrelated configurations of the NDI Optotrak system (in red) with respect to the PDF ellipsoids of 7 markers, the original NDI configuration is in cyan; (b) mean and STD of measurements before and after the local decorrelation.

The global data decorrelation is carried out with the assumption of whether the measurements of markers are mutually uncorrelated or not, as shown in Figure 6.6. It is shown that when mutually correlated in Figure 6.7, the ellipsoid has a larger size than that

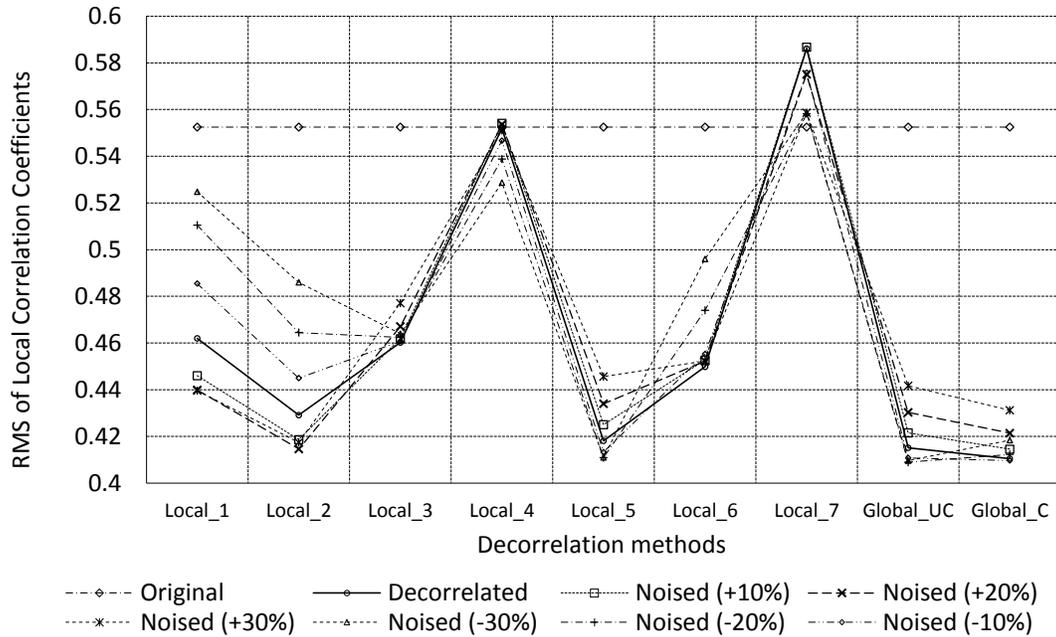
of Figure 6.6, due to the larger eigenvalues when the covariance matrix accounts for the mutual correlations.



**Figure 6.6:** The global decorrelated configuration of NDI for the entire set of markers (in red), when the marker data  $X_i$  are mutually uncorrelated.



**Figure 6.7:** Results when the data of the 7 markers are mutually correlated, with a larger size of ellipsoid due to larger eigenvalues.



**Figure 6.8:** Local and global data decorrelation comparisons, with various noisy levels from  $\pm 10\%$  to  $\pm 30\%$ . “Local\_1” to “Local\_7” mean the seven local data decorrelation methods at each markers. “Global\_UC” represents the global data decorrelation when mutually uncorrelated; “Global\_C” indicates the global data decorrelation when mutually correlated.

Figure 6.8 shows the RMS values for the local and global data decorrelation methods. Various noisy levels,  $\pm 10\%$ ,  $\pm 20\%$ , and  $\pm 30\%$  of the sensor configurations are applied for the sensitivity evaluating purpose. In Figure 6.8, “Local\_1” to “Local\_7” represent the local data decorrelation methods at the 1<sup>st</sup> to the 7<sup>th</sup> markers, respectively. Thus, there are totally 7 local decorrelation methods. “Global\_UC” stands for the global decorrelation when  $X_i$  are mutually uncorrelated; while “Global\_C” stands for the global decorrelation when  $X_i$  are mutually correlated. Without noise, the RMS values for the 7 local and 2 global methods are 0.462, 0.4291, 0.4602, 0.5519, 0.4181, 0.4498, 0.5861, 0.4151, and 0.4105, respectively. Clearly, the global decorrelation methods have lower values compared with the local methods. The global method when  $X_i$  are mutually correlated has

the lowest value, corresponding to the fact of existence of the mutual correlations of  $X_i$ . Also, noises have impact for all of methods, especially for some of the local methods, such as local\_1 and local\_2. As a conclusion, the global decorrelation method when  $X_i$  are mutually correlated is the best solution to perform the data decorrelation and obtain the optimal decorrelated sensor configuration.

## 6.5 Summary

For eliminating the correlations of the multiple markers' readings, the local and global data decorrelation methods have been presented based on the multivariate statistical analysis. For both methods, the eigen-decomposition is employed to explore the structure of the variance-covariance matrix and then the eigenvectors are assigned as the rotation matrix to align the principle axial directions of the ellipsoids, which are the surfaces of constant density of distribution. The difference is how to formulate the variance-covariance matrix. The local method formulates it from the each individual marker data; while the global method formulates it from the linear combination of the entire set of markers data. Evaluation and comparison under various noisy levels has proved a better performance of the global data decorrelation method in terms of the root mean squares values of the local correlation coefficients of markers. The RMS value has been reduced from 0.5525 to 0.4105, with a percentage of about 25.7%. The optimal decorrelated configuration for the measuring system has been identified based on the results. In such a configuration, the calibration of 3D measurements could be simplified by using one dimensional lookup tables.

# CHAPTER 7 CONCLUSIONS AND FUTURE WORKS

## 7.1 Conclusions

In this dissertation, pose estimation has been studied for position-based visual servoing with an application in robotic percussive riveting. The challenges of accuracy in position-based visual servoing include localization errors, poor absolute positioning accuracy of the robot, and measurement errors. For this reason, three categories of pose estimation have been investigated pertaining to these challenges, namely, static, dynamic, and robust pose estimation.

In static pose estimation, direct solutions of relative pose between two rigid bodies estimated from noisy point measurements are derived based on least-square methods. The inequality of conventional and direct LS solutions, and the equivalence of conventional and direct CTLS solutions have been found. Also, the results of static pose estimation are applied in localization of robotic riveting. In dynamic pose estimation, state estimation of relative pose between two rigid bodies have been developed based on iterative extended Kalman filter. Measurement errors have been tackled by outlier diagnosis for robust pose estimation, and a decorrelation method for measurement calibration. For measurement calibration, a decorrelation method is proposed for finding an optimal measurement configuration at which calibration can be conducted for each axis of measurement frame with less effect of correlation. The methods of pose estimation have been validated with simulations and experiments of a robotic riveting system.

## 7.2 Contributions

- Static, dynamic, robust pose estimation were systematically studied for precise rivet-in-hoe insertion of the developed robotic percussive riveting system.
- Different formulations of static pose estimation from points have been summarized, namely, Cartesian frame formulation, three point method, normalized directional vectors, and covariance matrix method. Normalized directional vectors was chosen due to no effect of point distribution.
- Direct solutions of static relative pose estimation based on least-square methods have been derived with and without the constraint of orthogonality for rotation matrix. The direct solution with constraint has been proved to be identical to the conventional indirect method. However, the direct solution without the constraint has been proved to be different from the conventional solution.
- Static pose estimation have been extended for localization of multiple frames of rigid body systems, including articulated rigid bodies, i.e., robot manipulators. A virtual target was proposed to align the robot frame and measurement frame. The virtual target consists of multiple points of the TCP of the robot at different poses within the workspace.
- Dynamic relative pose estimation has been shown to address a state estimation of relative poses between two rigid body during motion. The method tackles the issue of poor absolute positioning accuracy of the robot by estimating relative pose between robot TCP and working panel, which has been verified in simulation and experiments.
- For robustness against outliers of point measurements, an outlier diagnosis is shown based on the relaxation of rigid body constraints of points. This novel method has advantages of high breakdown point and ease of implementation. The method has been verified by both simulation and experiments.
- A decorrelation method is proposed for measurement calibration using multivariate statistical analysis to find an optimal sensor-to-target configuration. As a result, each

coordinate measurement is close to uncorrelated and it allows a simple calibration using one dimensional lookup table.

### **7.3 Future Works**

In static pose estimation, four formulations may be compared more thoroughly in terms of estimation accuracy. Also, the effect of number of points on static pose estimation accuracy may be of interest to be evaluated. For the considered relative pose problem, the direct formulation does not offer any redundancy in the provided input that might be exploited further in order to obtain an even optimal solution in future.

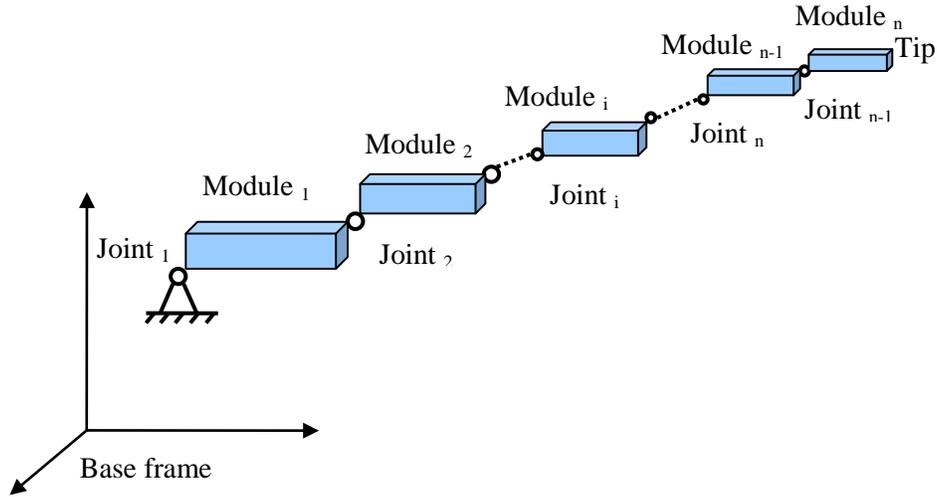
In dynamic pose estimation, the IEKF based algorithm was based on known shape of the targets, i.e., availability of  $A'$  and  $B'$ , homogeneous noise, assumption that the targets are within field of view and not features occlusion happens. As a future work, the algorithm may be extended to provide a general solution to the relative pose state estimation. In addition, more experimental data may need to be carried out to ensure the more reliable performance of the proposed algorithm, by considering longer trials with more challenging motion sequences. Also, a comparison of this work with respect to the single rigid body pose estimation algorithms might be of interest to be evaluated.

For the decorrelation method for measurement calibration, although real measurement data were used in the decorrelation experiment, it would be interesting to actually build lookup tables after the measurement system is reoriented at the global optimal configuration.

To fully integrate the proposed methodologies, software development of an overall interface needs to be further carried out with details that can be found in Appendix G.

## APPENDIX A - POSITION AND ORIENTATION

In this section, the position vector and rotation matrix are introduced to represent the pose of each module.



**Figure A.1:** Multiple modules system for a serial robot manipulator.

### A.1 Position Vector

The position of a point in space is represented with respect to a coordinate frame using a vector. In general, the vector components in the Cartesian coordinate are expressed as

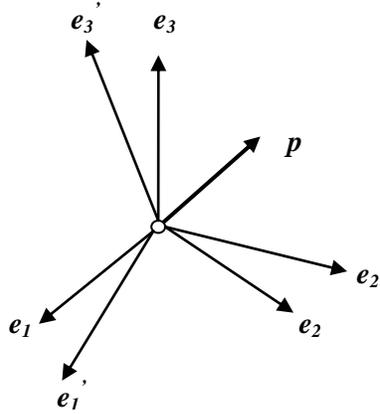
$$p = [p_1, p_2, p_3]^T \quad (\text{A.1})$$

A position vector, as shown in Figure A.2, can be expressed in terms of the frame axes in linear combination as

$$p = p_1 e_1 + p_2 e_2 + p_3 e_3 \quad (\text{A.2a})$$

or 
$$p = E p \quad (\text{A.2b})$$

where  $E = [e_1, e_2, e_3]$ . In the Cartesian coordinate system,  $E = [x, y, z]$ , and  $x, y, z$  are the unit vectors along  $x, y,$  and  $z$  axes respectively, that is,  $x = [1, 0, 0]^T$ ,  $y = [0, 1, 0]^T$ ,  $z = [0, 0, 1]^T$ .



**Figure A.2:** Position vector  $p$  [71].

## A.2 Rotation Matrix

A rotation matrix represents a linear transformation between two coordinate frames. In Figure A.2, position vectors can be expressed either in frame  $\{e_1, e_2, e_3\}$  (Equation A.2a) or frame  $\{e'_1, e'_2, e'_3\}$ ,

$$p = p'_1 e'_1 + p'_2 e'_2 + p'_3 e'_3 \quad (\text{A.3a})$$

or 
$$p = E p' \quad (\text{A.3b})$$

Since Equations (A.2) and (A.3) represent the same position vector, so

$$E' p' = E p \quad (\text{A.4})$$

It leads to

$$p = R p' \quad (\text{A.5})$$

where  $R$  is the rotation matrix given as

$$R = E'^T \cdot E \quad (\text{A.6})$$

Since  $e_i$  is orthogonal to  $e_j$ , then  $e_i \cdot e_j = \delta_{ij} = 1$  when  $i=j$ ;  $e_i \cdot e_j = 0$  when  $i \neq j$ . Hence,  $E$  is orthogonal, and  $E^{-1} = E^T$ .

$R$  is in fact defined by the dot product of two unit vectors, i.e., the direction cosine. It is

also called the tensor product, defined as

$$\mathbf{R} = (\mathbf{E} \otimes \mathbf{E}') = \begin{bmatrix} \mathbf{e}_1 \cdot \mathbf{e}'_1 & \mathbf{e}_1 \cdot \mathbf{e}'_2 & \mathbf{e}_1 \cdot \mathbf{e}'_3 \\ \mathbf{e}_2 \cdot \mathbf{e}'_1 & \mathbf{e}_2 \cdot \mathbf{e}'_2 & \mathbf{e}_2 \cdot \mathbf{e}'_3 \\ \mathbf{e}_3 \cdot \mathbf{e}'_1 & \mathbf{e}_3 \cdot \mathbf{e}'_2 & \mathbf{e}_3 \cdot \mathbf{e}'_3 \end{bmatrix} \quad (\text{A.7})$$

Reversing the order of Equation (A.4), Equation (A.5) becomes

$$\mathbf{p}' = \mathbf{R}'\mathbf{p} \quad (\text{A.8})$$

where

$$\mathbf{R}' = \mathbf{E}'^T \mathbf{E} \quad (\text{A.9})$$

Obviously

$$\mathbf{R}' = \mathbf{R}^T = \mathbf{R}^{-1} \quad (\text{A.10})$$

Hence,  $\mathbf{R}$  is orthogonal, and such that all columns are mutually orthogonal and have unit magnitude. In fact, it is proper orthogonal, meaning  $\det(\mathbf{R}) = 1$ .

It is clear that the nine elements of a rotation matrix are not all independent. Six dependencies or constraints between the elements can be easily found from a given rotation matrix,  $\mathbf{R} = [\mathbf{X} \ \mathbf{Y} \ \mathbf{Z}]$ :

$$\begin{aligned} |\mathbf{X}| &= 1 \\ |\mathbf{Y}| &= 1 \\ |\mathbf{Z}| &= 1 \end{aligned} \quad (\text{A.11a})$$

$$\begin{aligned} \mathbf{X} \cdot \mathbf{Y} &= 0 \\ \mathbf{X} \cdot \mathbf{Z} &= 0 \\ \mathbf{Y} \cdot \mathbf{Z} &= 0 \end{aligned} \quad (\text{A.11b})$$

As a result, three independent parameters representation is developed in the following section in order to express the rotation matrix conveniently, using the angle-set convention.

### A.3 Angle-set Representation of a Rotation

There are basically two methods to describe the orientation of a frame relative to a reference frame by angle-set conventions. According to Craig [94], one is Euler angles, and one is fixed angles. In the former representation, each rotation is performed about an axis of the moving coordinate system rather than one of the fixed reference frame. For the latter one, each of the three rotations takes place about an axis in the fixed reference frame.

For each method, 12 sets of conventions are employed according to different sequence of rotation about the X-Y-Z axes. One of them is introduced in details in this section, and the rest can be found in [94]. Usually, there is no particular reason to favor one convention over another, but various authors adopt different ones [94]. The following PRY angles (or X-Y-Z Euler angles) are applied in this thesis.

#### **(Tait) Bryan Angles (Pitch Roll Yaw, PRY) [or X-Y-Z Euler angles]**

In terms of pitch, roll, and yaw angle (PRY) [71], the three individual rotation matrices can be given as:

$$\mathbf{R}(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix} \quad (\text{A.12})$$

$$\mathbf{R}(\theta_y) = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix} \quad (\text{A.13})$$

$$\mathbf{R}(\theta_z) = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{A.14})$$

Then, the resulting rotation matrix in the global reference frame is given as:

$$\mathbf{R} = \mathbf{R}(\theta_x)\mathbf{R}(\theta_y)\mathbf{R}(\theta_z) \quad (\text{A.15})$$

If the order is reversed, it will become the rotation matrix in the local frame

$$\mathbf{R}^T = \mathbf{R}^T(\theta_z)\mathbf{R}^T(\theta_y)\mathbf{R}^T(\theta_x) \quad (\text{A.16})$$

Expanding Equation (A.15) leads to

$$\mathbf{R} = \begin{bmatrix} c\theta_y c\theta_z & -c\theta_y s\theta_z & s\theta_y \\ s\theta_x s\theta_y c\theta_z + c\theta_x s\theta_z & -s\theta_x s\theta_y s\theta_z + c\theta_x c\theta_z & -s\theta_x c\theta_y \\ -c\theta_x s\theta_y c\theta_z + s\theta_x s\theta_z & c\theta_x s\theta_y s\theta_z + s\theta_x c\theta_z & c\theta_x c\theta_y \end{bmatrix} \quad (\text{A.17})$$

When given three PRY angles, Equation (A.17) can be used to compute the rotation matrix directly. For the reverse problem, some elements in the Equation (A.17) are selected to determine the PRY angles for a given a rotation matrix, for example:

$$\begin{aligned} \theta_y &= \cos^{-1}\left(\left[\left(r_{23}\right)^2 + \left(r_{33}\right)^2\right]^{1/2}\right) \\ \theta_x &= \cos^{-1}\left(r_{33}/\cos\theta_y\right) \\ \theta_z &= \cos^{-1}\left(r_{11}/\cos\theta_y\right) \end{aligned} \quad (\text{A.18})$$

where  $r_{23} = -\sin(\theta_x)\cos(\theta_y)$ ,  $r_{33} = \cos(\theta_x)\cos(\theta_y)$ ,  $r_{11} = \cos(\theta_y)\cos(\theta_z)$ .

Different selection of elements from the given rotation matrix leads to different method for a solution. Bai and Teo [95] developed another solution using  $\text{atan2}(y, x)$ , a two-argument arctangent function that uses the signs of both  $x$  and  $y$  to identify the quadrant

in which the resulting angle lies:

$$\begin{aligned}\theta_x &= \text{atan2}(-r_{23}, r_{33}) \\ \theta_y &= \text{atan2}(r_{13}, -r_{23}\sin(\theta_x) + r_{33}\cos(\theta_x)) \\ \theta_z &= \text{atan2}(r_{21}\cos(\theta_x) + r_{31}\sin(\theta_x), r_{22}\cos(\theta_x) + r_{32}\sin(\theta_x))\end{aligned}\tag{A.19}$$

However, none of them is capable of solving this inverse problem with a unique solution from a rotation matrix in all four quadrants, which is called the quadrant sensitivity problem. Equation (A.18) is only valid for  $0 \leq \theta_y \leq 90^\circ$ ,  $0 \leq \theta_x \leq 180^\circ$ , and  $0 \leq \theta_z \leq 180^\circ$ ; while Equation (A.19) is valid when  $\theta_y$  lies in the first and fourth quadrants.

In other words, if  $\theta_y$  is located in the second and third quadrants, the values of  $\cos(\theta_y)$  would become negative. As a result, the two elements  $r_{23} = -\sin(\theta_x)\cos(\theta_y)$  and  $r_{33} = \cos(\theta_x)\cos(\theta_y)$  in Equation (A.19) change signs, and the values of  $\theta_x$  determined from  $\text{atan2}(r_{23}, r_{33})$  are no longer true.

So by modifying Equation (A.19),

$$\cos\theta_y = \sqrt{r_{23}^2 + r_{33}^2} = \sqrt{(-s\theta_x c\theta_y)^2 + (c\theta_x c\theta_y)^2} = \sqrt{c\theta_y^2}\tag{A.20}$$

then  $\theta_y$  can also be determined by

$$\theta_y = \text{atan2}(r_{13}, \sqrt{r_{23}^2 + r_{33}^2})\tag{A.21}$$

Even though a second solution exists, by using the positive square root in Equation (A.20) for  $\theta_y$ , we always can compute the single solution for which  $-90^\circ \leq \theta_y \leq 90^\circ$ , making it a one-by-one mapping orientation representation.

Attention should also be paid to the names pitch, roll and yaw angles, since they are often given to other related but different angle-set conventions; for instance, it is referred to the X-Y-Z fixed angles in Craig [94].

## APPENDIX B - GENERAL MOTION

This section presents the forward kinematic modeling of a serial robot with rigid bodies. The topic is to compute the position and orientation of the end-effector relative to the base coordinate as a function of the joint variables.

### B.1 General Motion of a Single Rigid Body

As shown in Figure B.1, the general motion of a single rigid body is the combination of rotation and translation, and the position vector  $p$  is

$$p = Rb' + h = h + Rb' \quad (\text{B.1})$$

where  $h$  is the vector of translation,  $b'$  is the body vector in local coordinate system, and  $R$  is the rotation matrix.

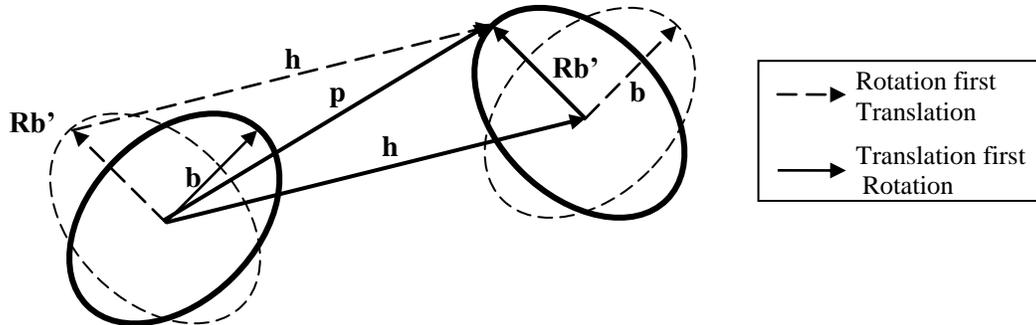


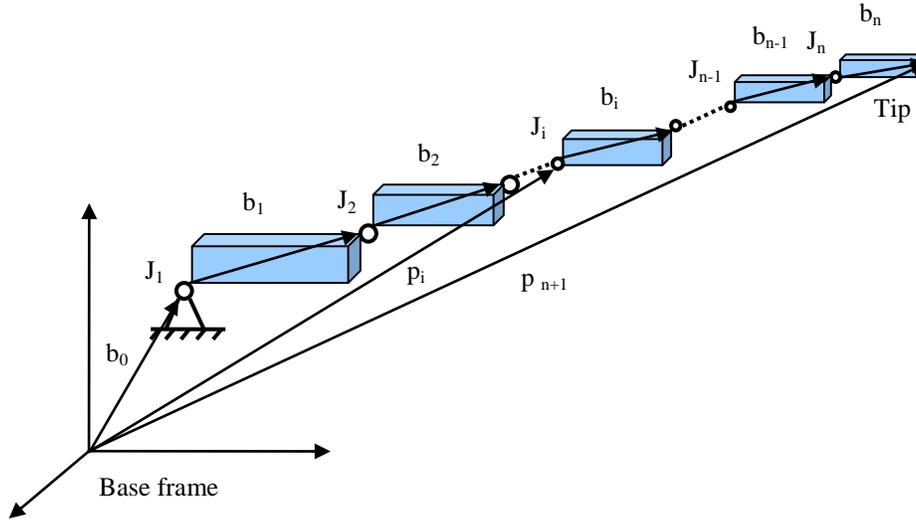
Figure B.1: General motion of a single module [71].

Clearly, Equation (B.1) is commutative, meaning the order of rotation and translation can be reversed, which can also be found in Figure B.1. When  $h$  is null, it becomes pure rotation.

### B.2 General Motion of Multiple Rigid Bodies

Figure B.2 shows the vector method used to compute the position of a multi-module

system.



**Figure B.2:** Vector method for a multi-module system [71]

The position of each joint from 1, 2 to  $i$  can be expressed respectively as:

Joint 1                     $p_1 = b_0$

Joint 2                     $p_2 = b_0 + R_{01}b'_1 = p_1 + R_{01}b'_1$

Joint 3                     $p_3 = b_0 + R_{01}(b'_1 + R_{12}b'_2) = p_1 + R_{01}R_{12}b'_2$

.....

Joint  $i$                      $p_i = b_0 + R_{01}(b'_1 + \dots + R_{i-1,i}b'_i) = p_{i-1} + R_{0,i-1}b'_{i-1}$                     (B.2)

where  $R_{i-1,i}$  defines the rotation between two coordinate systems attached to two adjacent modules  $i-1$  and  $i$ ;  $b'_i$  is the local body vector, representing the translation between two coordinate systems, or defining the  $i$ th joint to the  $(i+1)$ th joint in the  $i$ th local coordinate frame.

Clearly, Equation (B.2) is a recursive method for computing the position of a multi-module system. Similarly, the recursive method of computing rotation can be given as

$$R_{0i} = R_{01}R_{12}\dots R_{i-1,i} = R_{0i-1}R_{i-1,i} \quad (B.3)$$

Hence, in general, the pose (position and orientation) of the end-effector of a n-module system can be expressed as [71]

$$\text{Position} \quad \mathbf{p}_{n+1} = \sum_{i=0}^n \mathbf{R}_{0i} \mathbf{b}'_i = \sum_{i=0}^n \mathbf{b}_i \quad (\text{B.4})$$

$$\text{Orientation} \quad \mathbf{R}_{0n} = \prod_{j=1}^n \mathbf{R}_{(j-1)j} \quad (\text{B.5})$$

As for a robot system, it usually has the default home configuration or initial configuration setup. Therefore, it should be noted that all the employed parameters here may have static part and motion part. The static part is according to the initial configuration setup, and the motion part represents the movement of each joint.

$$\mathbf{R} = \mathbf{R}_s \mathbf{R}_m \quad (\text{B.6})$$

$$\mathbf{b} = \mathbf{b}_s + \mathbf{b}_m \quad (\text{B.7})$$

where  $\mathbf{R}_s$  and  $\mathbf{b}_s$  are initial configuration setup, which are the geometric parameters need to identify; and  $\mathbf{R}_m$  and  $\mathbf{b}_m$  are related to active joints, i.e., motors. The static part can be further expressed by the PRY angles rotation as

$$\mathbf{R}_s = \mathbf{R}_x \mathbf{R}_y \mathbf{R}_z \quad (\text{B.8})$$

$$\mathbf{b}_s = b_x \mathbf{x} + b_y \mathbf{y} + b_z \mathbf{z} \quad (\text{B.9})$$

where  $\mathbf{R}_x$ ,  $\mathbf{R}_y$ , and  $\mathbf{R}_z$  are the rotation about x, y and z axis of the configuration setup;  $b_x$ ,  $b_y$ , and  $b_z$  are the translation along the x, y, and z axe of the configuration setup.

In terms of different kinematic pairs, they may be expressed differently according to joint movements, as shown in Table B.1. As for robotics, usually only revolute and prismatic joints are considered in reality. In Table B.1,  $\mathbf{R}(\theta_z)$ ,  $\mathbf{R}(\theta_y)$ , and  $\mathbf{R}(\theta_x)$  are the rotation about z, y, and x axis of the joint respectively; while  $\mathbf{s}_z$  is the translation along z axis of the

joint. Conventionally, the first physical rotation or translation axis of a joint should be defined as axis z, second as y, and last as x. It should be noted this convention is totally different from the sequence of PRY angel set, which is used to represent orientation instead of a rotation matrix.

**Table B.1:**  $R_m$  and  $b_m$  of different kinematic pairs [71].

Joint	$R_m$	$b_m$
Revolute	$R(\theta_z)$	0
Prismatic	$R(0) = 1$	$s_z$
Cylinder	$R(\theta_z)$	$s_z$
Universal	$R(\theta_z)R(\theta_y)$	0
Spherical	$R(\theta_z) R(\theta_y) R(\theta_x)$	0

# APPENDIX C - MATRIX COMPUTATIONS

## C.1 The Rank of a Matrix

A set of vectors  $x_1, \dots, x_n$  is said to be linearly independent if  $\sum \alpha_i x_i = 0$  implies all  $\alpha_i = 0$ . If  $x_1, \dots, x_n$  are not linearly independent, they are said to be linearly dependent.

Let  $A$  be an  $m \times n$  matrix. The column rank of  $A$  is the maximum number of linearly independent columns it contains. The row rank of  $A$  is the maximum number of linearly independent rows it contains. It may be shown that the column rank of  $A$  is equal to its row rank. Hence the concept of rank is unambiguous. We denote the rank of  $A$  by [96]

$$r(A). \tag{C.1}$$

It is clearly that

$$r(A) \leq \min(m, n) \tag{C.2}$$

If  $r(A) = m$ , we say that  $A$  has full row rank.

If  $r(A) = n$ , we say that  $A$  has full column rank.

We have the following important properties:

$$r(A) = r(A^T) = r(AA^T) = r(A^T A) \tag{C.3}$$

$$r(AB) \leq \min(r(A), r(B)) \tag{C.4}$$

$$r(AB) = r(A), \text{ if } B \text{ is square of full rank} \tag{C.5}$$

$$r(A + B) \leq r(A) + r(B) \tag{C.6}$$

The column space of  $A$ , denoted  $\mathcal{M}(A)$ , is the set of vectors

$$\mathcal{M}(A) = \{y: y = Ax \text{ for some } x \text{ in } \mathbb{R}^n\}. \tag{C.7}$$

## C.2 Singular Value Decomposition

Let  $A$  be a real  $m \times n$  matrix with  $r(A) = r > 0$ . Then there exists an  $m \times r$  matrix  $U$  such that  $U^T U = I_r$ , an  $n \times r$  matrix  $V$  such that  $V^T V = I_r$  and an  $r \times r$  diagonal matrix  $\Sigma$  with positive diagonal elements, such that [96]

$$A = U\Sigma V^T. \quad (\text{C.8})$$

## C.3 The Moore-Penrose (MP) Inverse

The Moore-Penrose inverse was introduced by Moore (1920, 1935) and rediscovered by Penrose (1955). There is a huge literature on generalized inverses or the Pseudo-Inverse, of which the MP inverse is one particularly useful example due to its uniqueness with minimum Frobenius norm [68].

### Definition

An  $n \times m$  matrix  $X$  is the MP inverse of a real  $m \times n$  matrix  $A$  if

$$\begin{aligned} \text{(i)} \quad AXA &= A & \text{(ii)} \quad XAX &= X \\ \text{(iii)} \quad (AX)^T &= AX & \text{(iv)} \quad (XA)^T &= XA \end{aligned}$$

The MP inverse of  $A$  is denoted as  $A^+$ .

### Theorem 1

For each  $A$ ,  $A^+$  exists and is unique. The unique M-P inverse of  $A = U\Sigma V^T$  is

$$A^+ = V\Sigma^+ U^T = \sum_{i=1}^r \frac{v_i u_i^T}{\sigma_i}, \quad r(A) = r \quad (\text{C.9})$$

It is the unique minimal Frobenius norm solution:  $\min \|AX - I_m\|_F$ .

### Some Properties

$$A^+ = (A^T A)^+ A^T = A^T (A A^T)^+ \quad (\text{C.10})$$

$$A^+ = (A^T A)^{-1} A^T, \text{ if } r(A) = n \quad (\text{C.11})$$

$$A^+ = A^T(AA^T)^{-1}, \text{ if } r(A) = m \quad (\text{C.12})$$

$$A^+ = A^{-1}, \text{ if } m = n = r(A) \quad (\text{C.13})$$

## C.4 The Solution of Linear Equation Systems

An important property of the Moore-Penrose inverse is that it enables us to find explicit solutions of a system of linear equations [96].

### Theorem 2

The general solution of the homogeneous equation  $Ax = 0$  is

$$x = (I_n - A^+A)q, \quad (\text{C.14})$$

where  $q$  is an arbitrary vector of appropriate order. The solution is unique and is  $x = 0$ , if and only if  $r(A) = n$ . If the solution is not unique, then there exist an infinite number of solutions given by (C.14).

### Theorem 3

A necessary and sufficient condition for the vector equation  $Ax = b$  to have a solution is that

$$AA^+b = b, \quad (\text{C.15})$$

in which case the general solution is

$$x = A^+b + (I_n - A^+A)q, \quad (\text{C.16})$$

where  $q$  is an arbitrary vector of appropriate order.

If there exists at least one solution, we say that the system  $Ax = b$  is consistent. The system is consistent for every  $b$  if and only if  $r(A) = m$ . If the system is consistent, its solution is unique if and only if  $r(A) = n$ . Clearly if  $m = n = r(A)$ , then  $A$  is non-singular and the unique solution is  $A^{-1}b$ .

## C.5 Least Squares Method and Gauss-Markov Theorem

### Theorem 4 (Least Squares Method)

Let  $A$  be a given  $m \times n$  matrix, and  $b$  a given  $n \times 1$  vector. Then [96]

$$(A\hat{x} - b)'(A\hat{x} - b) \geq b'(I - AA^+)b \quad (\text{C.17})$$

for every  $\hat{x}$  in  $\mathbb{R}^k$ , with equality if and only if

$$\hat{x} = A^+b + (I - A^+A)q \quad (\text{C.18})$$

for some  $q$  in  $\mathbb{R}^k$ .

*Note.* From Appendix C.3, the solution using M-P inverse of  $A$  has the minimum Euclidean norm among all solutions, given by

$$\hat{x} = A^+b = \sum_{i=1}^r \frac{u_i^T b}{\sigma_i} v_i, \quad r(A) = r \quad (\text{C.19})$$

In the special case where  $A$  has full column rank  $n$ , we have  $A^+ = (A^T A)^{-1} A^T$  and hence a unique vector  $x_*$  exists which minimizes  $(Ax - b)'(Ax - b)$  over all  $x$ , namely

$$x_* = (A^T A)^{-1} A^T b. \quad (\text{C.20})$$

The vector  $\hat{x}$  is called the least squares solution and  $A\hat{x}$  is called the least squares approximation to  $b$ . Thus  $\hat{x}$  is the ‘best’ choice for  $x$  whether the equation  $Ax = b$  is consistent or not. If  $Ax = b$  is consistent, then  $\hat{x}$  is the solution; if  $Ax = b$  is not consistent, then  $\hat{x}$  is the least squares solution.

### Theorem 5 (Gauss-Markov)

Consider the standard linear regression model  $(b, Ax, \sigma^2 I)$ , i.e.

$$b = Ax + \varepsilon, \quad (\text{C.21})$$

where  $A$  has full column rank  $n$  and the disturbances  $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n$  are uncorrelated, i.e.

$$E[\varepsilon] = 0, \quad E[\varepsilon\varepsilon'] = \sigma^2 I_n. \quad (\text{C.22})$$

The best (minimum variance) affine unbiased estimator of exists if, and only if, in which case

$$\hat{x} = (A^T A)^{-1} A^T b = A^+ b \quad (\text{C.23})$$

with variance matrix

$$\text{var}(\hat{x}) = \sigma^2 (A^T A)^{-1}. \quad (\text{C.24})$$

*Note.* Compared with Theorem 4, In other words, the best linear unbiased estimator for  $b$  is the least squares solution of  $b = A\hat{x}$ . Note that the method of least squares, however, is a purely deterministic method which has to do with approximation, not with estimation.

**Example: Least squares solution of  $D_f = RD_o$**

Consider the equation of  $D_f = RD_o$ , where  $R \in \mathbb{R}^{m \times m}$ ,  $r(R) = m$ ,  $D_o$  and  $D_f \in \mathbb{R}^{m \times n}$ ,  $r(D_o) = r(D_f) = m$ . According to Theorem 4, the least squares solution of  $R$  is given by

$$\hat{R} = D_f D_o^+ = D_f D_o^T (D_o D_o^T)^{-1} \quad (\text{C.25})$$

*Proof.* The least-square errors of  $D_f = RD_o$  can be formulated as

$$\|\Delta D_f\|^2 = (\hat{R}D_o - D_f)(\hat{R}D_o - D_f)^T \quad (\text{C.26})$$

By taking the first derivative and equal to zero, we can find the estimated matrix that minimizes the sum of square errors.

$$\frac{\partial \|\Delta D_f\|^2}{\partial \hat{R}} = 2(\hat{R}D_o - D_f)(-D_o^T) = 0 \quad (\text{C.27})$$

Hence, the LS solution of  $R$  is

$$\hat{R} = D_f D_o^T (D_o D_o^T)^{-1} \quad (\text{C.28})$$

where  $D_o^T (D_o D_o^T)^{-1}$  is the pseudo-inverse of  $D_o$  in C.12.

## C.6 Rotation of Subspaces (Constrained Least-Squares)

**Theorem 6:** Suppose  $D_f \in \mathbb{R}^{m \times n}$  and  $D_o \in \mathbb{R}^{m \times n}$  are two sets of data matrix obtained by performing two measurements or experiments.  $m$  is the size of each data or dimension of the subspace and  $n$  is the number of measurements.  $R = VU^T$  is the solution for the *orthogonal Procrustes problem*, which is the possibility that  $D_o$  can be rotated into  $D_f$  is explored by solving the following problem: [68]

$$\text{minimize } \|D_f - RD_o\|_F \quad \text{subject to } R^T R = I_m \quad (\text{C.29})$$

where  $V$  and  $U$  represent the SVD of  $D_o D_o^T$ , i.e.

$$D_o D_o^T = U \Sigma V^T$$

*Proof.* Recall that the Frobenius matrix norm of  $C$  is the trace of a matrix  $CC^T$ , denoted as  $\|C\|_F^2 = \text{tr}(CC^T)$ . It follows that if  $R \in \mathbb{R}^{m \times m}$  is orthogonal, then

$$\begin{aligned} \|D_f - RD_o\|_F^2 &= \text{tr} \left[ (D_f - RD_o)(D_f - RD_o)^T \right] \\ &= \text{tr}(D_f D_f^T) + \text{tr}(D_o D_o^T) - 2\text{tr}(RD_o D_f^T) \end{aligned} \quad (\text{C.30})$$

Thus, is equivalent to the problem of maximizing  $\text{tr}(RD_o D_f^T)$ . If  $D_o D_o^T = U \Sigma V^T$  is the SVD of this matrix, then

$$\text{tr}(RD_o D_f^T) = \text{tr}(RU \Sigma V^T) = \text{tr}(V^T R U \Sigma) \leq \text{tr}(\Sigma) \quad (\text{C.31})$$

Clearly, the upper bound is attained by setting  $R = VU^T$  for then  $V^T R U = I_m$ . Obviously, the minimum value of error norm is

$$\min \|D_f - RD_o\|_F^2 = \|D_f\|_F^2 + \|D_o\|_F^2 - 2\text{tr}(\Sigma) \quad (\text{C.32})$$

# APPENDIX D - LEAST-SQUARE ESTIMATION OF TRANSFORMATION PARAMETERS BETWEEN TWO POINT PATTERNS

Umeyama [40] refined Arun *et. al* [36]'s SVD solution and proved the following theorem for estimating the similarity transformation parameters between two point patterns.

**Theorem:** Let  $X = (x_1, x_2, \dots, x_n)$  and  $Y = (y_1, y_2, \dots, y_n)$  be corresponding point patterns in  $m$ -dimensional space. The minimum value of the mean squared error

$$e^2(R, t, c) = \frac{1}{n} \sum_{i=1}^n \|y_i - (cRx_i + t)\|^2 \quad (\text{D.1})$$

of these two point patterns with respect to the similarity transformation parameters (R: rotation, t: translation, and c: scaling) is given as follows:

$$\varepsilon^2 = \sigma_y^2 - \frac{\text{tr}(DS)^2}{\sigma_x^2} \quad (\text{D.2})$$

where

$$\mu_x = \frac{1}{n} \sum_{i=1}^n x_i \quad (\text{D.3})$$

$$\mu_y = \frac{1}{n} \sum_{i=1}^n y_i \quad (\text{D.4})$$

$$\sigma_x^2 = \frac{1}{n} \sum_{i=1}^n \|x_i - \mu_x\|^2 \quad (\text{D.5})$$

$$\sigma_y^2 = \frac{1}{n} \sum_{i=1}^n \|y_i - \mu_y\|^2 \quad (\text{D.6})$$

$$\Sigma_{xy} = \frac{1}{n} \sum_{i=1}^n (x_i - \mu_x)(y_i - \mu_y)^T \quad (\text{D.7})$$

and let a singular value decomposition of  $\Sigma_{xy}$  be  $U\Sigma V^T$  ( $\Sigma = \text{diag}(d_i)$ ,  $d_1 \geq d_2 \geq \dots \geq d_m \geq 0$ ), and

$$S = \begin{cases} I & \text{if } \det(\Sigma_{xy}) \geq 0 \\ \text{diag}(1, 1, \dots, 1, -1) & \text{if } \det(\Sigma_{xy}) < 0 \end{cases} \quad (\text{D.8})$$

$\Sigma_{xy}$  is a covariance matrix of  $X$  and  $Y$ ,  $\mu_x$  and  $\mu_y$  are mean vectors of  $X$  and  $Y$ , and  $\sigma_x^2$  and  $\sigma_y^2$  are variances around the mean vectors of  $X$  and  $Y$ , respectively.

When  $\text{rank}(\Sigma_{xy}) \geq m - 1$ , the optimum transformation parameters are determined uniquely as follows:

$$R = V^T S U \quad (\text{D.9})$$

$$t = \mu_y - c R \mu_x \quad (\text{D.10})$$

$$c = \frac{\text{tr}(DS)^2}{\sigma_x^2} \quad (\text{D.11})$$

## APPENDIX E - GENERAL FORMULATION FOR THE OUTLIER DIAGNOSIS BY RELAXATION

A function to calculate the total support of non-outlier provided to  $p_i$  by all markers  $p_j$

[87]

$$s_i^k = \frac{1}{n} \sum_{j=1}^n \varphi_j(c(\varepsilon_{ij}), s_j^{k-1}) \quad (\text{E.1})$$

where

$s_i^k$  - Merit score for  $i$ th marker at  $k$ th iteration

$s_j^{k-1}$  - Merit score for  $j$ th marker at  $(k-1)$ th iteration

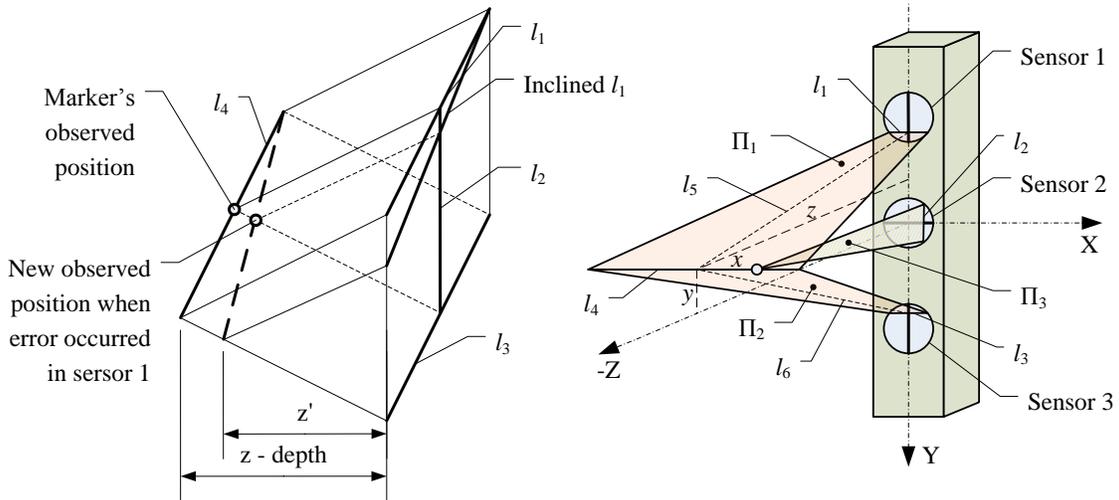
$\varphi_j(c(\varepsilon_{ij}), s_j^{k-1})$  - A function to calculate the support of non-outlier given to  $p_i$  by  $p_j$ , by combining the merit scores of  $p_j$  ( $s_j^{k-1}$ ) and the compatibility  $c(\varepsilon_{ij})$ . These two factors can be combined in various ways. It can be a product, e.g.,  $\varphi_j = c(\varepsilon_{ij}) \cdot s_j^{k-1}$ , or to use their minimum, i.e.,  $\varphi_j = \min(c(\varepsilon_{ij}), s_j^{k-1})$ , or etc.

$c(\varepsilon_{ij})$  - A function to calculate the compatibility between  $p_i$  and  $p_j$ . It can be a continuous function of the error of the Euclidean distance  $\varepsilon_{ij}$ , for example,  $c(\varepsilon_{ij}) = 1/(1 + \varepsilon_{ij}^2)$ , or a discrete function with integers by thresholding,  $c(\varepsilon_{ij}) = \{-1, 0, 1\}$

$\varepsilon_{ij}$  - The error of Euclidean distance between the nominal value and the measured value, defined by the rigid body constraints

## APPENDIX F - MEASURING SYSTEM MODELING

The measuring system is a line-sensor based Optical CMM. The principle of measuring system is illustrated in Figure F.1. The CMM has three line CCD sensors, represented by sensor 1, 2 and 3. At each CCD sensor, there is a projection plane perpendicular to the line sensor that pass through the point feature, the focal and the image point, denoted by  $\Pi_1$ ,  $\Pi_2$ , and  $\Pi_3$ . The intersects of these three projection planes establish the triangulation of the point of interest, whose coordinates are represented as  $p(x, y, z)$ . The modeling basically solves the coordinates based on the triangulation of planes  $\Pi_1$ ,  $\Pi_2$ , and  $\Pi_3$ .



**Figure F.1:** Principle of Optoelectronic Coordinate Measurement Machines [88].

Here are equations of three planes, as shown in Figure 3:

$$\Pi_1 : a_1x + b_1y + c_1z + d_1 = 0 \quad (\text{F.1})$$

$$\Pi_2 : a_2x + b_2y + c_2z + d_2 = 0 \quad (\text{F.2})$$

$$\Pi_3 : a_3x + b_3y + c_3z + d_3 = 0 \quad (\text{F.3})$$

Since planes  $\Pi_1$  and  $\Pi_2$  are parallel to the  $X$  axis and its normal vectors  $\vec{n}_1$  and  $\vec{n}_2$  are vertical to the  $X$  axis, then  $a_1$  and  $a_2$  are zero. When calculating the coordinate values of a marker,  $y$  and  $z$  can be determined first by eqs. (F.1) and (F.2), yields [88]

$$\begin{aligned} l_5 : b_1 y + c_1 z + d_1 &= 0 \\ l_6 : b_2 y + c_2 z + d_2 &= 0 \end{aligned} \quad (\text{F.4})$$

or

$$\begin{bmatrix} b_1 & c_1 \\ b_2 & c_2 \end{bmatrix} \begin{pmatrix} y \\ z \end{pmatrix} = \begin{pmatrix} -d_1 \\ -d_2 \end{pmatrix} \quad (\text{F.5})$$

And substituting  $y$  and  $z$  values into eq. (F.3),  $x$  value can be obtained. Therefore, each component of the coordinates from the calculation mutually depends on each other's calculation. As a result, the coordinates are correlated and require a decorrelation procedure during the calibration of the measurement system prior to the pose estimation.

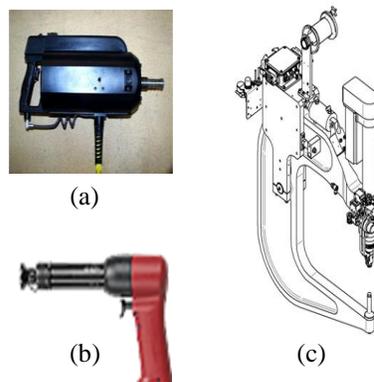
## **APPENDIX G - ROBOTIC PERCUSSIVE RIVETING**

Riveting and welding represent two primary joining methods for the assembly of structural components that require strong joint strength. Compared with welding mainly a fusion method, riveting a mechanical method generates no thermal deformation, hence widely used for joining high thermal conductive materials such as aluminum sheet metals used in aircraft assembly [1]. There are hundred thousands of rivets in a regional aircraft and millions in a large continental aircraft. Overall, the operation of aircraft assembly is divided into three stages: subcomponent assembly, component assembly, and line assembly. The subcomponent assembly is the first step to construct the base components for four major sections, namely, fuselage, wing, cockpit and empennage. The component assembly is the middle step to join the subcomponents to form an individual major section. The line assembly is the last step to assemble a whole aircraft by connecting the four major sections together.

The current riveting process in aerospace manufacturing entails a mix of manual riveting, semi-automated riveting, and automated riveting. The semi-automated and automated riveting machines are widely used in North America and Europe, but only limited to component assembly, such as wing skin panels and fuselage skin panels. Subcomponent assembly and line assembly are still conducted manually. The labor incurred producing these subassemblies/assemblies amounts to as much as fifty percent of the total cost. Manual riveting operations are tedious, repetitious, prone to error, and likely causing health and ergonomic problems [1].

In principle, there are two riveting methods, the first called squeezing (or one-shot)

riveting, where a large upsetting force is applied to deform a rivet instantly. This method requires a large riveter operating under high pressure beyond the yield strength of aluminum rivets in a range over 500 lb force. As shown in Figure G.1 (a), this type of riveter is made of either a hydraulic cylinder or an electromagnetic piston, very heavy, bulky and usually needing a lift-assisted device if used for manual operation. The automated and semi-automated riveting machines employ this type of riveter; hence they are gigantic and only limited to riveting large, simple and relatively flat components. The second method is called percussive (or hammering) riveting, where a small impulsive force is applied to deform a rivet accumulatively by a series of hits. As shown in Figure G.1 (b), this method uses a rivet gun in size of a regular hand-held power tool, very compact and light, operating under much lower pressure in a range less than 100 psi, very safe and energy efficient. Manual riveting employs this principle.



**Figure G.1:** Rivet guns and C-frame riveter.

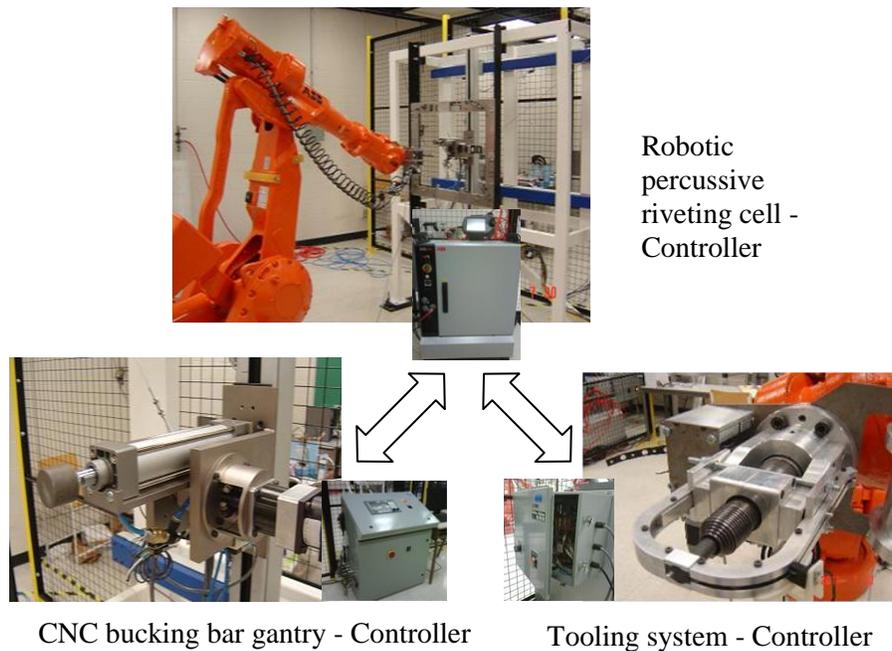
Research on robotic riveting has been mainly centering on squeezing riveting that utilizes heavy-duty industrial robots of large size ( $> 100$  Kg payload). In the automotive industry, squeezing robotic riveting systems have been fully developed and commercialized for joining metal parts. This technology is called robotic self-piercing riveting, in which a C-

frame tooling, as shown in Figure G.1 (c), is designed to have a squeezing riveter mounted on one end as a punch and the other end serving as a hitting base [89]. This system has been widely used for automotive chassis assembly. The application of robotic technology in aerospace manufacturing has been significantly slower than that in automotive manufacturing [90]. Though not commercially available, squeezing robotic riveting systems have been researched in the past by Boeing [91] and recently by EADS in Germany affiliated with AirBus [92]. In addition, a robotic system has been implemented at Bombardier in Montreal that uses two giant Kuka robots to hold large panels that are riveted on a C-frame squeezing riveting machine [2].

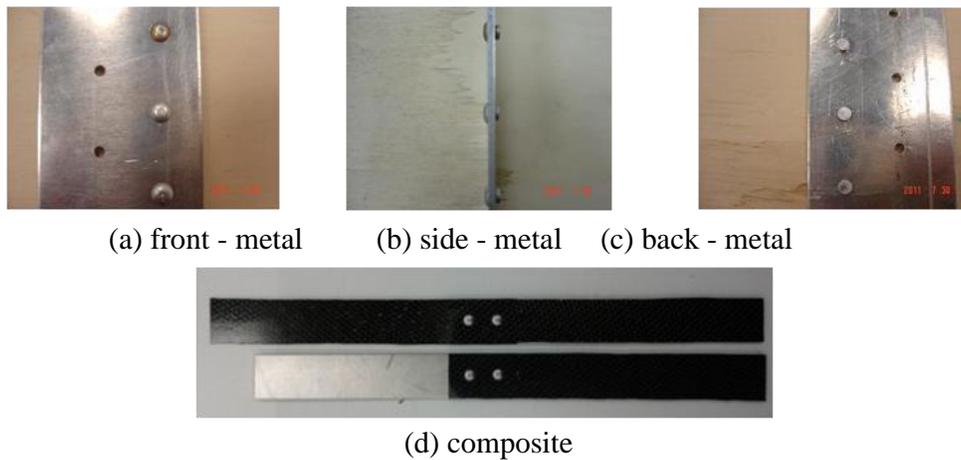
The research results presented in [3] and this dissertation have been applied to implement our robotic percussive riveting system. Figure G.2 shows the physical system involving three controllers for three subsystems: robot, tooling and bucking bar gantry. All controllers are integrated, with the robot controller being the main one for synchronization. The entire riveting process is automated through synchronization. Furthermore, the choice of a gantry system instead of a second robot allows it to serve as a jig for mounting sheet metals. A complete riveting control sequence has been created in Figure G.4, starting from position the gun → position the bucking bar → insert rive → extend the bucking bar → riveting → retract the bucking bar → retract the gun → move to the next spot. This sequence has been successfully tested and implemented to perform percussive riveting on sheet metal and composite panels [93], as shown in Figure G.3.

The software development is undergoing to integrate the methodologies of the proposed pose estimation, as shown in Figures G.5 and G.6. The interface is developed as an add-in button in ABB RobotStudio, which is a software of simulation, offline and online

programming developed by ABB Robotics, Sweden. After performing the localization based on static pose estimation, the transformation results can be typed into the input interfaces, including vision w.r.t. robot, and vision w.r.t. panel, as shown in Figure G.6. The methods of dynamic and robust pose estimation are being programming into the interface as part of robot path guidance, as shown in Figure G.5 (b). The decorrelation method is being programming into the acquisition interface, as shown in Figure G.6(a).



**Figure G.2:** Robotic Percussive Riveting System [3].



**Figure G.3:** Robot Riveted Samples [3].

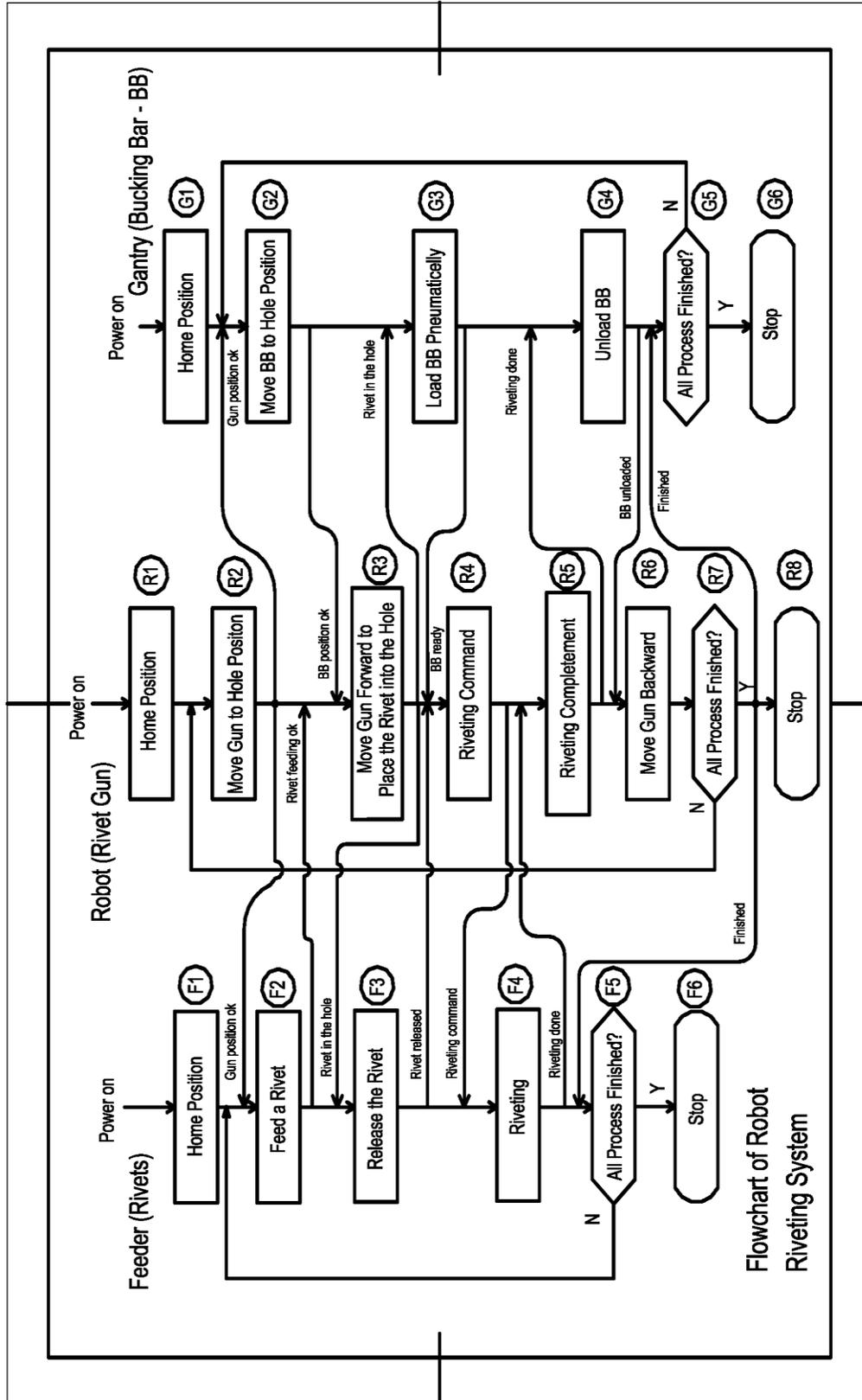
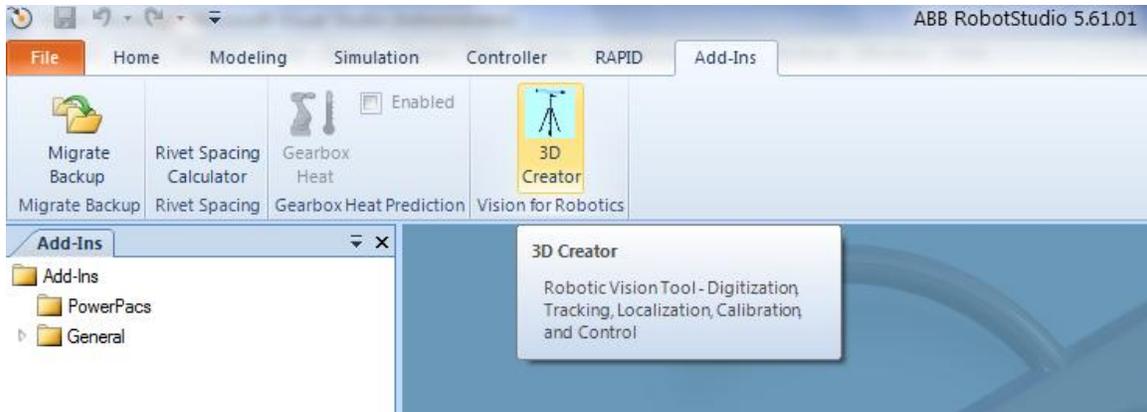
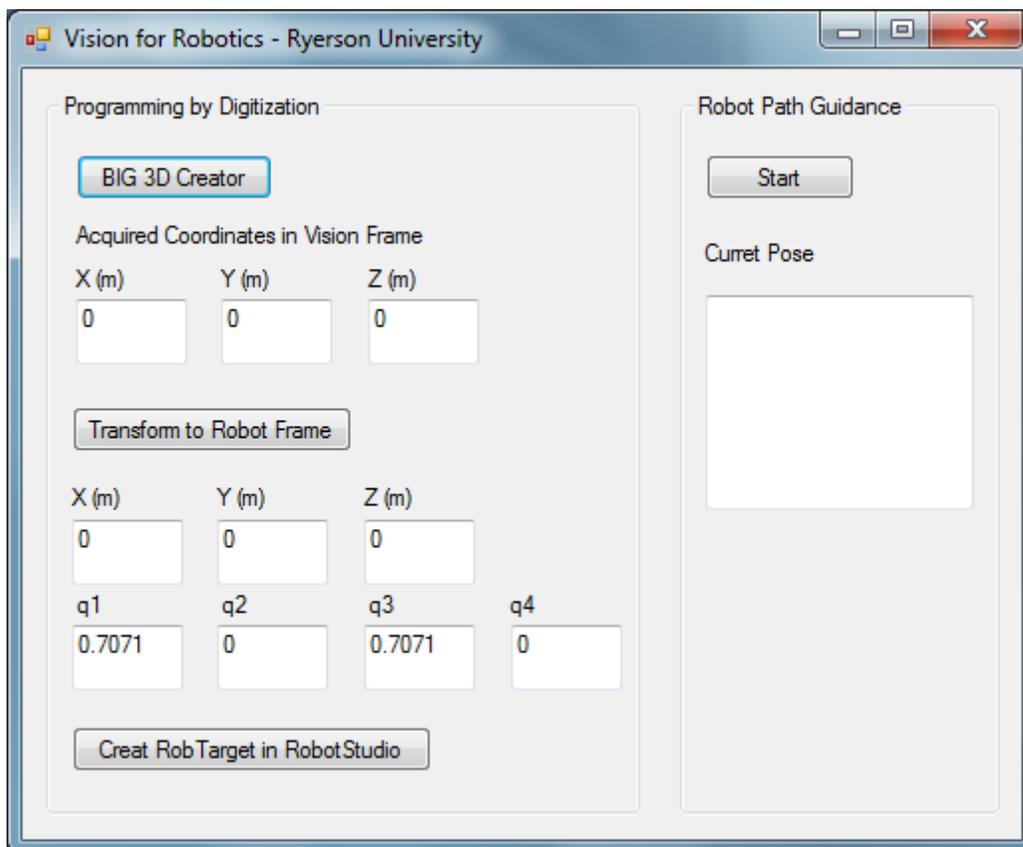


Figure G.4: Flowchart of Process of Robotic Percussive Riveting System.

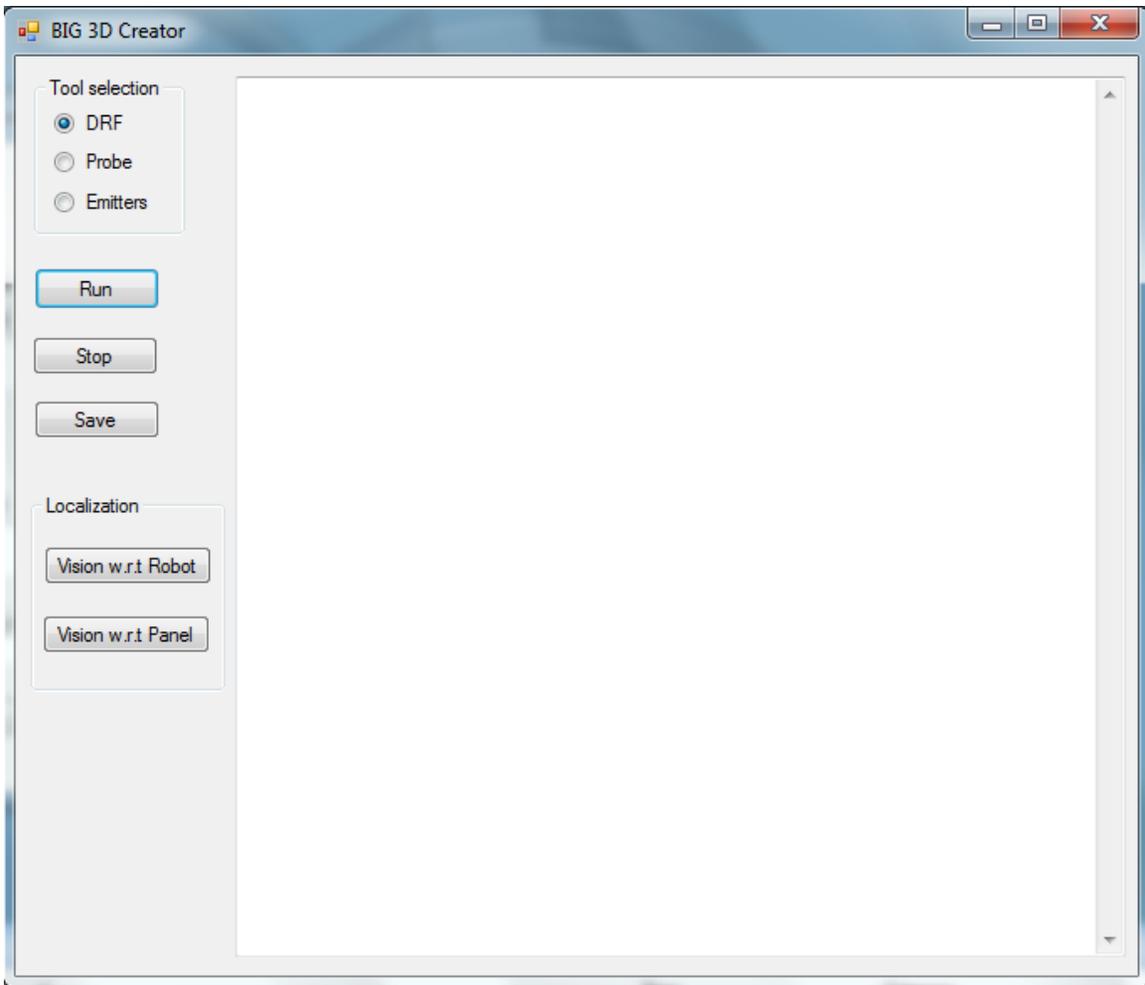


(a) Interface developed as an Add-In button of ABB RobotStudio.

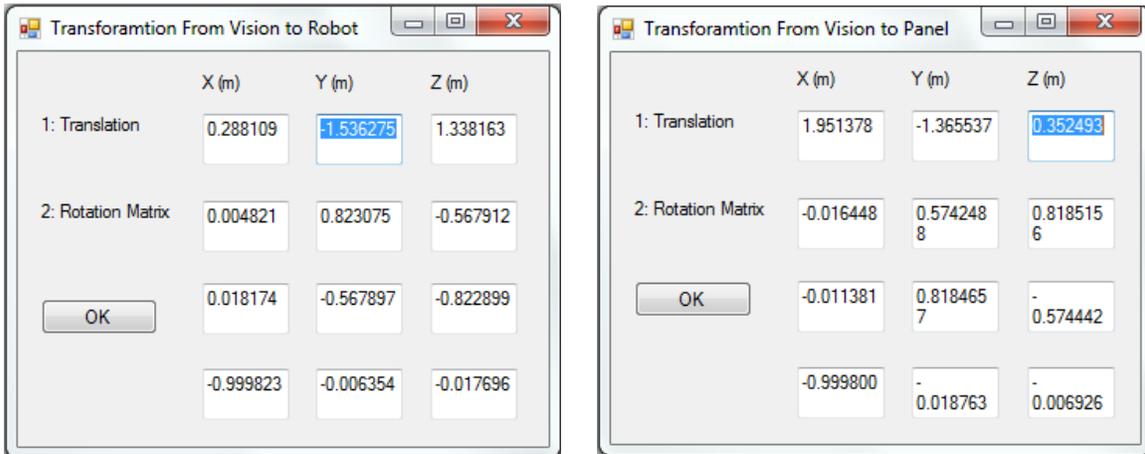


(b) Interface of vision guidance.

**Figure G.5:** Graphical user interface development for robotic riveting system.



(a) Acquisition interface of the Optical CMM (BIG 3D Creator).



(b) Input interfaces for localization.

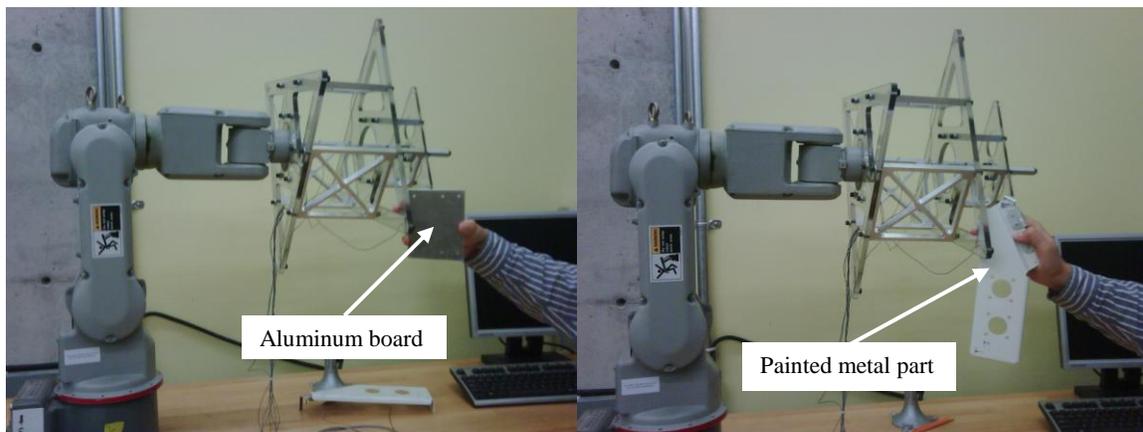
**Figure G.6:** Interfaces for acquisition and localization.

# APPENDIX H - ROBUSTNESS EVALUATING EXPERIMENTS FOR NDI DATA ACQUISITION

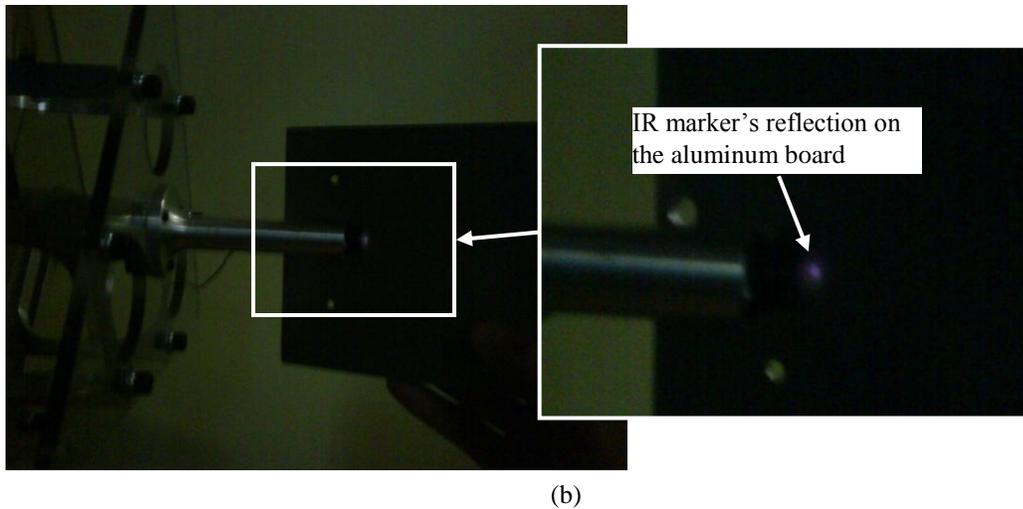
In order to evaluate the robustness of the NDI Optotrak 3020 system, various distractions are considered in the experiments: reflection, illumination (white, laser, infrared), missing data and creating outliers.

## H.1 Reflection Distraction

As depicted in Figure H.1(a), aluminum boards and painted metal parts are tested to prove the existence of reflection of infrared (IR) markers. The reflection on the aluminum board can be clearly noticed in Figure H.1(b). The reflected IR could be misinterpreted as a real marker and its coordinate value could be calculated by the photogrammetry system. However, when the aluminum board is far enough, the reflection seems to be weakened until disappearance. Besides, it depends on the reflecting characteristics of the material. In any circumstances, the reflection issue should be brought to our attentions during metrology integrations in aerospace manufacturing and assembly fields.



(a)



**Figure H.1:** IR marker's reflection on an aluminum board and a painted metal part.

## H.2 Illumination Distraction

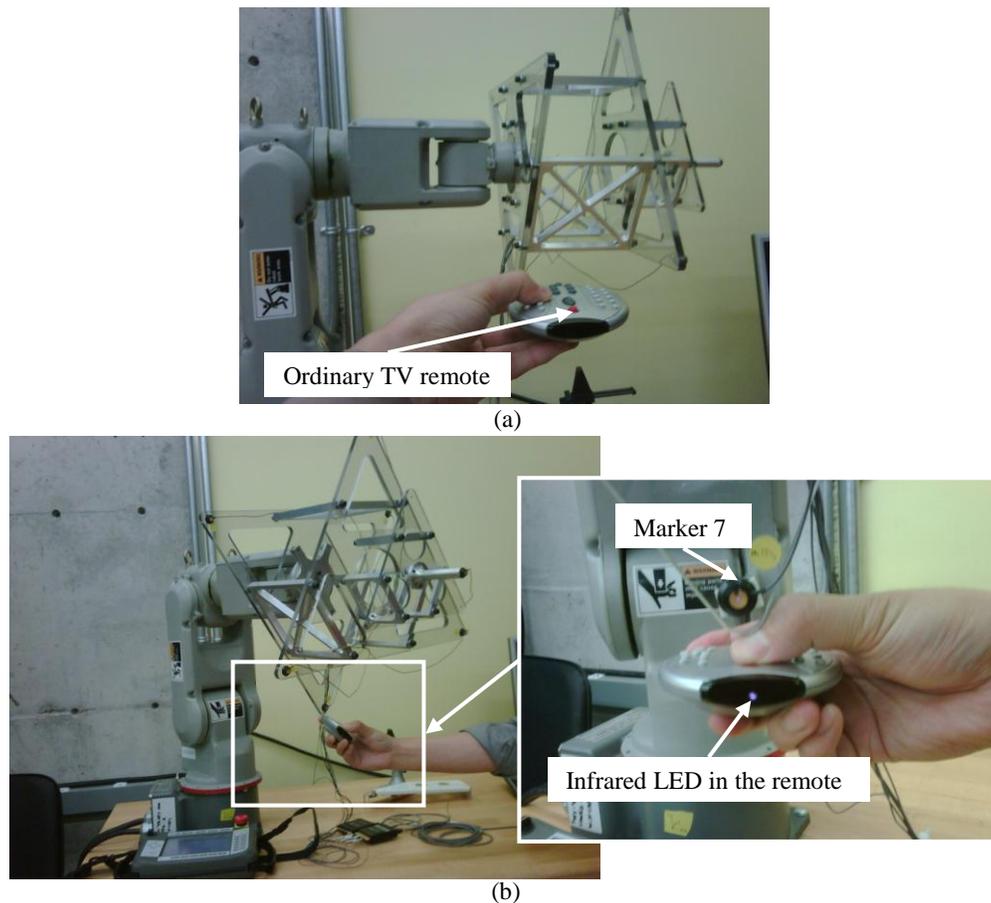
- White/Laser lights

Basically, no distraction caused by white/laser lights was discovered from the experiments, since the wave length of the laser tested is far beyond the IR markers'.

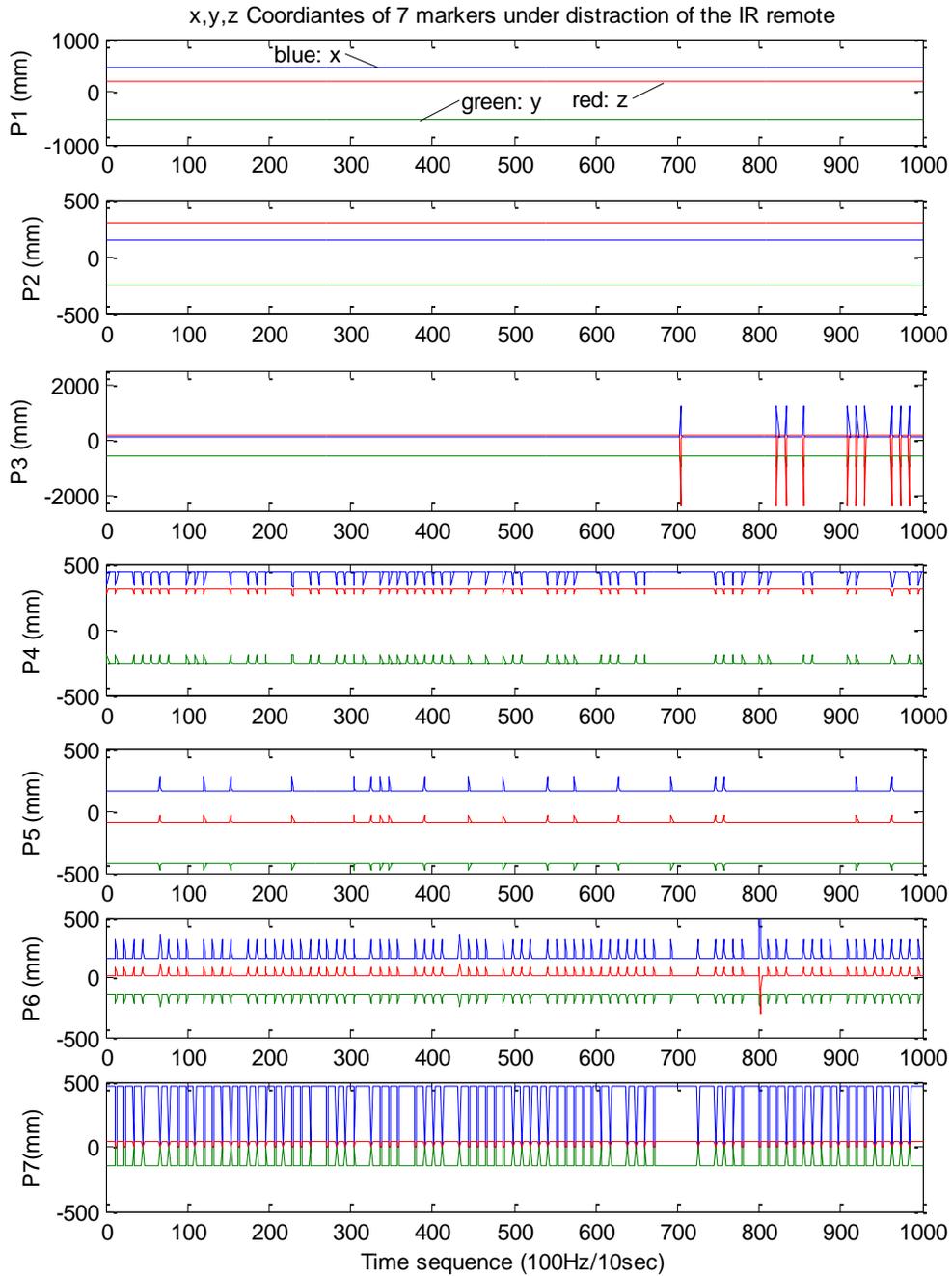
- Infrared Light Distraction

In Figure H.2(a), when the markers are out of sight of NDI cameras, emitting infrared lights from an ordinary TV remote are observed and calculated instead. In Figure H.2(b), during the data acquisition, the remote is pressed down and placed beside one of the markers (marker 7). The distractions of the IR remote can be quantified based on the acquired data, as shown in Figure H.3. Note that the blue, green and red lines represent the  $x$ ,  $y$  and  $z$  coordinates. Clearly, the remote causes missing data at the closest marker (marker 7) and disturbance at the nearby markers (markers 3-6). It may not affect a few

markers that are far from the remote (markers 1&2). The reason behind this phenomenon could be the image extracking algorithm for the line CCD sensors of the NDI cameras. Most importantly, the observed outliers of infrared lights are not carefully filtered due to the lack of recognition of the characteristics of the infrared marker, such as the wave length. As a suggestion, infrared light distraction should be avoided in the metrology environment.



**Figure H.2:** (a) The NDI Optotrak system observes the IR LED lights of the remote instead, when the markers are not observable due to away from the camera sensing direction. (b) Demonstration of Infrared LED remote's distraction during the data acquisition: the remote was pressed down and placed beside a marker (marker 7).



**Figure H.3:** The positions of 7 markers under the distraction of the remote: missing data at marker 7; disturbances at markers 3-6; no effect at markers 1 and 2. The blue, green and red lines represent the x, y and z coordinates.

### H.3 Missing data and outliers

- Missing data

The existence of missing data was achieved by obstructing the visibility of markers, such as swinging an arm between the markers and the camera. The results have shown that the missing data doesn't affect the pose estimation of the rigid body as long as there are at least 3 visible markers remaining for the acquisition and the further pose estimation.

- Outliers

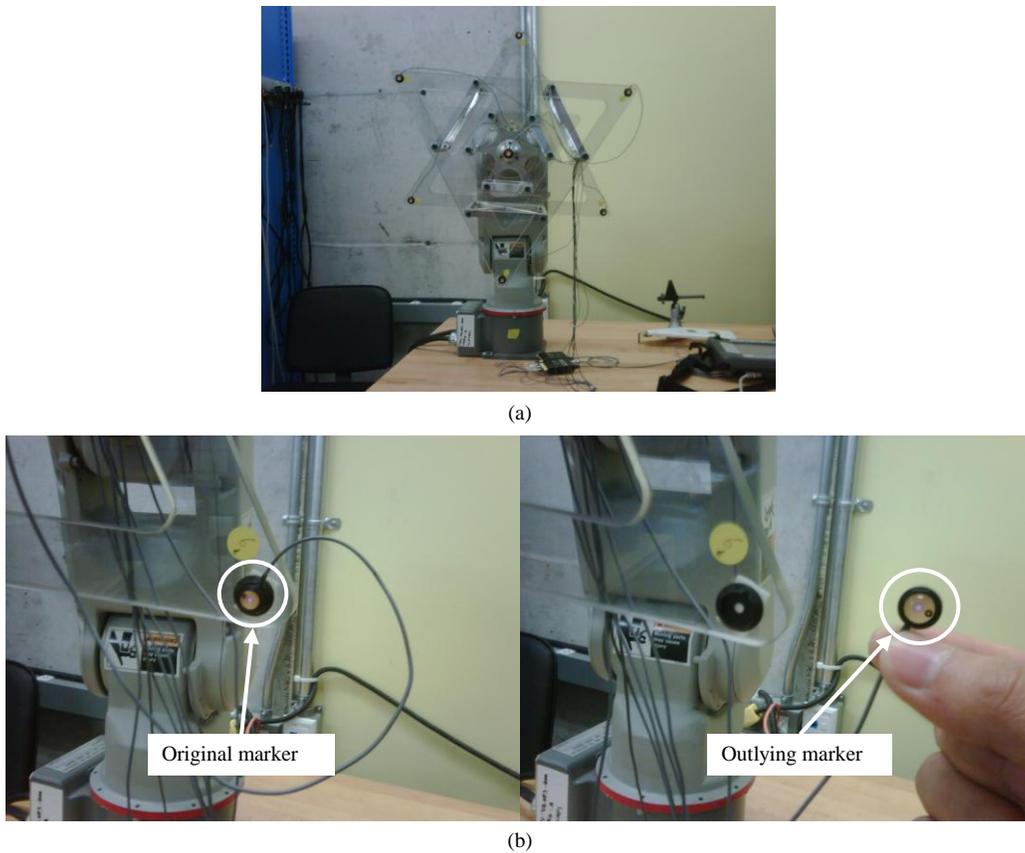
As a definition, the outlier is the gross measurement error that lies outside of the overall distribution of errors. In our case, it could be caused by the false correspondence of markers due to reflection, lighting distraction, sensor failures, or even the matching algorithm, etc.

In order to demonstrate the occurrence of outliers, a marker is detached from the target-mounted case and put at a distance from the original position. Figure H.4(a) shows the original set-ups of 7 markers attached to the cubic target. Figure H.4(b) depicts the way of creating an outlier from the target-mounted case.

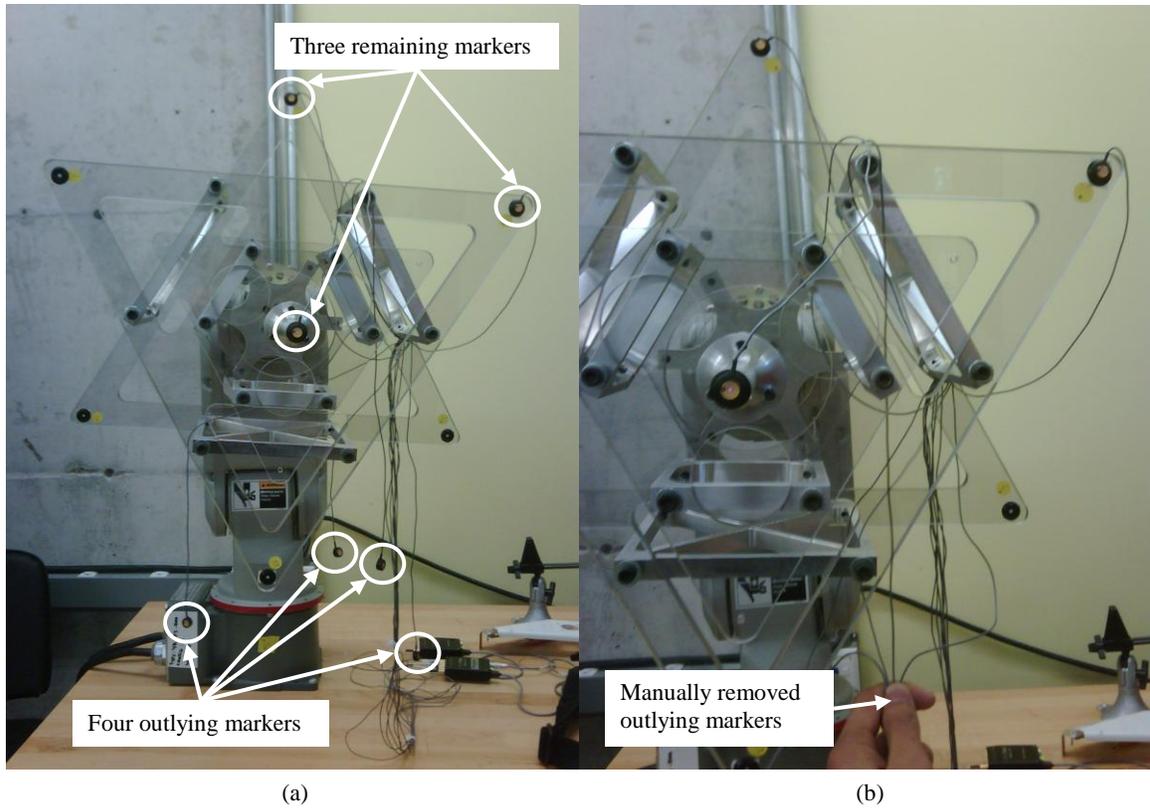
From the experiments, it is found that the pose estimation is highly sensitive to the outliers due to the nature of least-square methods that tend to accommodate the squares of all residuals. First, the outliers at least degrade the accuracy of pose estimation. Second, when the outliers are dominant and severe it leads to the failure of the pose estimation algorithm of the NDI system. As illustrated in Figure H.5, four outliers broke down the pose estimation even when three markers remained and the pose estimation was

restored immediately when these outlying markers were manually removed by holding them to ensure the invisibility to NDI cameras.

This has proved that the outliers have significant influence on the NDI pose estimation solutions, since these solutions are based on the least-square methods. Therefore, we should endeavor to avoid this outlier caused catastrophic failure event by providing a more robust pose estimation algorithm.



**Figure H.4:** (a) Original set-ups with 7 markers attached to the cubic target; (b) a way of creating an outlying marker by detaching the marker from the mounted case.



**Figure H.5:** (a) With four outlying markers, the pose estimation failed even when there are three remaining markers; (b) once four outliers were manually removed, the pose estimation of the rigid body was restored immediately.



## APPENDIX J - PUBLICATIONS AND PATENTS

### *Journal Papers*

1. **Lin, Y.**, Xi, F.F., Guo, S., Klim, G., and Chan, V. (2014). Robotic Riveting of Composite Materials. *Canadian Aeronautics and Space Journal*. CASJ-2014-0005. Paper under review.
2. **Lin, Y.**, Tu, X.W., Xi, F.F., & Chan, V. (2013). Robust Pose Estimation with an Outlier Diagnosis Based on a Relaxation of Rigid Body Constraints. *ASME Journal of Dynamic Systems, Measurement, and Control*, 135, 014502.
3. **Lin, Y.**, Xi, F.F., Mohamed, R.P., & Tu, X.W. (2010). Calibration of Modular Reconfigurable Robots Based on a Hybrid Search Method. *ASME Journal of Manufacturing Science and Engineering*, 132, 061002.
4. Xi, F.F., **Lin, Y.**, & Tu, X.W. (2013). Framework on robotic percussive riveting for aircraft assembly automation. *Advances in Manufacturing*, 1-11.
5. Zhao, Y.M., **Lin, Y.**, Xi, F.F., and Guo, S. (2014). Calibration-based Iterative Learning Control for Path Tracking of Industrial Robots, *IEEE Transactions on Industrial Electronics*, 14-TIE-0555, Paper under review.
6. Mohamed, R., Xi, F.F., & **Lin, Y.** (2014). A Combinatorial Search Method for the Quasi-Static Payload Capacity of Serial Modular Reconfigurable Robots. *Mechanism and Machine Theory*. Submitted.
7. Song, T., Xi, F.F, Guo, S., Ming, Z. & **Lin, Y.** (2014). A Comparison Study of Algorithms for Surface Normal Determination Based on Point Cloud Data, *Precision Engineering*. In press.

### *Conference Papers*

8. **Lin, Y.**, Xi, F.F., Klim G., Dakdouk D., Zeng C. & Chan, V. (2013, April). Robotic Riveting of Composite Materials. In Aerospace Manufacturing Technologies Symposium, CASI 60th Aeronautics Conference and AGM on, Toronto, Canada.
9. **Lin, Y.**, Tu, X.W., Xi, F.F., & Perron, C. (2010, July). A data decorrelation method for 3D position measurements from a line sensors based photogrammetric measuring system. In Advanced Intelligent Mechatronics (AIM), 2010 IEEE/ASME International Conference on (pp. 90-95), Montreal, Canada.
10. Zhao, Y.M., Xi, F.F., Guo, S., **Lin, Y.** (2014, June). Calibration-based Iterative Learning Control for Path Tracking of Robotic Riveting System. In The 11th World Congress on Intelligent Control and Automation (WCICA 2014), Shenyang, China.
11. Helal, M., Xi, F.F., & **Lin, Y.** (2013, Sept.). Robotic Tooling Self-Calibration. In 2013 SAE AeroTech Congress & Exhibition (13ATC-0225), Montreal, Canada.

### ***Patents***

12. Xi, F., **Lin, Y.**, Dakdouk, D., Helal, M., & East, B., 2013, Automated Percussive Riveting System, WO Patent WO/2013/152440.

### ***Technical Reports***

13. **Lin, Y.**, and Tu, X.W., Robust Pose Estimation with an outlier diagnosis based on a relaxation of rigid body constraints, Aerospace Manufacturing Technology Centre (AMTC), National Research Council Canada (NRC), LTR-AMTC-2010-0070, 7/23/2010.
14. **Lin, Y.**, Tu, X.W., and Perron, C. Data Analysis of NDI Readings, Aerospace Manufacturing Technology Centre (AMTC), National Research Council Canada (NRC), LTR-AMTC-2010-0019, 2/2/2010.

## REFERENCES

- [1] Jr, Flake C. Campbell, *Manufacturing technology for aerospace structural materials*. Elsevier, 2006.
- [2] B. Monsarrat, E. Lavoie, G. Cote, M. De Montigny, et al. "High Performance Robotized Assembly System for Challenger 300 Business Jet Nose Fuse Panels," *AeroTech 2007*, Los Angeles, CA, USA.
- [3] F. Xi, Y. Lin, and, X.W. Tu, "Framework on robotic percussive riveting for aircraft assembly automation." *Advances in Manufacturing*: 1-11, 2013.
- [4] F. Xi, *Materials and Manufacturing (AER507)*. Ryerson University.
- [5] ABB Robotics. *Product Specification IRB 4400*. 3HAC 8770-1. pp. 47.
- [6] Y. Q. Fei and X. F. Zhao, "An assembly process modeling and analysis for robotic multiple peg-in-hole," *Int. Journal of Intelligent and Robotic Systems*, vol. 36, pp. 175-189, 2003.
- [7] T. Arai, et al, "Hole research planning for peg-in-hole problem," *Manufacturing Systems*, vol. 26, no. 2, pp.119-124, 1997.
- [8] K. Sathirakul and R. H. Sturges, "Jamming conditions for multiple peg-in-hole assemblies," *Robotica*, vol. 16, pp. 329-345, 1998.
- [9] R. H. Sturges and S. Laowattana, "Virtual wedging in three-dimensional peg insertion tasks," *Journal of Mechanical Design*. vol. 118, pp. 99-105, 1996.
- [10] D. R. Strip, "Insertion using geometric analysis and hybrid force-position control: Method and Analysis," in Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 1744-1751, 1988.
- [11] D. E. Whitney, "Quasi-static assembly of compliantly supported rigid parts," *ASME Journal of Dynamic Systems Control*, vol. 104, pp. 65-77, 1982.
- [12] Janabi-Sharifi, L. F. Deng, and W. J. Wilson, "Comparison of basic visual servoing methods," *IEEE/ASME Trans. on Mechatronics*, vol. 16, no. 5, pp. 967-983, 2011.
- [13] L. E. Weiss, A. C. Sanderson, and C. P. Neuman, "Dynamic sensor based control of robots with visual feedback," *IEEE Journal of Robot and Automation*, vol. RA-3, no. 5, pp. 404-417, Oct. 1987.
- [14] W. J. Wilson, C. C. W. Hulls, and G. S. Bell, "Relative end-effector control using

- Cartesian position-based visual servoing,” *IEEE Trans on Robotics and Automation*, vol. 12, no. 5, pp. 684–696, Oct. 1996.
- [15] B. Espiau, F. Chaumette, and P. Rives, “A new approach to visual servoing in robotics,” *IEEE Trans. on Robotics and Automation*, vol. 8, no. 3, pp. 313–326, Jun. 1992.
- [16] F. Chaumette, “Potential problems of stability and convergence in image-based and position-based visual servoing,” in *The Confluence of Measurement and Control*, LNCIC Series, No 237, 1998.
- [17] O. Kermorgant, and F. Chaumette, “Combining IBVS and PBVS to ensure the visibility constraint,” in *IEEE/RSJ Int. Conf. on the Intelligent Robots and Systems*, pp. 2849-2854, 2011.
- [18] Y. Li, F. Xi, and K. Behdinan, “Modeling and Simulation of Percussive Riveting for Robotic Automation”, *ASME Journal of Computational and Nonlinear Dynamics*, Vol. 5, No. 2, 021011. 2010.
- [19] Cuypers, Wim, Nick Van Gestel, André Voet, J-P. Kruth, J. Mingneau, and P. Bleys. "Optical measurement techniques for mobile and large-scale dimensional metrology." *Optics and Lasers in Engineering* 47, no. 3 (2009): 292-300.
- [20] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, Wiley, 2001.
- [21] F. Janabi-Sharifi, and M. Marey, “A Kalman-Filter-Based Method for Pose Estimation in Visual Servoing,” *IEEE Trans. Robotics*, vol. 26, no. 5, pp. 939-947, 2010.
- [22] V. Lippiello, B. Siciliano, and L. Villani, “Adaptive extended Kalman filtering for visual motion estimation of 3D objects,” *Control Engineering Practice*, 15, pp. 123-134, 2007.
- [23] S. Jeon, M. Tomizuka, and T. Katou, 2009, “Kinematic Kalman Filter (KKF) for Robot End-Effector Sensing,” *ASME J. Dyn. Sys., Meas., Control*, vol. 131, p. 021010.
- [24] K. Halvorsen, T. Soderstrom, V. Stokes, and H. Lanshammar, “Using an Extended Kalman Filter for Rigid Body Pose Estimation,” *ASME J. Biomechanical Engineering*, vol. 127, pp. 475-483, 2005.

- [25] W. F. Xu, Y. Liu, B. Liang, Y.S. Xu, and W.Y. Qiang, "Autonomous Path Planning and Experiment Study of Free-floating Space Robot for Target Capturing," *J. Intell. Robot Syst.*, 51, pp. 303-331, 2008.
- [26] F. Aghili, and K. Parsa, "Motion and Parameter Estimation of Space Objects Using Laser-Vision Data," *J. of Guidance, Control, and Dynamics*, vol. 32, no. 2, pp. 537-549, 2009.
- [27] D.J. McTavish, R. Schumacher, and G. Okouneva, "Kalman filtering for dynamic pose and relative motion estimation in orbit," *Can. Aeronaut. Space J.*, vol. 53, no. 3/4, pp. 95-105, 2007.
- [28] F. Xi, D. Nancoo, and G. Knopf, "Total least-squares methods for active view registration of three-dimensional line laser scanning data," *ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 127, March 2005, pp. 50-56.
- [29] Y. Lin, X. Tu, F. Xi, and V. Chan, "Robust Pose Estimation with an Outlier Diagnosis Based on a Relaxation of Rigid Body Constraints", *ASME Journal of Dynamic Systems, Measurement and Control*, 2013.
- [30] B. K. P. Horn, "Close-form solution of absolute orientation using unit quaternions," *J. Optical Society of America A*, vol. 4, 1987, pp. 629-642.
- [31] J. A. Williams, M. Bennamoun, and S. Latham, "Multiple view 3D registration: a review and a new technique," In Proc. IEEE Int. Conf. Systems, Man and Cybernetics (SMC '99), vol. 3, 1999, pp. 497-502.
- [32] J. Angeles, "Automatic computation of the screw parameters of rigid-body motions. part 1: finitely-separated positions," *ASME J. Dynamic Systems, Measurements, and Control*, vol. 108, 1986, pp. 32-38.
- [33] P. H. Schonemann, "A generalized solution of the orthogonal Procrustes problem," *Psychometrika*, vol. 31, no. 1, 1966, pp. 1-10.
- [34] B. P. Wrobel, "Minimum solutions for orientation," in *Calibration and Orientation of Cameras in Computer Vision*, Gruen, and Huang Eds., Springer, 2001, ch. 2, pp. 33-35.
- [35] D. W. Eggert, A. Lorusso, and R. B. Fisher, "Estimating 3-D rigid body transformations: a comparison of four major algorithms," *Machine Vision and Applications*, vol. 9, 1997, pp. 272-290.

- [36] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 9(5), 1987, pp. 698-700.
- [37] B. K. P. Horn, H. M. Hilder, and S. Negahdaripour, "Close-form solution of absolute orientation using orthonormal matrices," *J. Optical Society of America A*, vol. 5, 1988, pp. 1127-1135.
- [38] O. D. Faugeras, and M. Hebert, "The representation, recognition, and locating of 3-D objects," *Int. J. Robotics Research*, vol. 5, no. 3, 1986, pp. 27-52.
- [39] M. W. Walker, and L. Shao, "Estimating 3-D location parameters using dual number quaternions," *CVGIP: Image Understanding*, vol. 54, no. 3, 1991, pp. 358-367.
- [40] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Trans. Pattern Analysis Machine intelligent*, vol. 13, 1991, pp. 376-380.
- [41] K. Kanatani, "Analysis of 3-D rotation fitting," *IEEE Trans. Pattern Analysis Machine intelligent*, vol. 16, no. 5, 1994, pp. 543-549.
- [42] D. Goryn, S. Hein, "On the estimation of rigid body rotation from noisy data," *IEEE Trans. Pattern Analysis Machine intelligent*, vol. 17, 1995, pp. 1219-1220.
- [43] J. A. Ramos, and E. I. Verriest, "Total least squares fitting of two point sets in m-D," in Proc. IEEE Conf. Decision & Control, 1997, pp. 5048-5053.
- [44] G. Wen, D. Zhu, S. Xia, and Z. Wang, "Total least squares fitting of point sets in m-D," in Proc. Computer Graphics International (CGI'05), 2005, pp. 82-86.
- [45] K. Kanatani, *Geometric Computation for Machine Vision*, Oxford Science Publications, 1993, pp. 112-114.
- [46] N. Ohta, and K. Kanatani, "Optimal estimation of three-dimensional rotation and reliability evaluation," in *Computer Vision – ECCV'98*, B. Neumann and H. Burkhardt Eds., Springer, 1998, pp. 175-187.
- [47] B. Matei, and P. Meer, "Optimal rigid motion estimation and performance evaluation with bootstrap," in Proc. CVPR'99, June 1999.
- [48] C. V. Stewart, "Robust parameters estimation in computer vision," *SIAM Review*, vol. 41, no. 3, 1999, pp. 513-537.
- [49] P. Meer, D. Mintz, and A. Rosenfeld, "Robust regression methods for computer

- vision: a review,” *Int. J. of Computer Vision*, vol. 6, no. 1, 1991, pp. 59-70.
- [50] P. J. Rousseeuw, and A. M. Leroy, *Robust Regression and Outlier Detection*, Wiley, New York, 1987.
- [51] V. J. Hodge, and J. Austin, “A survey of outlier detection methodologies,” *Artificial Intelligence Review*, vol. 22, 2004, pp. 85-126.
- [52] P. Filzmoser, “A multivariate outlier detection method,” in *Proc. of the Seventh Int. Conf. on Computer Data Analysis and Modeling*, vol. 1, 2004, pp. 18-22.
- [53] N. T. Trendafilov, “On the  $\ell_1$  Procrustes problem,” *Future Generation Computer Systems*, vol. 19, 2003, pp. 1177-1186.
- [54] O. Enqvist, and F. Kahl, “Robust optimal pose estimation,” D. Forsyth, P. Torr, and A. Zisserman (Eds.): *ECCV 2008, Part I, LNCS 5302*, Springer-Verlag Berlin Heidelberg, 2008, pp. 141-153.
- [55] A. F. Siegel, R. H. Benson, “A robust comparison of biological shapes,” *Biometrics*, vol. 38, 1982, pp. 341-350.
- [56] A. F. Siegel, J. R. Pinkerton, “Robust comparison of three-dimensional shapes with an application to protein molecule configurations,” Technical Report #217, series 2, Department of Statistics, Princeton University, 1982.
- [57] Z. Zhang, “Iterative point matching for registration of freeform curves and surfaces,” *Int. J. of Computer Vision*, 13(2), 1994, pp. 119-152.
- [58] X. Pennec, and J. P. Thirion, “A framework for uncertainty and validation of 3-D registration methods based on points and frames,” *Int. J. of Computer Vision*, 25(3), 1997, pp. 203-229.
- [59] X. H. Zhuang, and Y. Huang, “Robust 3D pose estimation,” *IEEE Trans. PAMI*, 16(8), 1994, pp. 818-824.
- [60] P. Boulanger, V. Moron, and T. Redarce, “High-speed and non-contact validation of rapid prototyping parts,” in *Proc. SPIE Rapid Product Development Technologies*, June 1996, pp. 46-90.
- [61] P. L. Rosin, “Robust pose estimation,” *IEEE Trans. Sys., Man, and Cybernetics-part B: Cybernetics*, vol. 29, no. 2, April 1999, pp. 297-303.
- [62] R. Kumar, and A. R. Hanson, “Robust methods for estimating pose and a sensitivity analysis,” *CVGIP: Image Understanding*, vol. 60, no. 3, November 1994, pp. 313-

342.

- [63] G. Taylor, L. Kleeman, “Visual perception and robotic manipulation”, Springer Tracts in Advanced Robotics, Volume 26, 2006.
- [64] S. Hutchinson, G. D. Hager, and P. I. Corke, “A tutorial on visual servo control”, *IEEE Transactions on Robotics and Automation*, 12(5): 651-670, 1996.
- [65] D. Kragic and H. I. Christensen, “Survey on visual servoing for manipulation”, Technical Report ISRN KTH/NA/P-02/01-SE, KTH, 2002.
- [66] A. C. Sanderson, L. E. Weiss, “Image-based visual servo control using relational graph error signals”, In proc. IEEE Conference on Cybernetics and Society, pages 1074-1077, 1980.
- [67] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, Society for Industrial and Applied Mathematics, 2000.
- [68] G. H. Golub, C. F. Van Loan, *Matrix Computations*, JHU Press, 1996.
- [69] Y. Lin, X.W. Tu, C. Perron, and F. F. Xi, "A data decorrelation method for 3D position measurements from a line sensors based photogrammetric measuring system," In *Advanced Intelligent Mechatronics (AIM)*, 2010 IEEE/ASME International Conference on (pp. 90-95), 2010.
- [70] A. Nubiola, M. Slamani, A. Joubair, and I. A. Bonev, “Comparison of two calibration methods for a small industrial robot based on an optical CMM and a laser tracker”, *Robotica*, pp 1 – 20, 2013.
- [71] F. Xi, “Computational dynamics (ME8138) lecture notes”, Ryerson Univ., Toronto, ON. Canada, c2007.
- [72] P. I. Corke, “Robotics, Vision & Control”, Springer 2011, ISBN 978-3-642-20143-1.
- [73] Y. Lin, F. F. Xi, R. P. Mohamed, and X. W. Tu, “Calibration of Modular Reconfigurable Robots Based on a Hybrid Search Method,” *ASME Journal of Manufacturing Science and Engineering*, 132, 061002, 2010.
- [74] G. S. Cox, and G. De Jager, “A Survey of Pattern Matching Techniques and a New Approach to Point Pattern Recognition,” Proc. IEEE South African Symp. on Comm. and Signal Processing, pp. 243-248, 1992.
- [75] S. Ranade, and A. Rosenfeld, “Point Pattern Matching by Relaxation,” *Pattern Recognition*, vol. 12, pp. 269-275, 1980.

- [76] P. J. Rousseeuw, and A. M. Leroy, *Robust Regression and Outlier Detection*, Wiley, New York, 1987.
- [77] J. C. Kent, N. Trigui, W. C. Choi, Y. G. Guezennec, and R. S. Brodkey, "Photogrammetric calibration for improved three-dimensional particle tracking velocimetry (3-D PTV)," in *Proc. SPIE Optical Diagnostics in Fluid and Thermal Flow*, vol. 2005, pp. 400–412.
- [78] C. Ogleby, "Photogrammetry," in *The Focal Encyclopedia of Photography*, 4th ed., 2007, pp. 583–584.
- [79] A. Gruen, "Fundamentals of videogrammetry – a review," *J. Human Movement Science*, vol. 16, 1997, pp. 155–187.
- [80] H-G, Maas, "Concepts of real-time photogrammetry," *J. Human Movement Science*, vol. 16, 1997, pp. 189–199.
- [81] H. A. Beyer, "Geometric and radiometric analysis of a CCD-camera based photogrammetric close-range system," Ph.D. dissertation, Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland, 1992.
- [82] A. Gruen, and T. S. Huang, *Calibration and orientation of Cameras in Computer Vision*, Springer, 2001, pp. 113–169.
- [83] T. Luhmann, S. Robson, S. Kyle, and I. Harley, *Close Range Photogrammetry*, Caithness: Whittles Publishing, 2006, pp.116–122.
- [84] C. S. Fraser, and S. Cronk, "A hybrid measurement approach for close-range photogrammetry," *ISPRS J. Photogrammetry and Remote Sensing*, vol. 64, 2009, pp. 328–333.
- [85] R.A. Johnson, and D.W. Wichern, *Applied Multivariate Statistical Analysis*, 5th ed., Prentice Hall, 2002, pp. 426–476.
- [86] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, Philadelphia, PA: SIAM, 2000, ch. 7.
- [87] Y. Lin, and X.W. Tu, "Robust Pose Estimation with an outlier diagnosis based on a relaxation of rigid body constraints," LTR-AMTC-2010-0070, 7/23/2010.
- [88] Y. Lin, X.W. Tu, and C. Perron, "Data Analysis of NDI Readings," LTR-AMTC-2010-0019, 2/2/2010.
- [89] J. Mortimer, "Jaguar uses X350 car to pioneer use of self-piercing rivets," *Industrial*

*Robot: An International Journal* 28, no. 3: 192-198, 2001

- [90] B. Morey, "Robot Seeks Its Role in Aerospace," *Manufacturing Engineering*, Vol. 139, No. 4, 2007.
- [91] J. Inman, B. Carbrey, R. Calawa, J. Hartmann, et al, "Flexible Development System for Automated Aircraft Assembly." Paper presented at SAE Aerospace Automated Fastening Conference & Exposition, Bellevue, WA, USA, October, 1996.
- [92] R. Kleebaur, "Where Precision Counts Above All Else," *The High Flyer*, No. 2, 2006.
- [93] Y. Lin, F.F. Xi, G. Klim, D. Dakdouk, C. Zeng, and V. Chan, "Robotic Riveting of Composite Materials," In Aerospace Manufacturing Technologies Symposium, CASI 60th Aeronautics Conference and AGM on, Toronto, Canada. 2013.
- [94] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, Third Edition, Pearson Education, 2005.
- [95] S. Bai, and M. Y. Teo, "Kinematic calibration and pose measurement of a medical parallel manipulator by optical position sensors," *Journal of Robotic Systems*, Wiley, 2003, pp. 202-209.
- [96] J. R. Magnus, and H. Neudecker. *Matrix Differential Calculus with Applications in Statistics and Econometrics* (revised ed.) Wiley. 1999.
- [97] Y.M. Zhao, Y. Lin, F.F. Xi, and S. Guo. "Calibration-based Iterative Learning Control for Path Tracking of Industrial Robots," *IEEE Transactions on Industrial Electronics*, 14-TIE-0555, 2014, Paper under review.
- [98] Y.M. Zhao, F.F. Xi, S. Guo, Y. Lin, "Calibration-based Iterative Learning Control for Path Tracking of Robotic Riveting System," In The 11th World Congress on Intelligent Control and Automation (WCICA 2014), Shenyang, China.