博士論文

Research and Development of a Tiny TOF Laser Line Sensor System for Task-Oriented Robotic Applications Based on Active Local Sensing

(能動局所センシングに基づくタスク指向ロボット応用のための小型 TOF レーザラインセンサシステムの研究開発)

Xiangyu Chen

陳　相羽

Research and Development of a Tiny TOF Laser Line Sensor System for Task-oriented Robotic Applications based on Active Local Sensing

Xiangyu Chen

A Dissertation
Presented to the Faculty
of The University of Tokyo
in Candidacy for the Degree
of Doctor of Philosophy

Recommended for Acceptance
by the Department of
Creative Informatics
Adviser: Masayuki Inaba

August 2017

Abstract

In a lot of real world robotic tasks, the perceptual incompleteness limits the effectiveness of robot agent systems when encountering a complex world environment with uncertainty. The appearance of powerful sensors and scene understanding approaches greatly enhanced the robot perception from several levels in recent years. However, the robot perception is still awkward because the perception hardware and software are not flexible and customizable when handling different tasks.

This dissertation addresses this very prevalent issue on robotics research and proposed our novel systems and approaches. There are basically two methodologies to handle the incomplete perception problem, enhance the sensing ability and equip reasoning ability. To improve the sensing flexibility, we design a tiny TOF laser line sensor which can be flexibly installed on any part of the robot for multi-purpose applications. We also discover that by providing some prior of the real world knowledge like functionality, geometry and physics, we can design a reasoning vision cognition framework to make a guess of environment even without direct observation and then verify the recognition results using our tiny sensor. The lightweight flexible, customizable and robust features of the tiny laser line sensor are demonstrated in several active local sensing based task-oriented robotic applications.

A novel lightweight tiny laser line range sensor system based on the Time-of-Flight (TOF) principle is developed. Attribute to the delicate circuit design and optical attachments, the sensor is as small as $35[mm] \times 27[mm] \times 30[mm]$ and less than $20[g]$ while achieve 256 line effective pixels under a single measurement with a range of $0.05[m] \sim 2[m]$. A higher measurement rate($60 \sim 100[Hz]$) can be achieved in short range application. We model the overall errors of the sensor and formulate calibration methods, achieving repeatable accuracy and measurement bias both within $2[cm]$ with our tested ambient lighting conditions and measurement ranges.

The active local verification with reasoning based vision approach is demonstrated through the application of agricultural tomato harvesting task. The difficult lies in detecting the pedicel of each tomato which is very small and cluttered. On the vision side, we consider a simple fact that with respect to the gravity and interaction forces, every tomato remains stable due to the physics rules. According to this assumption, a probabilistic model is created and the picking order in the branch is assigned under the evaluated geometrical structure. Given the guesses, we apply the tiny laser sensor to verify and detect where the pedicel is through local sensing.

The task-oriented robotic applications of metallic tools grasping and multi-link aerial robot manipulation using tiny laser line sensor are documented. We developed

a reflectance guide local sensing framework using the customized tiny laser sensor output to align the robot end-effector to the tools with metallic surface of high reflectance to achieve grasping. For aerial robot application, the multi-link aerial robot whole-body object manipulation application using multiple tiny laser sensor system is illustrated. The tiny size, lightweight and as well as multiple sensors operation framework demonstrated the feasibility and effectiveness of this difficult robotic application.

Acknowledgements

First and foremost I want to express my sincere thanks to my advisor Prof. Masayuki Inaba. For the last three years of my Ph.D. student period, he provides me a lot of priceless precious advices whenever I have any hesitations for my research and problems of daily life as a foreign student. I feel so honourable to be your Ph.D. student and it is you that shows me the endless enthusiasm to robotics research and the way to be a good researcher in this field.

Prof. Kei Okada has very kindly offered a lot of precious suggests and ideas to my research and taught me how to create a opensource software and manage it. I would like to thank him.

Dr. Hiroaki Yaguchi, Dr.Yohei kakiuchi, Dr. Shunichi Nozawa and Dr. Kotaro Nagahama also provided priceless helps to me for these 3 years and I want to give them my sincere thanks.

Dr. Fumihito Sugai helped me a lot in hardware research and when I was building the laser sensor, he offered me his sophisticate experiences in the circuits design. Without him, I can not have built the very complex hardware system myself. I would like to offer my thanks to him.

Dr. Wesley P. Chan has been a responsible tutor to me and he taught me a lot in JSK systems, also the English writing skills when I am trying to write a research paper. Krishneel Chaudhary often discuss about the research with me and we go the gym for training together in the last 3 years. I want to give my since thanks.

Thanks also go to the other MBZIRC team members Moju zhao, Kohei kimura, Hiroto Mizohana, Fan shi and Tomoki Anzai. We work together and fight together, no matter the result we do not regret.

I want to also thank all the nice and kind JSK members and I want to say it is great to have you guys as my lab mates and I feel so grateful to know you guys here in JSK.

To my parents.

Contents

Abstract iii
Acknowledgements v
List of Tables xi
List of Figures xii

1 Introduction **1**
1.1 Robot Perception 3
1.2 Robot Active Range Sensing 4
1.3 Task-oriented 3D Visual Reasoning and Local Sensing 5
1.4 Research Goals and Contributions 7
1.5 Dissertation Outline 8

2 Motivation and Related Works **11**
2.1 Incomplete Perception 11
2.2 Active Range Sensing 14
2.2.1 Lidar Sensor 14
2.2.2 3D Camera 16
2.3 Reasoning and Local Sensing based Vision Cognition 18
2.3.1 3D Visual Reasoning 18
2.3.2 Local Sensing 19

3 Development of Lightweight Tiny TOF Laser Line Sensor System **21**
3.1 Introduction 21
3.2 Sensor Hardware Design 22
3.2.1 Architecture Overview 22
3.2.2 Logical Timing 26
3.2.3 Laser Diode Charge and Drive 27
3.2.4 Data Acquisition and Correction 28
3.3 Calibration Analysis 29

3.3.1 Hardware Pulse Delay Analysis . . . 30
3.3.2 Errors from Non-constant Reflected Light . . . 31
3.3.3 Comprehensive Calibration . . . 33
3.3.4 Propagation Delay Caused by System Temperature . . . 34
3.4 Software Configuration for Calibration . . . 35
3.4.1 Sensor Firmware Configuration . . . 35
3.4.2 Laser Sensor Calibration Tool . . . 36
3.5 Experimental Results . . . 38
3.6 Summary . . . 43

4 Active Local Verification with Reasoning-Based Vision Recognition in Robot Tomato Harvesting 48
4.1 Robot In Agriculture . . . 48
4.1.1 Agriculture Harvesting Automation State of Art . . . 50
4.1.2 Vision Meets Cognition . . . 51
4.2 Harvesting Humanoid System . . . 51
4.2.1 Robot Platform . . . 51
4.2.2 Harvesting Gripper Evaluation . . . 53
4.2.3 Installation of Hand-Camera . . . 54
4.3 Reasoning based Vision Cognition . . . 57
4.3.1 Tomato Branch Clustering . . . 57
4.3.2 Ransac-based Primitive Model Segmentation . . . 58
4.3.3 Physical and Geometric Reasoning . . . 60
4.4 Local Sensing Based Picking Verification using Tiny Laser Sensor . . . 63
4.4.1 Tiny Laser Sensor Continuous Measurement Mode . . . 63
4.4.2 Laser Data Real-time Clustering using DBSCAN . . . 64
4.5 Experimental Results . . . 67
4.6 Summary . . . 73

5 Development of Metallic Tools Grasping Robot System Based on Reflectance Feedback Using Tiny Laser Sensor 74
5.1 Participation in MBZIRC and Metallic Tools Grasping Task . . . 74
5.2 A Task-oriented High Power Field Robot Platform For Robotic Challenge 75
5.2.1 Hardware and Software Architectures . . . 77
5.2.2 Mobile Wheeled Base Kinematics and Optimized Feedback Control . . . 79
5.2.3 Experimental Results of HRP2G in Test Field . . . 82

5.2.4 Discussion . . . 86
5.3 Robot Wrench Manipulation Based on Customized Tiny Laser Sensor using Object Reflectance Information . . . 88
5.3.1 Raw Intensity Data Continuous Measurement Mode . . . 88
5.3.2 Experiments on Local Sensing based Wrench Picking . . . 90
5.3.3 Discussion . . . 92
5.4 Summary . . . 92

6 Vision Recognition and Local Sensing Based Object Manipulation for Aerial Robot Systems 94
6.1 Target Detection and Picking with Vision-Based Drone Systems . . . 94
6.1.1 Platforms . . . 95
6.1.2 System Framework Design . . . 97
6.1.3 Experiments . . . 101
6.1.4 Discussion . . . 104
6.2 Multi-link Aerial Robot Whole-body Object Manipulation Using Lightweight Tiny Laser Line Sensor . . . 104
6.2.1 Multi-Sensors Framework with Sequence Measurement Mode . 107
6.2.2 Sampling based Object Detection using Whole-body Laser Sensing Data . . . 107
6.2.3 Experiments . . . 110
6.2.4 Discussion . . . 110
6.3 Summary . . . 111

7 Conclusion and Perspectives 114
7.1 A Low-cost Lightweight Tiny Laser Line Sensor for Robotic Applications 115
7.2 Tomato Harvesting Robot System with Reasoning and Active Local Verification based Vision Approach . . . 115
7.3 Task-oriented Robotic System Applications Based on Active Local Sensing Using Lightweight Flexible Tiny Laser Line Sensor System . . 116
7.4 Contributions . . . 116
7.5 Future Work . . . 116

A Laser Sensor Design Details 118
A.1 Hardware Schematic . . . 118
A.2 Calibration Objects . . . 118

B Publications and Awards **123**
B.1 First Author Publications 123
B.2 Other Publications 123
B.3 Awards 124

Bibliography **125**

List of Tables

3.1 Sensor Measurement Noise(Published in Chen et al. 2016 [1]) 40
3.2 Statistic Measurement Result of Different Materials 43
3.3 Comparison of Popular Range Sensors 45

4.1 Specification of HRP2W(Published in Chen et al. 2015 [2]) 53
4.2 Harvesting experiment((Published in Chen et al. 2015 [2])) 69
4.3 Laser Guide Picking Experiment 71

5.1 Specification of HRP2G(Published in Chen et al. 2016 [3]) 77

6.1 Specification of customized DJI M100 in MBZIRC Challenge 3 96

List of Figures

1.1 The tiny laser line sensor system(Published in Chen et al. 2016 [1]) . 6
1.2 Roadmap of dissertation outline 10

2.1 Phase Shift ToF Measurement 15

3.1 Laser sensor hardware overview 23
3.2 Laser sensor hardware architecture(Modified from Chen et al. 2016 [1]) 24
3.3 Illustration of sensor operation flow(Modified from Chen et al. 2016 [1]) 25
3.4 Distance sensor pixel structure(Published in Chen et al. 2016 [1]) . . 27
3.5 Laser Drive Circuits 28
3.6 Basic logic timing(Published in Chen et al. 2016 [1]) 30
3.7 Adjust the logic timing using high resolution timer 32
3.8 Tiny Laser Sensor Calibration Software Tool 36
3.9 Laser calibration demonstration of multi-discharge data(Modified from Chen et al. 2016 [1]) 37
3.10 Laser calibration using 6d polynomial regression(Modified from Chen et al. 2016 [1]) 38
3.11 Laser Calibration Temperature vs Discharge Ratio 39
3.12 Laser calibration dynamic measurement noise analysis(Modified from Chen et al. 2016 [1]) 40
3.13 Laser Scan Demo(Published in Chen et al. 2016 [1]) 41
3.14 Measurement Result of Different Materials 42
3.15 Popular Range Sensors in Robotics 44
3.16 Tilt Laser Scan using Dynamixel Servo 46
3.17 Sensor applications(Modified from Chen et al. 2016 [1]) 47

4.1 Participation in tomato harvesting challenge(Published in Chen et al. 2015 [2]) 49
4.2 HRP2W Robot(Published in Chen et al. 2015 [2]) 52

4.3 Illustration of gripper model(Published in Chen et al. 2015 [2]) 54
4.4 Vision algorithm flow diagram(Modified from Chen et al. 2015 [2]) . . 56
4.5 Illustration of pedicel direction distribution(Published in Chen et al. 2015 [2]) 62
4.6 Pedicle Detection using Laser Sensor in Tomato Harvesting 65
4.7 Harvesting motion(Modified from Chen et al. 2015 [2]) 68
4.8 Tomato branches of different geometric complexity(Published in Chen et al. 2015 [2]) 69
4.9 Detection and harvesting results(Modified from Chen et al. 2015 [2]) 70
4.10 Local Sensing based Motion Alignment 72

5.1 HRP2G Robot Platform for MBZIRC (Published in Chen et al. 2016 [3]) 76
5.2 Software Framework(Published in Chen et al. 2016 [3]) 76
5.3 HRP2G Gripper Design(Published in Chen et al. 2016 [3]) 78
5.4 Idea ICR Model(Published in Chen et al. 2016 [3]) 80
5.5 HRP2G Picking the Wrench(Published in Chen et al. 2016 [3]) 83
5.6 Mobile Base Control Experiment(Published in Chen et al. 2016 [3]) . 85
5.7 Experiment on MBZIRC task(Published in Chen et al. 2016 [3]) . . . 87
5.8 Reflectance Based Clustering Demonstration 89
5.9 Wrench Manipulation by Customized Magnet Gripper using Local Sensing Laser 91
5.10 Wrench Manipulation by General Purpose Robot Gripper using Local Sensing Laser 93

6.1 Task 3 Platform: DJI M100 95
6.2 Task 3 Platform: Hydrus 96
6.3 Challenge 3 Gripper 97
6.4 Challenge 3 State Machine 100
6.5 Challenge 3 GAZEBO Simulation 102
6.6 Pick, Place and Search Demonstration 103
6.7 Local Sensing based Multi-link Aerial Robot Manipulation 105
6.8 Laser Data Output of Different Shape Objects Manipulation 106
6.9 Demonstration of Object Local Sensing using Multiple Laser Sensor for Multi-link Aerial Robot Manipulation 109
6.10 Manipulation Demonstration 111
6.11 Multiple Laser Local Sensing based Object Manipulation for Multi-link Aerial Robot 112

A.1 Hardware Bottom PCB Schematic 1 119
A.2 Hardware Bottom PCB Schematic 2 120
A.3 Hardware Top PCB Schematic 121
A.4 Laser Sensor Calibration Objects 122

Chapter 1

Introduction

Attribute to the evolutions of information technology science over the past half century, tremendous impressive milestones have been achieved in robotics field. Nowadays robots have been used in different areas among traditional industrial assembly[4], high labour-dense agriculture[5][6][7], high accurate medical surgery[8][9][10], autonomous warehouse management[11][12], autonomous parcel delivery[13][14][15] and senior care service applications[16][17][18], which accelerate the productivity of the society.

In addition, for recent decades, researchers are focusing on the intelligent robot systems. Compared to the traditional machine-like robots that just follow a certain movement routine, the intelligent robot systems can handle tasks that contain more environment uncertainty, request for more sensing and control accuracy or require more system autonomy ascribe to the developments of robot perception, control and machine learning technologies. The intelligent robotics system have several advantages with respect to the real world environment:

Tolerance to the environment uncertainty: The real world is very complicate and the environment can be very clattered for the robots. Thus being capable to obtain a good perception of the surroundings by various sensors, generate an approximate model of the environment through recognition and reasoning and achieve an accurate execution by robust control are very important in real world robotics tasks. Human beings are born with such ability to adapt to the environment uncertainty since we have a very delicate sensing system and very complex brain cerebra cortex to process, decide and control, which is under millions of years long evolution. With these basic intelligent "abilities" that human beings conclude and put into robots, the robots can handle common real world tasks more accurately.

Sensing and control accuracy: The development of perception and control technology make it possible for nowadays robots to attain a very accurate measurement of environment through powerful sensing system and achieve a precise control results by the advanced executors. As it is wide known that compared to human beings, robots excel in handling repetitive and high-precision tasks due to the different control framework. However, the perception sensors for robots were "big", "fat" and poor in resolution and accuracy before. Recent integrate sensors enable the robots to have a higher perception rate and accurate perception data than human beings and also sense some information that we can not percept. For example, 3D sensors can give a "mm" level accuracy in 3D distance while human beings can only give a approximate guess of the distance, and these sensors are becoming smaller and lighter in size and weight.

System autonomy: Autonomy is becoming a considerable standard to evaluate a robotics system. Almost all robot tasks requires a certain percentage of robotics system autonomy. Especially for the unmanned and hazard tasks like disaster rescue, nuclear leakage clean up, outer space operations, which within these tasks, human beings can only teleoperate the robot while all the instant information and decision must be processed immediately and properly by the robot itself. As well as in service robotics area, system autonomy means to release human labour and reduce the labour cost. Until now, although cobots[19] seems to be an acceptable and promising solutions for service and industrial robots right now, increasing the autonomy percentage of the intelligent robot systems and finally achieve fully autonomous still remains as a life-long goals for robotics researchers.

More recently, there are some robotics challenges held by both government and non-government organizations all over the world. Including the DARPA(Defense Advanced Research Projects Agency) Grand Challenge and Urban Challenge for Autonomous vehicles where teams from Stanford University[20] and CMU[21] won the first place respectively, the DARPA Robotics Challenge for humanoid robots where the SHAFT company robot[22][23] won the trial and the KAIST Hubo[24] won the final and the most recent MBZIRC(Mohamed Bin Zayed International Robotics Challenge) for drones. Through these challenges, the whole society can learn the cutting edge technologies in robotics field. Some of the technologies received investment from the industry and turned into real business like autonomous driving cars which is a hot topic for both research and industry. There are also some research projects related to the intelligent robot systems conducting collaborations with industrial companies. News and market evaluation reports also point out that there will be significant

growth in intelligent robot market as the news and reports themselves are growing exponentially[25]. Thus we can infer that in the near future the intelligent robots will become more and more common and popular in people's daily life.

1.1 Robot Perception

The perception plays a very important role in intelligent robot systems. It is how robot obtain the environment information, namely the bridge between the outside environment and the robot. For human beings, we have five senses: vision, audition, olfaction, somatosensation and gustation, where 83% of information come from the sight, 11% from the hearing, 3.5% comes from the smell, only 1.5% and 1.0% from the touch and taste respectively[26]. In robot sensing system, the researchers mainly use camera and microphones to collect vision and audition information. Through decades of sensing technology developments, recent sensors can achieve higher measurement frequency and resolution than our sensory organs, but these sensors still lack the flexibility since our biological structures are very exquisite. Meanwhile, how to correctly and intelligently understand the collected data is still a unsolved problem in robotics.

Some researchers tried to solve the first problem by building the structure of human beings eyeball system using artificial muscles[27][28] while for more cases multi-sensors system are applied to compensate for the flexibility like hand-eye[29], active local sensing[30] and range sensing systems[31]. The development of range sensors make it possible for robots to acquire a more precise 3D data than human beings, which is a big advantage of robots compare to human beings because human's vision system is basically rough stereo vision and passive sensing which means human can not tell the accurate range data. Considering all these advantages, we can predict future importance and necessity of range sensing technology and the use of range data in robotics applications.

For the second problem there are several methodologies[32] in robotics research. The traditional pattern recognition(PR) method uses the pre-knowledge of the objective scene and utilize the filtering, clustering and signal processing based method to perform the recognition. In addition, the range sensors provide one more dimension information to the data and brings new approaches like manhattan restriction[33], 3D features[34], and real world physical reasoning[35] and geometrical reasoning[36]. Another prevalent methodology is machine learning(ML)[37], which is either connected or independent compared PR. In recent 5 years, attribute to the powerful computational ability of GPU and the huge amount of big data, the appearance and realization

of deep neural network(Deep Learning)[38] significantly changes the whole information technology field, including computer vision[39], natural language process[40] and robotics control[41] and yielded less dependency on the traditional PR method.

The traditional PR method gives the reason why it work and why it does not work so that researchers can always make improvements and evaluate the stability of the method under certain environment condition. However, the PR method will become super complex and hard to maintain if we want to make it widely applicable. On the other hand, the machine learning method could provide very impressive results and seems very robust as long as the model is good and the training data is "big" or "appropriate". However, researchers still can not give a "strict" proof to the system stability or results convergence of the whole framework, which is why so far for a lot of robotics applications, the traditional PR methods are widely used. Hopefully in the coming future, with the "evolution" of the deep learning, researchers may come up with a more common, powerful, robust and mathematical reasonable general method, namely, the complete AI(Artificial Intelligence) for robotics.

1.2 Robot Active Range Sensing

Active range sensors emit their own energy source(mostly different kinds of waves). The energy source hits objects echo from the objects is detected and measured by the sensors. Most animals including human beings use passive range sensing, namely stereo vision system to obtain the rough 3D distance data of the environment while some animals possess active biosonar system like dolphins and bats so that they can be the accurate predators. Range sensing also contributes to the precision of robotics applications.

The mainstream of active range sensors are divided into two big types, the sound wave based and the light wave based. The first type is called sonar sensor for a lot of time and the basic principle is measuring the time between the sending pulse and the echo as the travelling speed for the sound is constant in the same medium and fast enough in ordinary measurement[42][43]. Sonar sensors are cheap, easy to use and applicable in underwater tasks while the constrains are also obvious: short measurement range, low measurement frequency and sensitive to noise. The other very popular type is using light waves and there are several kinds of sensing principles: direct triangular measurement, structure light measurement and the time of flight(ToF) measurement[44].

Laser triangulation principle is widely used in surveying and it is simple, intuitive and affordable[45][46]. However, there will be blind area with respect to the length of the baseline and the size increases if the system demand longer measurement range. Structure light technology is also based on triangulation principle, but much more complex. In structure light range sensors, the laser emitter projects certain light patterns, due to the difference shape of the objects and different range and as a consequence, the reflected pattern in receiver camera will get deformed in structure and scale, by analysing these distortions compared to the original projected patterns, depth information could be estimated[47][48]. The structured light sensors are precise and high resolution while the challenges remain in high reflectance and thin objects. The time of flight(ToF) principle is the most intuitive way. Like sonar sensors, since the velocity of sound and light are almost constant, measuring the time delay between the emit and receive the distance data can be obtained. ToF is a very flexible range sensing method and the measurement ranges from centimetres to thousands of kilometres[49][50][51]. For robotics, the Lidar(Light Detection And Ranging) sensors are used decades before to do obstacle avoidance, SLAM. The appearance of 3D Lidar sensors make it possible for researchers to do 3D objects recognition, 3D reconstruction and 3D SLAM. Recently, the commercial affordable ToF camera which is mainly build for gaming, contributes to the robotics research since the accuracy, frequency and measurement range get improved ascribe to the ToF good features. Thus we can predict that similar kind of ToF sensors are potentially demanded in some unconventional applications in robotics research.

1.3 Task-oriented 3D Visual Reasoning and Local Sensing

Visual reasoning is a big and ultimate goal of visual intelligence. Researches are working on the common approach that by providing captured data, based on the pre-knowledge and extracted features, deduces the object affordances[52][53], scene physical structure[54], movement causality[55], etc. Although recent 3D range sensing technology brings new perspective in reasoning research, so far there are no approaches that can handle all the common tasks since it is almost impossible to define all the pre-knowledges. As a matter of fact, these methods are applied in certain scene and indeed helps improving the results that conducted by traditional pattern

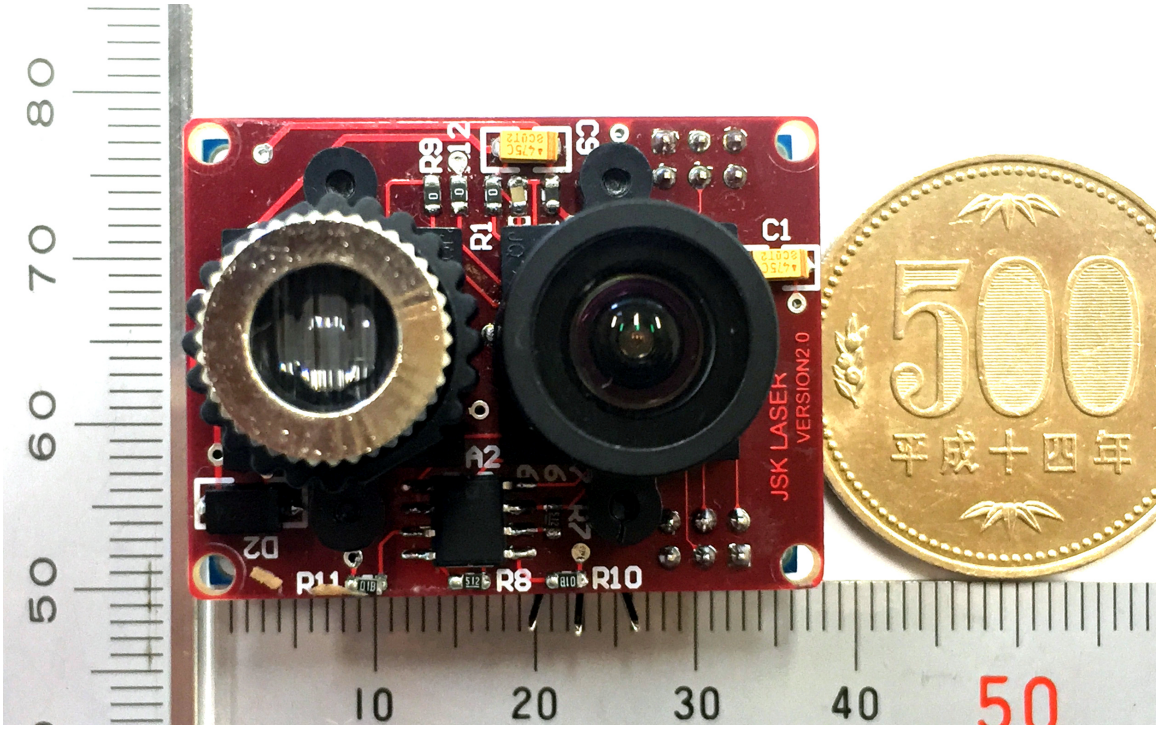


Figure 1.1: The tiny laser line sensor system(ruler unit: mm)

recognition method, while as high level vision approach in task-oriented robotics applications.

Local sensing is a general method to compensate for the sensing incompleteness and the high level vision recognition errors due to the environment uncertainty and complexity. The main features of local sensing are concluded as three aspects[56]. Firstly it is targeting at providing a set of relative information with respect to the task, mostly low level information like target distance, reflectance and orientation. Secondly the sensors for local sensing are mostly on-board sensors and as a consequence the size of the sensors plays an important role in selection and evaluation. Lastly, the corresponding related actions and motions are generally a "reflex" since the data is also low level.

It is very necessary to apply reasoning based methods combined with the local sensing strategy in intelligent robot tasks since it always happens that the sensors can not obtain all the information for the robot to make decision and finish the motion. The predefined knowledge will definitely help the robot to get a more robust recognition results[57][58], and the local sensing will then return confirmation from the low level data to the upper recognition to generate the final decision.

1.4 Research Goals and Contributions

In this dissertation we are mainly conducting the system research and development of intelligent robot perception which is considered as traditional but also of great significance nowadays. We consider that at most of the time it is very difficult for intelligent robot systems to obtain the adequate information directly from the sensors to make decision in certain tasks. Addressing on this very common problem, we are trying to both improve the sensing ability and vision intelligence and find a way to combine them into the robot system. Targeting at achieving some of the very difficult robot tasks that demand both sensing ability and vision intelligence, the research not only proposes novel software approaches, but also develops the customized hardware system.

We demonstrate our algorithm and hardware system in task-oriented robotics applications and show the feasibility and robustness of methodology. The contributions of this dissertation includes the following:

Firstly a tiny laser line sensor based on ToF principle is developed. We designed the hardware of both circuits and optical attachments and struggled to make it as small as possible. To reduce the cost and size, the sensor logic is controlled by a 32-bits MCU(Micro Control Unit) that is the one of the newest and most advanced commercial product. In order to handle the noise from the hardware side, we install narrow band optical filter and power management circuits. Besides the hardware part, we also develop a calibrate method for the sensor data at both hardware and software side. The temporary version of sensor where we published in [1] is of an ultra-tiny size of $35[mm] \times 27[mm] \times 30[mm]$ and weight of $20[g]$. The sensor can provide both reflectance information and distance information and based on our caliberation algorithm, the sensor can achieve standard bias of less than 0.5% and a floating measurement error of around 1.5% with respect to the measurement range. The sensor can be adopted in a lot of robotics applications and its small size and lightweight make it very suitable for local sensing.

Secondly, an active local verification with reasoning based vision recognition framework and the harvesting humanoid robot system are proposed toward a very difficult robot task, tomato harvesting[2]. Since tomatoes always gather together in one branch and the stems that hold each tomato could be hardly detected, we developed a algorithm that adopts the physical and geometrical reasoning methods which simply bypassed the detection of the stems and deduce where the stems should be from the input 3D information. Thus making it possible for a humanoid robot

to harvest the tomatoes using scissors, namely selective harvesting exactly like what human beings do. Our lightweight laser line sensor demonstrates its advantages in detecting the stem as an active local verification process to compensate for the vision recognition error and handle the environment uncertainty based on the optimized features of our laser sensor. The recognition results, harvesting results and local sensing results are showed and analysed, the vision algorithm feasibility and the sensor system framework effectiveness are justified.

Thirdly, based on the flexibility of our laser sensor system, we demonstrate how our sensor system can help in some difficult robotic applications. Such as in the robotic challenge MBZIRC, the metallic tools manipulation task, combined with the laser sensor intensity data, the robot systems[3] we developed can show the interesting demonstrations of metallic tools grasping based on reflectance feedback. Another example is the aerial robot object manipulation tasks. The lightweight, multi-sensor applicable features of our tiny laser line sensor can contribute to the local sensing of object manipulation, especially for the multi-link aerial robot which transforms and holds to object. The experiments of our robot challenge task-oriented systems are demonstrated and the analysis of the feasibility and the potential improvements are discussed by the present results.

1.5 Dissertation Outline

The roadmap of this dissertation is illustrated in Figure 1.2. In the following of the dissertation, the next part Chapter 2 describes the motivation of this research and the state of art literatures in the related fields. Chapter 3 describes the development of the lightweight tiny laser line sensor, which can be considered as the main contribution in this dissertation. In this chapter, the details of how to develop the sensor in both hardware and software are described. The advantages of our delicate hardware design is emphasised inside this chapter and for the software part, we analysis the systematic errors for the sensor and proposed our hardware and software calibration methods which previous work has not addressed. Chapter 4 documents the tomato harvesting robot system with reasoning based vision recognition and active local sensing verification approaches. It is also demonstrated in this chapter that the reasoning based vision recognition method can give a new perspective in handling the issue of inadequate observation. In Chapter 5 we focus on the hardware and software development of task-oriented robot systems for metallic tools grasping based on the reflectance feedback of the metal surface. We demonstrate the effectiveness of

how a flexible local sensing tiny sensor can help handling the difficulties in this task. Chapter 6 documents the vision recognition and flexible local sensing applications for aerial robot manipulation tasks, especially for the multi-link aerial robot, which illustrates the considerable advantages of our sensor in this kind of applications. The last part Chapter 7 concludes the whole dissertation.

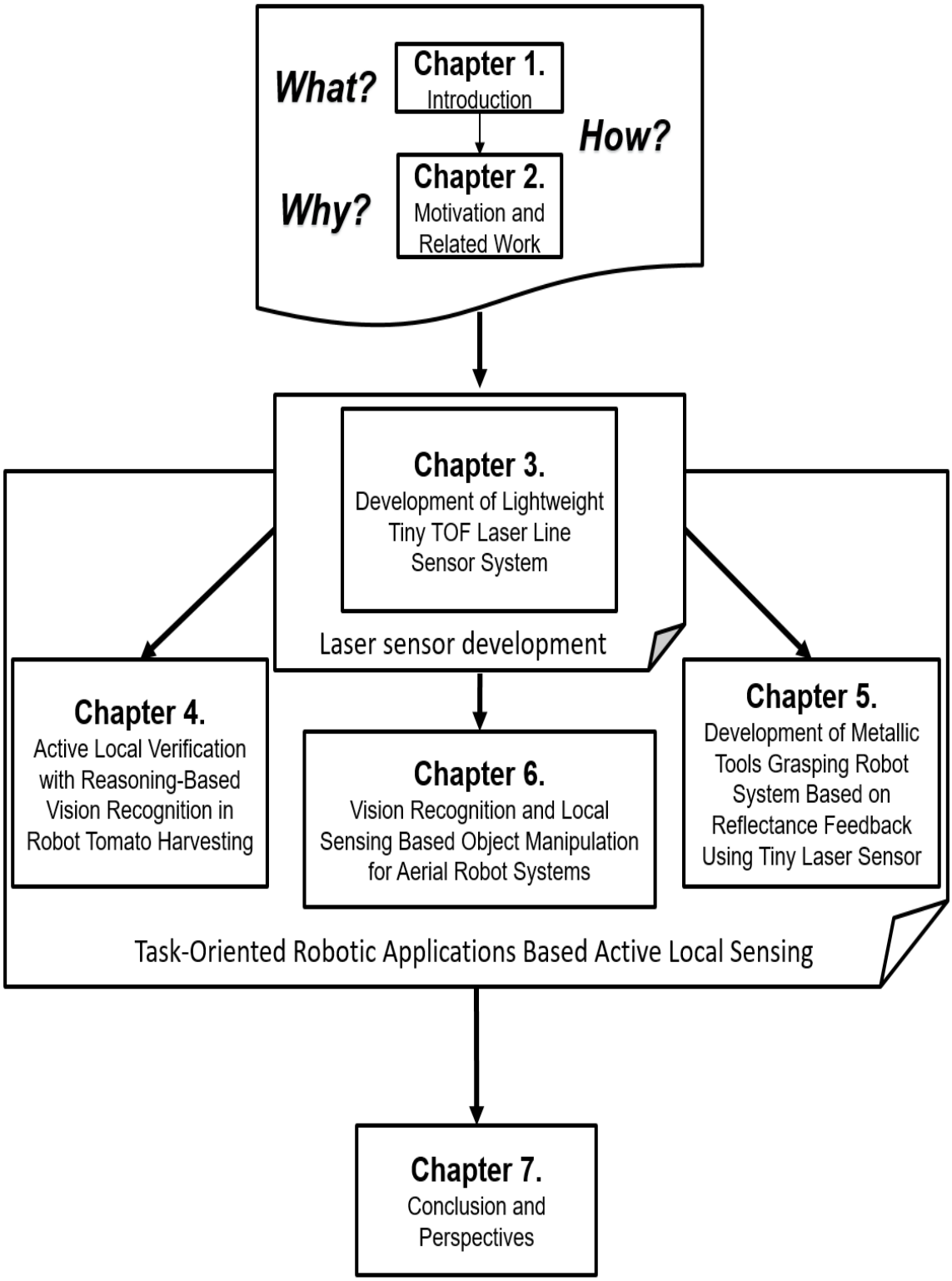


Figure 1.2: Roadmap of dissertation outline

Chapter 2

Motivation and Related Works

In this chapter we first states the motivation of this research by discussing some of the common exiting problems in robotics systems which have been concluded from other literatures. We also model the mathematical expression of the problems and claim several solution methodologies. Toward these methodologies, a general review of the literatures and a discussion of the existing studies are then conducted in the later sections of this chapter.

2.1 Incomplete Perception

"The world is its best model and the trick is to sense it appropriately and often." [59][60] Disappointed with the performance of the first general-purpose mobile robot Shakey [61] Rodney A. Brooks speaks out the very fundamental problem of robot systems in his very famous article that criticizes the tradition symbolic representation based AI approach. The sentences from the philosophers are always short but significant, and lead to conjectures [62] that:

1. **Conjecture 1:** "If perception is complete, then an agent whose actions are a function of percepts only (i.e. a purely reactive agent) can potentially be effective in all environments."

2. **Conjecture 2:** "An agent with incomplete perception must maintain internal state information to remain effective"

However, the perception of an agent is only complete when all the relevant features can be observed at any situation and time. Features are relevant when the evaluation of the next action in a situation is depending upon those features. As a matter of

fact, it is impossible to observe all the relevant features of the world and even the definition of the complete relevant features is obscure. [62] illustrated some of the reasons for incompleteness of perception since it is ubiquitous for robot systems:

1. **Sensor limitations:** Limited FOV(field of view), distance and resolution limitations, sensor error output.
2. **Physical Obstructions:** A very common problem.
3. **Monetary cost of sensors:** Price, size, weight, etc.
4. **Computational costs:** The more information, the more computational costs.
5. **Mutually exclusive sensing:** Destructive sensing.

As the issues above can not be reasonably overcome, thus a conclusion was drawn that the perceptual incompleteness is a pervasive property of most autonomous robot systems in most domains. However, perceptual incompleteness as the robot systems are, by obtaining more relevant features to determine the next choice of movement, the systems can achieve relatively effective results under the internal state(in Conjecture 2) since the agents do attain a more appropriate senses.

Consider this, there are several methodologies that directly and indirectly make an attempt to improve the effectiveness of perception incomplete agent systems in their domains. To explicitly explain the philosophy, we paraphrase these obscure sentences to intuitive mathematical expressions, note it as an optimization problem and the objective function is the evaluation function $\mathcal{F}$ of the robot effectiveness that:

$$\operatorname{argmax} \mathcal{F} \tag{2.1}$$

denotes that for a continuous perceptions $\mathcal{P}_i \in \mathbb{P}$ in the environment which imply potential world features:

$$\mathcal{P}_i \models \Lambda_i \tag{2.2}$$

where Λ_i is the features produced by the perception $\mathcal{P}_i$. Defines the set that contains all the relevant features $\Lambda^{\mathcal{R}}$ where $(\Lambda^{\mathcal{R}})^C = \Phi$, the complete perception is when $\Lambda_i = \Lambda^{\mathcal{R}}$, and the evaluation function $\mathcal{F}$ is based on the observed world features and it should satisfy the following equations that:

$$\forall \Lambda \neq \Lambda^{\mathcal{R}}$$

$$\mathcal{F}(\Lambda) < \mathcal{F}(\Lambda^{\mathcal{R}}) \tag{2.3}$$

$$\forall \Lambda_i \subseteq \Lambda_j$$

$$\mathcal{F}(\Lambda_i) \leq \mathcal{F}(\Lambda_j) \tag{2.4}$$

Thus we obtain the model for the objective evaluation function $\mathcal{F}$ which in the domain of perception $\mathbb{P}$ that:

$$\underset{\mathcal{P}_i}{\operatorname{argmax}} \mathcal{F}(\Lambda_i) \tag{2.5}$$

$$\begin{aligned} s.t. \qquad & \mathcal{P}_i \models \Lambda_i. \\ \forall \Lambda \neq \Lambda^{\mathcal{R}}, \quad & \mathcal{F}(\Lambda) < \mathcal{F}(\Lambda^{\mathcal{R}}). \\ \forall \Lambda_i \subset \Lambda_j, \quad & \mathcal{F}(\Lambda_i) \leq \mathcal{F}(\Lambda_j). \end{aligned}$$

The perceptual incompleteness here can be ascribed to two factors. First, as state above, the sensors are trapped with their limitations that it is somehow impossible to attain the complete data in the world, which is expressed as:

$$\forall \mathcal{P}_i \in \mathbb{P}$$

$$\mathcal{P}_i \subset \mathcal{P}^{\mathcal{F}} \tag{2.6}$$

where $\mathcal{P}^{\mathcal{F}}$ is the full observation. The second factor is that even a full observation was to be achieved, the physical obstructions and computational costs can provide constrains to the production of observed features in the world that:

$$\nexists \mathcal{P} \models \Lambda^{\mathcal{R}} \tag{2.7}$$

Thus, the directly and indirectly methodologies that researcher are trying hard to handle the perceptual incompleteness problem and improve the effectiveness of the robot systems are basically dealing with the perception $\mathcal{P}$ and the interpret models $\models$ from $\mathcal{P}$ to Λ so that the intersection of Λ and $\Lambda^{\mathcal{R}}$ can be maximized:

$$\underset{\mathcal{P},\models}{\operatorname{argmax}} (\Lambda \bigcap \Lambda^{\mathcal{R}}) \tag{2.8}$$

where $\mathcal{P} \models \Lambda$. To solve this, researchers have done enormous significant progresses in recent decades.

2.2 Active Range Sensing

The most intuitive solution is to improve the perception ability, namely the $\mathcal{P}$ in Equation 2.8. To do so, the sensor systems need to be newly created or upgraded. The development of 3D range sensing starts at 1980s [31][63] in laboratories and these sensors become very popular in robotics applications after the 1990s. Thanks to the early development of 3D sensory technologies[44], in the beginning of this century the specifications of the range sensors have improved a lot like the accuracy, frequency and resolution.

2.2.1 Lidar Sensor

Lidar is a general surveying method for range measuring. The very popular Laser Range Finders (LRFs), a typical type of Lidar sensors such as the initial version of Hokuyo LRF [49][50] and another well-known SICK LRFs [64], have been widely used in a lot of robot systems. These LRFs have relatively long ranges of over 30$[m]$ with errors less than 2$[cm]$, which makes them indispensable to measuring outdoor distances. For example, the Stanford team [65] in the DARPA Urban Challenge used a 3D lidar sensor named Velodyne to support autonomous vehicle driving.

These Lidar sensors share the same technical perspective of basic phase shift ToF measurement. As Figure 2.1 shows, the illumination signal as note as s^i, which after being reflected from the object and received by the APD(Avalanche Photo Diode) sensor note as s^r. Inside the sensor the s^i is also collected as reference signal for synchronization and note as s^{ref} and s^{ref} can have multiple phases to help calculation. Since the emit light travels a distance to the object and get reflected back, there will be a time delay between s^{ref} and s^r, which will results in the phase difference ϕ. The distance can be expressed as since the s^i is a periodic signal:

$$L = \frac{1}{2} \times \frac{cT\phi_d}{2\pi} = \frac{c\phi}{4\pi f} \tag{2.9}$$

where c is the speed of light in certain medium and the f is the signal frequency. Since the reflected incident signal s^r is correlated with the reference signal s^{ref}. The general method yields the correlation function:

$$Corr(s^r, s^{ref}) = \int_{-\infty}^{\infty} s^r(t) s^{ref}(t+\tau)\,\mathrm{d}t \tag{2.10}$$

Then the ϕ can be obtained by processing the correlation function. According to Equation 2.9, the higher the frequency of the periodic signal f is, the higher the

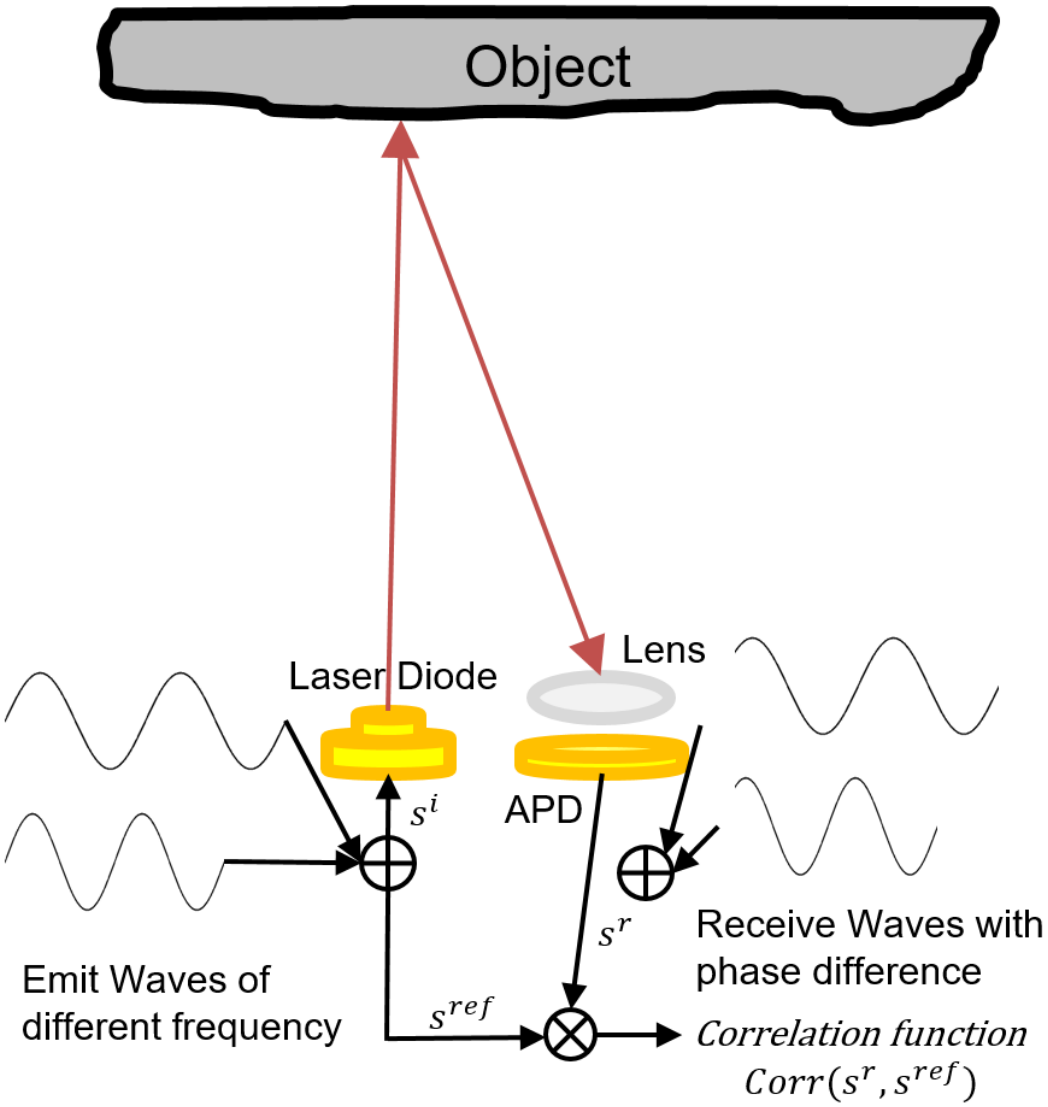


Figure 2.1: Phase Shift ToF Measurement(The frequencies of emitted signals are depending on the targeting measurement distance and system sampling rate.)

distance resolution are. However, this will lead to a consequence that for longer distance, the reflected light signal may exceed one period if the frequency is too high. To address the dilemma, signals with different frequency are emitted through the laser diode alternately like in [50] used two signal waves of frequency $46.55[Mhz]$ and $53.2[Mhz]$. The lidar sensors use spin motors to obtain 360 degree scans, which sacrifices the size of the sensors and makes them impossible for some smaller applications. In addition, the cost of such sensors is prohibitive for common use with the minimum cost being at least several thousand US dollars.

Researchers have been investigating cheaper and more accessible designs for lidar sensors. Konolige et al. [45] proposed a laser range finder sensor using the triangle measuring principle . This approach installs a point laser emitter and a line Charged-Coupled Device (CCD) sensor with a fixed baseline, obtains full 360 degree $6[m]$ range measurements with less than $3[cm]$ errors, and costs about $30. Despite of the measurement range and frequency, this kind of solution seems promising for ordinary commercial applications.

2.2.2 3D Camera

The development of CCD and CMOS technology make it possible to build a static compact 3D camera sensor system rather than mechanically scanning measurement while yield line and plane range measurement. There are basically two popular type of measuring principles, the structured light and the ToF based measurement.

The structured light approach borrows the geometrical theory from the stereo vision and tackles the very difficult matching problem by projecting the decoded patterns. The very common methods use binary and gray coded fringe patterns [66][67]. Multiple frames are used so that the pattern matching can be realized for higher resolution. As the structured light methods rely on triangulation principle, the measurement occlusion of convex surface are dealt by install multiple projectors or multiple receiver cameras. Since 2010, Microsoft and PrimeSense published structured light based RGB-D cameras, namely Kinect and PrimeSense, which initiated a huge revolution in the indoor range sensing field. Thanks to the reverse engineered results from the ROS community[68], the methodology details of these sensors are deduced since there are no official documents from the manufacture. The high precision, small size and affordable price of these sensors make them perfect for robot applications and the skeleton detection SDKs from Microsoft and OPENNI accelerate the research of SLAM [69], 3D reconstruction [70], human-robot interaction [71] and etc.

Most recently, Microsoft published the second version of Kinect [72], namely Kinect v2, which is based on the TOF principle, improving the resolution and accuracy of both RGB and depth images where the details of comparison from these two versions of Kinect can be found [73]. The study of TOF range cameras began at decades ago and there are mainly three approaches are currently employed.

The phase shift based principle which is also widely applied in LRFs is adopted in Kinect v2. Consider equation 2.10, the reflected incident signal can be noted as s_i^r of the ith pixel incident signal. Thus the correlation function turns out to be a vector or an image matrix in line [74] and 2D image [75] cases respectively. Since the sinusoidal is widely used in this case and the reference signal and incident signal can be expressed as:

$$s^{ref}(t) = cos(\omega t) \tag{2.11}$$

$$s_i^r(t) = A_i cos(\omega t - \phi_i) + C \tag{2.12}$$

then the correlation function of the ith pixel is given as:

$$C_i(\tau) = Corr(s_i^r, s^{ref}) = \frac{A_i}{2} cos(\phi_i + \tau) + C \tag{2.13}$$

since the reference signal is well known and we choose the phase to be $\tau = \frac{n\pi}{2}, n = 0, 1, 2, 3$, thus the phase and amplitude of the ith pixel can be noted as:

$$\phi_i = \arctan(\frac{C_i(\tau_3) - C_i(\tau_1)}{C_i(\tau_0) - C_i(\tau_2)}) \tag{2.14}$$

$$A_i = \frac{1}{2}\sqrt{(C_i(\tau_3) - C_i(\tau_1))^2 + (C_i(\tau_0) - C_i(\tau_2))^2} \tag{2.15}$$

The second approach is based on an optical shutter technology [76][77]. For the shift based method, so many parameters need to be considered and the system is relatively complex, which is suitable for measurement of more than 3 metres but results in high expense. This approach makes it possible to make low-cost commercial products with acceptable accuracy especially for local sensing applications. [51] claimed that there are only a few of researches about this kind of approach and the existing one just reveals little details.

Another principle is newly proposed in recent years [78][79][80]. It is basically a CMOS time-of-flight (TOF) range image sensor using single-layer gates on field oxide structure for photo conversion and charge transfer. The n-type buried layer enables the high-speed charge transfer for the sensor thus by simply comparing the ratio

of capture electrons between two charge drains. Since this approach only depends on slightly improvement of CCD sensors, the cost can be reduced and more pixels resolution can be easily achieved. Based on this measurement principle, we designed the hardware circuits and calibration methods for this type of commercial sensors. In the next chapter we will evaluate our intact sensor system and explain how we address the problems which previous works did not consider and for the following chapters we demonstrate how we customize and apply our flexible laser system in various of difficult robotic applications and evaluate the effectiveness of our sensor in these applications.

2.3 Reasoning and Local Sensing based Vision Cognition

2.3.1 3D Visual Reasoning

Back to the equation 2.8, except directly improve the $\mathcal{P}$, researchers also came up with idea that by accumulating multiple observations and combining them together, the agents are able to generate better world features to make decision of next action since it is obvious that:

$$\forall \mathcal{P}_i \models \Lambda_i$$

$$\mathcal{P}^n = \bigcup_i^n \mathcal{P}_i \models \Lambda^n \tag{2.16}$$

$$\forall \Lambda_i, \exists \Lambda_i \subset \Lambda^n \tag{2.17}$$

There are some researches [81][82][83] tackled in this methodology and the difficulty is to solve the very tradition NBV(Next Best View) [84] problem so that the observation can generate the most useful information toward to task.

Recently, another popular methodology attracts the interests of researcher since the profuse 3D data provide a new perspective on 3D physical and geometrical vision reasoning. Reasoning-base vision recognition generally consists of four aspects: Functionality, Physics, Intentionality and Causality[85, 86, 87, 88]. Research in this area focuses on determining "what is where", and trying to add the very fundamental knowledge to the vision system to achieve reasonable results. The idea of utilizing fundamental knowledge could help solving certain difficult object detection problems. Grabner *et al.*[85] suggest that the detection of chairs and tables should not only con-

centrate on the geometric features, but the functionality such as the definition of the action of "sitting" by humans. In addition, Bo Zheng *et al.* [58] provided methods for building a more "Reasonable" 3D pointcloud map in indoor environments that can infer potential falling objects based on pointcloud physical relations and observation of human daily actions.

Instead of directly handling the NBV problem to obtain more observations, This approach makes guesses of the unobservable parts of the objectives according to the prior knowledge of the scene. It can complete the 3D missing part of the object from the unreasonable points from the sensors since all the detected voxels have to satisfy the constraints of the physics and geometry [86]. It can also generate the stability of the object by considering the geometrical stability of the object with respect to he gravity[58]. The insight of this methodology is improving the features generation model $\models$ in equation 2.8, where:

$$\forall \mathcal{P}_i \models \Lambda_i$$
$$\exists \mathcal{P}_i \models_{reasoning} \Lambda_i \bigcup \Lambda^r \tag{2.18}$$

where Λ^r is the guess of features from reasoning model $\models_{reasoning}$. According to equation 2.4, obviously we have:

$$\mathcal{F}(\Lambda_i) \leq \mathcal{F}(\Lambda_i \bigcup \Lambda^r) \tag{2.19}$$

which demonstrates the importance of this approach to the effectiveness of robot systems. However, since basically this method only gives a guess of the potential world features, the generated features set Λ^r should contains probabilistic parts and the final justification from other observation is required so that the validation and accuracy can be guaranteed. Under this framework, this approach shares the same perspective of NBV methodology while evaluation of the NBV for reasoning approach is task-oriented. Local sensing is designed to handle with this problem.

2.3.2 Local Sensing

Local sensing is an essential complement to high level sensing and reasoning with respect to the environment uncertainty and complexity. [56] concludes the three main features of the local sensing: the low level relative output information, the size and weight of sensors and the "reflex" like immediate response.

Local sensing is widely used in swarm robotics[89] since in swarm robotics an individual robot is a simple behaviour agent that just takes actions according to the

sensory data and internal memory [90]. Therefore, the global interpreting of the perception data is not necessary.

A more general local sensing framework is applied in manipulation using proximity sensors. Proximity sensors are designed to handle short-range for proximity measurement so that these kind of sensors can only provide limited information inadequate for recognition and high level processing [30]. Proximity sensors are also functionally different from tactile sensors [91] since the proximity sensors ensure the effective grasp [92] or safety collision detection [93] and the tactile sensors provide contact information.

The proximity sensors are often provides distance data by acoustic and optical measurement. In a lot of applications, optical methods are appreciated since the measuring frequency is much higher than using sound information. [94] proposed a high speed net-structure proximity sensor using reflection of infrared light. Attribute to their novel design and analog processing, the time consuming complex processing in CPU is avoided and the system can response at $1khz$. Therefore the demonstration of robust grasping of robot hand can be shown in [95]. The size, weight, resolution and real-time ability are the main features of a proximity sensor for local sensing.

The combination between reasoning based vision recognition and local sensing will result in reasonable and reliable actions for the task-oriented robot systems. We will demonstrate our framework in the later chapters.

Chapter 3

Development of Lightweight Tiny TOF Laser Line Sensor System

3.1 Introduction

For robotic sensing, depth sensing plays an important role because accurate range measurements are helpful for reconstructing scene geometries, which is required by tasks of manipulation, SLAM, and scene understanding. In terms of accuracy in these applications, laser range sensors provide more accurate and dense distance measurements compared to other kinds of sensors such as sonar and IR sensors.

There are several popular laser range sensors in commercial market and dramatically accelerate the development of robot systems. Lidar(Light Detection And Rangeing) systems are widely used in outdoor environment like autonomous driving [96][97], field mobile robot SLAM [98] and drone [99]. Lidar systems are good at range sensing from tens of centimetres to several tens of metres for both indoor and outdoor applications and the error can be as less as several centimetres. However, most Lidar systems equipped with motor executers so that the size, weight and power consumptions are less considered. Besides, the Lidar systems are not as affordable as other type of sensors and the price is from thousands of US dollars to tens thousands of US dollars. Another prevalent type is the RGB-D cameras, like the well-known Microsoft Kinect. These sensors were supposed to be designed for gaming market, the good performance and affordable price make them very popular in robotics research [69] [70].

In Chapter 2 we mentioned that there are some limitations from the sensors in robot systems. Especially for the monetary cost of sensors like the price, size and

weight, are not well considered in most of the commercial products and researches. These unconventional sensors are very necessary and can be applied in robot local sensing for task-oriented robot systems like small mobile robot and drone due to the small size and light weight.

In this chapter, we describe the design of a tiny line laser range sensor based on the TOF principle [78][80] for unconventional applications with strict requirements on size and weight. We also analyse the systematic errors and design efficient calibration methods for the sensor. The sensor has the following specifications:

1) Lightweight and tiny with size of $35[mm] \times 27[mm] \times 30[mm]$ **and weight of** $20[g]$**.**

2) Powered by a Single USB Power Supply.

3) Power Consumption: $< 1[W]$.

4) Measurable Range: $0.05[m] \sim 3[m]$.

5) High Frequency Measurement: higher than $60[hz]$ within a shorter range.

6) Low Cost: less than $150 for a prototype.

3.2 Sensor Hardware Design

This section discusses the hardware architecture of the sensor. The following subsections give details respectively on the hardware design of the circuits board, the optical design, and the logic timing. For hardware details, please refer to our open-source repository[1]

3.2.1 Architecture Overview

As Figure 3.2 shows, the sensor's main components are a laser diode, a range sensor, optical attachments, and a 32-bit Micro Control Unit (MCU) with powerful peripheral circuits. The main parts of the system include:

Laser Emitter: The $905[nm]$ laser diode is programmed to emit pulsed light with a constant pulse width of around $40[ns]$, and the pulse duty rate is adjustable for laser eye-safety. The pulse signals are generated by the $4.608[Ghz]$ high resolution timer and then the IC propagation delay and MOSFET delay can be accurately issued by the hardware configuration.

Range Sensor Receiver: The range sensor receiver we use is recently released by Hamamatsu Corp, which is equipped with Ghz level switches and a high speed

[1] `https://github.com/cretaceous-creature/jsk_laser`

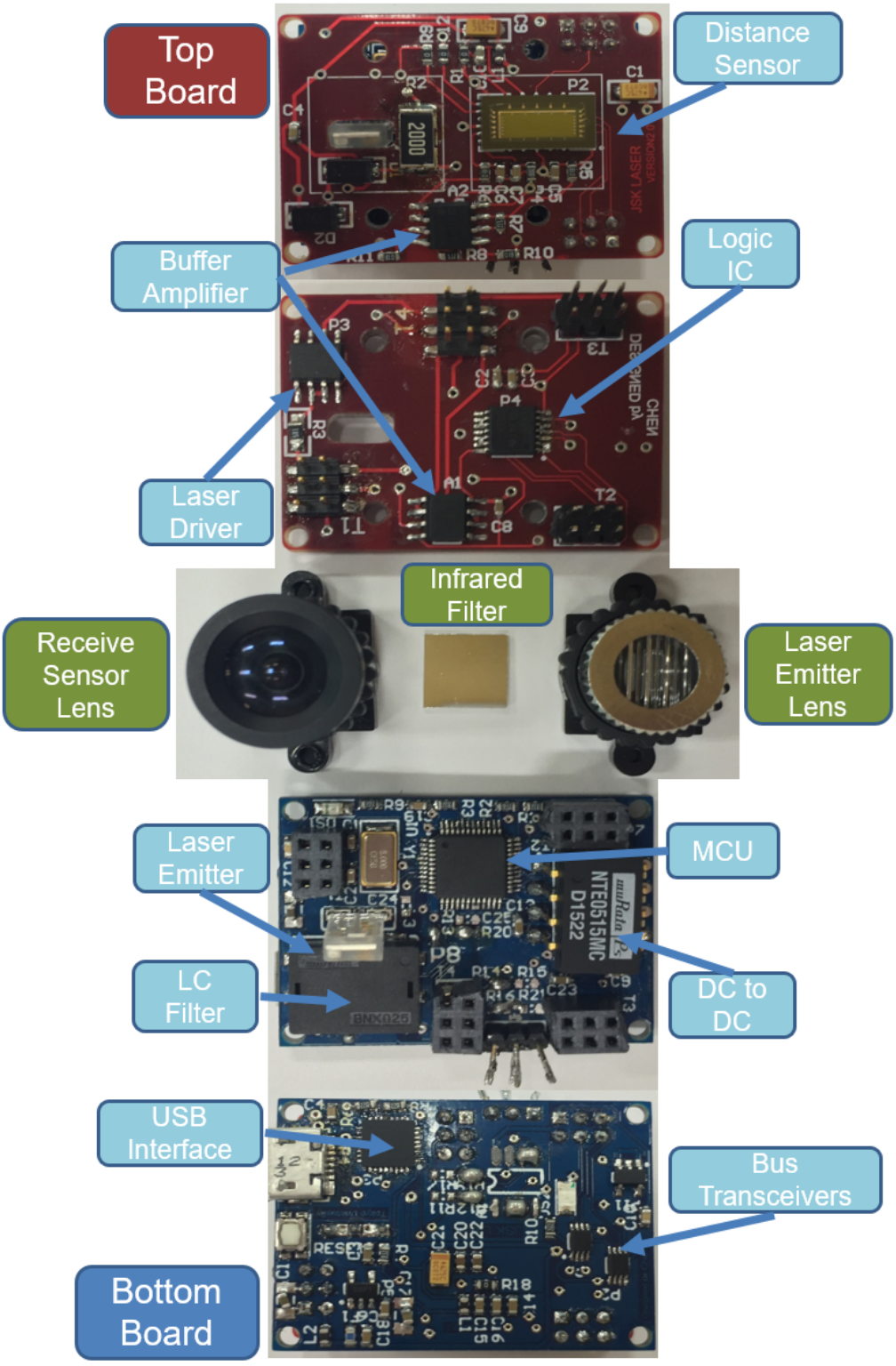


Figure 3.1: Hardware overview(The red PCB board is the top board and the blue is the bottom one; The PCB boards have 4 layers and the inner 2 layers are the power signals)

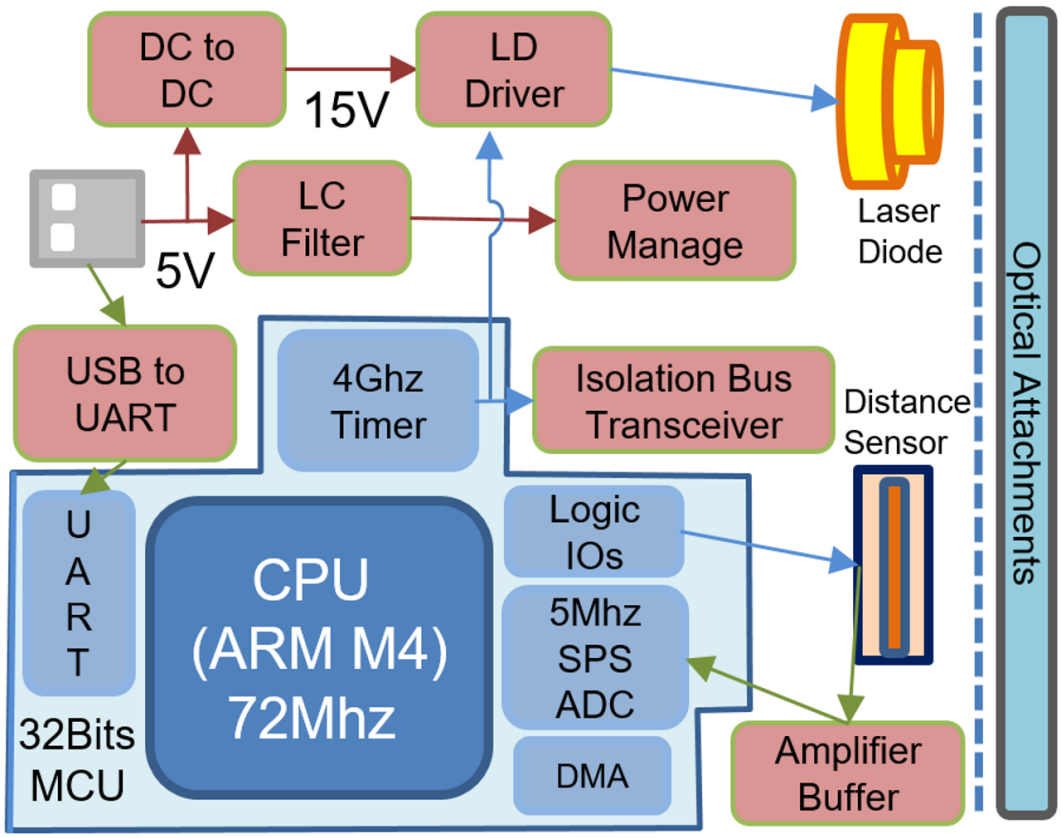


Figure 3.2: Hardware architecture flowchart. (The meanings of the arrows. Red: Power connections; Green: Data flows; Blue: Logic control signals. Illustration of the details in section 3.2).

charge transfer pixel structure. We use the 4.608$[Ghz]$ high resolution timer to control the logic timing of the high speed switches. The output of the sensor is the voltage reading of the inner capacitors. Through the amplifier buffers, the output signals are collected by the ADCs and transferred to the memory by the DMA module. The signal data is then transformed to distance measurements based on calibrated parameters.

Optical Attachments: This sensor is attached to both a laser lens and a camera lens. The laser emitter uses a line laser lens with emitting angle ranging from $30° \sim 120°$. The corresponding camera lens with similar field of view is attached to the range sensor receiver. Without compromising the size, the aperture of the lens is less than $F2.0$. To reduce influence from ambient lighting, we also equip an infra-red narrowband filter on the camera lens. The measurable range can be customized by selecting lenses with different fields of view.

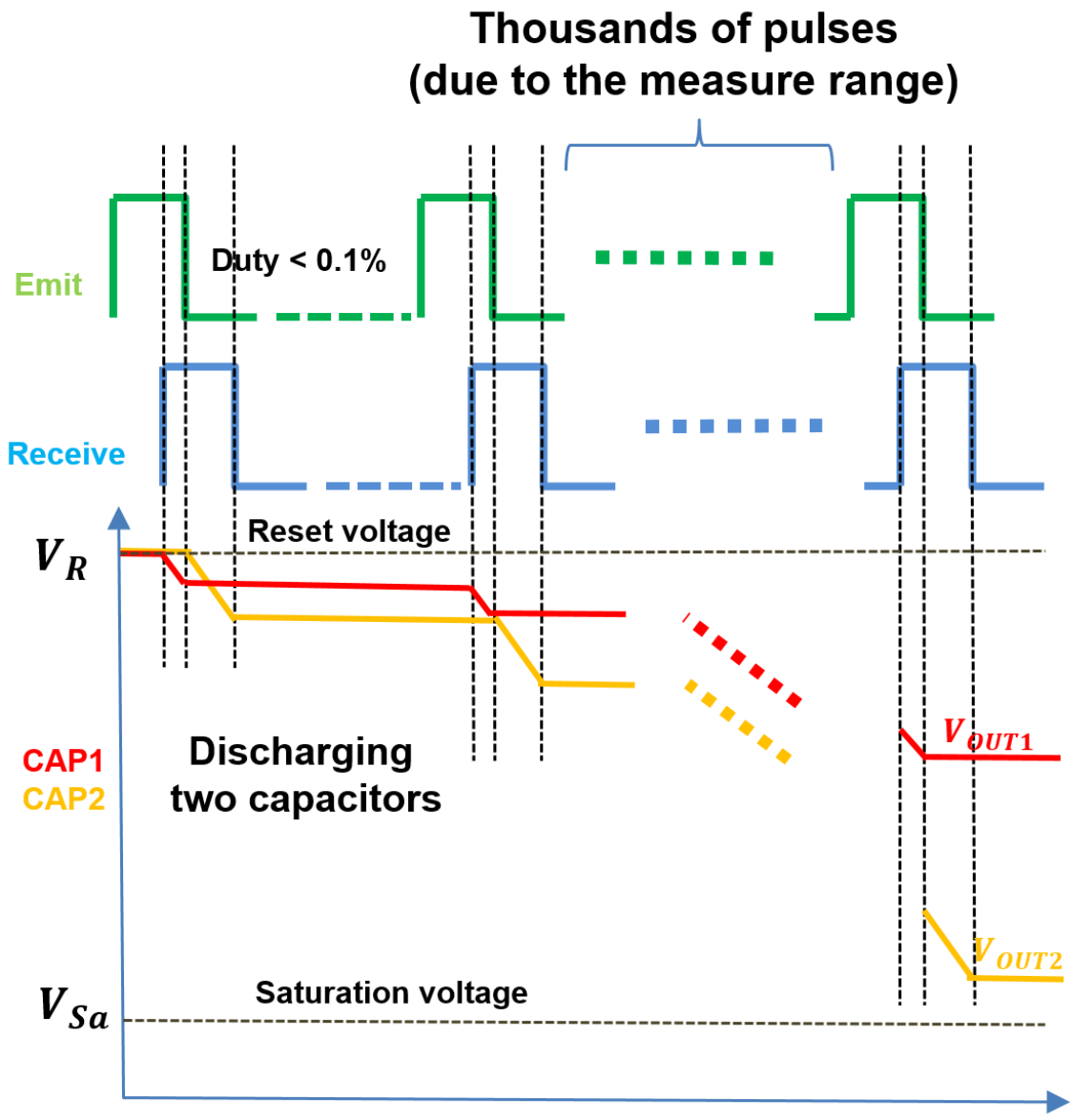


Figure 3.3: Illustration of sensor operation flow(The reset voltage and the saturation voltage are constant when give a constant reset pixel power supply; The range of the duty rate is constrained by the laser diode limitation and laser safety consideration).

Micro Control Unit: The main controller we use is a 32-bits MCU integrated with powerful peripheral circuits, including 12-bit Analog-Digital Converters (ADCs) with 5 Mega Samples per Second (MSPS) converting rate and a high resolution timer with up to 4.608$[Ghz]$ frequency and approximately 217$[ps]$ resolution. These peripherals make it possible to provide accurate calibration of the sensor timing sequences.

3.2.2 Logical Timing

According to the datasheet of the range sensor [100], there are two capacitors for each pixel inside the sensor, both of which are connected to the photosensitive area through a high speed switch. When the reflected light is received, the capacitor with corresponding switch turned on will discharge until a saturation voltage is reached, as Figure 3.2 shows.

The whole timing logic is shown in Figure 3.2 and Figure 3.6, which illustrate the basic TOF measurement principle. In Figure 3.6, Q_1, Q_2 are the direct measurements of the capacitors, T_0 is the pre-set constant laser pulse width, and T_d denotes the unknown delay of the reflected light. Because the distance can be derived from T_d using the speed of light, we would like to infer T_d from the direct capacitor measurements. Assuming ideal situations when the power of the emitted light and the reflected light are constant, note as the ideal waveform shown in Figure 3.6, by the characteristics of the capacitors we have:

$$\frac{Q_2}{Q_1 + Q_2} = \frac{T_d}{T_0}. \tag{3.1}$$

Under ideal conditions when the two capacitors of a pixel have the same characteristics, we have:

$$T_d = \frac{T_0 V_2}{V_1 + V_2}, \tag{3.2}$$

where V_1, V_2 are the voltage outputs, linear with respect to Q_1, Q_2, the measurements of the range sensor. For all the 256 pixels, there is:

$$\boldsymbol{V} = \left[\frac{V_2^{(1)}}{V_1^{(1)}+V_2^{(1)}} \quad \frac{V_2^{(2)}}{V_1^{(2)}+V_2^{(2)}} \quad \cdots \quad \frac{V_2^{(256)}}{V_1^{(256)}+V_2^{(256)}} \right]^{\mathsf{T}}, \tag{3.3}$$

where $V_1^{(n)}, V_2^{(n)}$ are the outputs of the nth pixel. With delay $T_d \in [0, T_0]$ and measured distance of the nth pixel $L^{(n)} \in [0, \frac{cT_0}{2}], n = 1, \ldots, 256$, we have:

$$\boldsymbol{L} = \frac{cT_0}{2}\boldsymbol{V}, \tag{3.4}$$

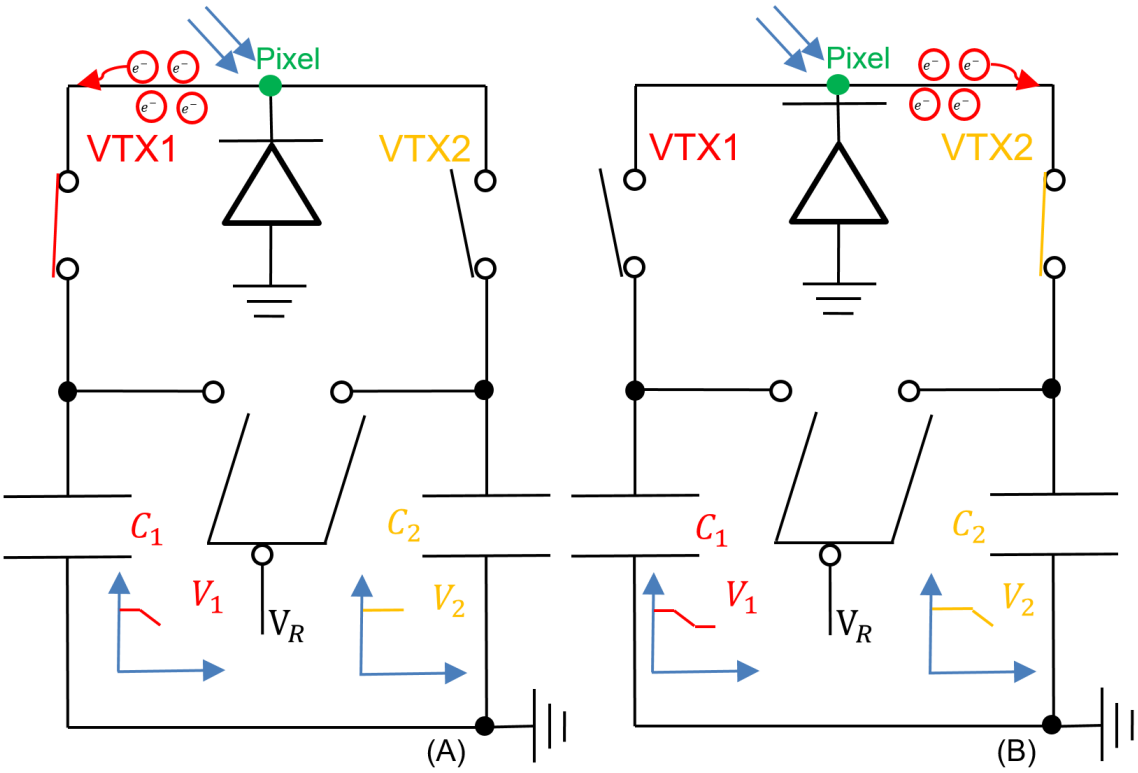


Figure 3.4: Distance sensor pixel structure. (A)Switch VTX1 on, VTX2 off, capacitor C_1 discharges. (B)Switch VTX2 on, VTX1 off, capacitor C_2 discharges.

where c is the speed of light of $3 \times 10^8[m/s]$. If we set $T_0 = 40[ns]$, the measurable range is $[0, 6]$ meters theoretically.

3.2.3 Laser Diode Charge and Drive

We applied the Osram SPL LL90-3 [101] hybrid pulse laser diode(PLD) emitter in our hardware. To achieve the sudden high and dense power pulse energy within tens of nano seconds, capacitors of about $100nf$ are placed near the laser diode and get charged by the power supply as Figure 3.5 shows. The trigger signal connects with a current limitation resistor to the gate of the MOSFET with a input capacitor of about $300pf$. Since the rise time is a very important factor in our system, we need to charge the MOSFET input capacitor as soon as possible. Thus a resistor of $5.1Ohm$ is used and result in high charge current of about $1A$ with several nano seconds for trigger signal. Therefore the trigger signal is generated by a high speed power MOSFET driver which itself can be directly triggered by TTL-level IOs. The charge resistor $R1$ is related to the pulse frequency since to fully charge the capacitor C, about $3 \sim 5$

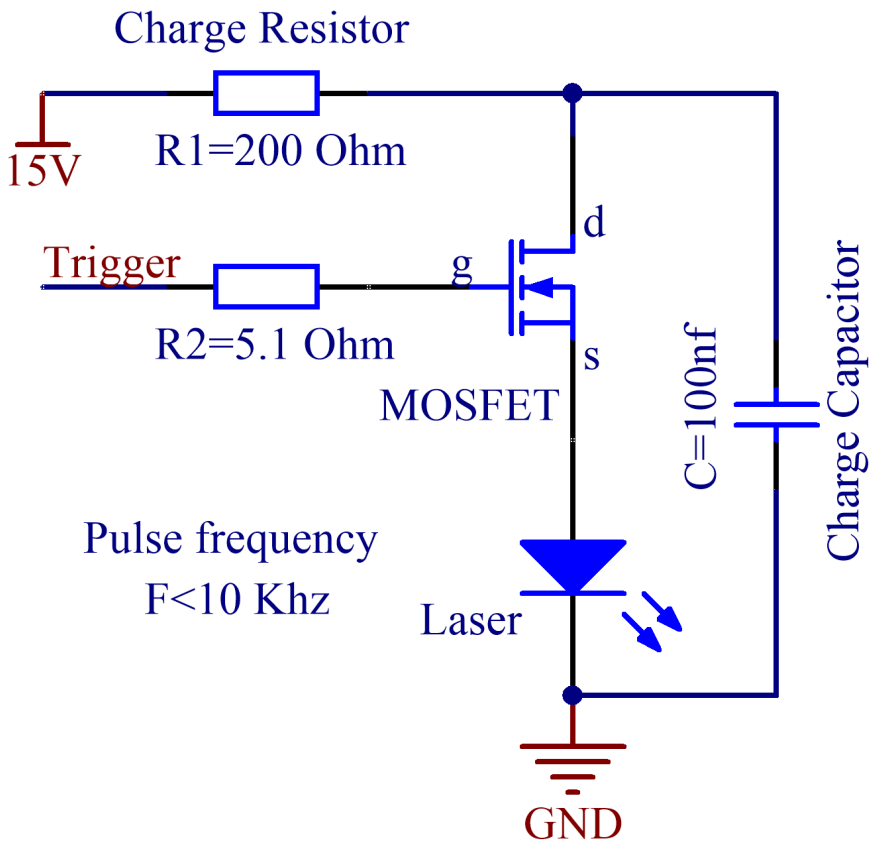


Figure 3.5: Laser Drive Corresponding Circuits: The MOSFET, Laser Diode and Charge Capacitor are included in the hybrid laser module SPL LL90-3; The value of the resistors are carefully chosen to satisfy our system.

times constant RC where in this case $R1 \times C$ is required. Reducing the $R1$ will lead to the high current on $R1$ thus a bigger resistor package should be used. Considering the maximum duty rate of pulse diode is 0.1% and the resistor size we choose the $200Ohm$ power resistor and could achieve the maximum pulse frequency of $10Khz$.

3.2.4 Data Acquisition and Correction

According to Equation 3.1, distance information is derived from the quantity of electrons discharged by the reflected light between the two capacitors. Before measuring, each sensor pixel needs to be reset, namely, recharging the two capacitors of each pixel. The voltages of a certain the two capacitors after reset are denoted as V_{R1}, V_{R2}. Note that the measurements voltage V_1, V_2 are the offset voltages compared to the reset voltage V_{R1}, V_{R2}.

To acquire the data V_{R1}, V_{R2}, we follow the logic timing of the range sensor and the output connection to the MCU's two ADCs through the amplifier buffers. The data is handled by the system DMA and directly moved to the memory or the serial port (UART) for later processing. There is a saturation problem that requires careful handling. The measurement is done by accumulating energy of the reflected light in the capacitors. Within a frame of 256 pixels, some pixels will become saturated earlier than other pixels due to different measurement ranges. In order to obtain both short range and long range data without being saturated, we continuously read out and store the data, and if the current data saturates, we approximate its current value using the value from the last frame.

Because the two capacitors do not have exactly the same characteristics, we consider sensitivity ratio, leakage voltage and other factors as indicated in the datasheet of the camera sensor. The suggested distance estimation equation given by the manufacturer is:

$$L = a\frac{V_2 - V_{\text{le}}}{(V_1 R_{\text{sen}} - V_{\text{le}}) + (V_2 - V_{\text{le}})}\frac{cT_0}{2} - D_{\text{ofs}}, \tag{3.5}$$

where V_1, V_2 are the discharging amount of the capacitors and R_{sen} is the sensitivity ratio between the two capacitors of each pixel, V_{le} is the leakage amount of the capacitors, and a, D_{ofs} are the linear fitting coefficients of the equation. However, we find this correction model fails to produce adequate results. In the next section, we will analyze major errors and provide our method for calibration.

3.3 Calibration Analysis

As in the literature [78], the experiments use the linear model and only perform the experiment by sending delay to the logic timing circuits and simulate the results of changing the target distance. It is inevitable that some of the systematic errors are ignored. Firstly, the light source effect may bring non-linearity part into the system. When we have assumed that the emitted light and the reflected light have constant power, the resulting expression is straightforward. However, the actual light power is rarely constant, and it is more like a quadratic function [101] as Figure 3.6 shows. Secondly, the dynamic measurement operation results should be considered since the reflected light amount is no longer constant and it is necessary to consider the effects of different reflected light amount. Lastly, as a system, it is better to consider what kind of factors like the temperatures, reflectance of the objects and ambient light may

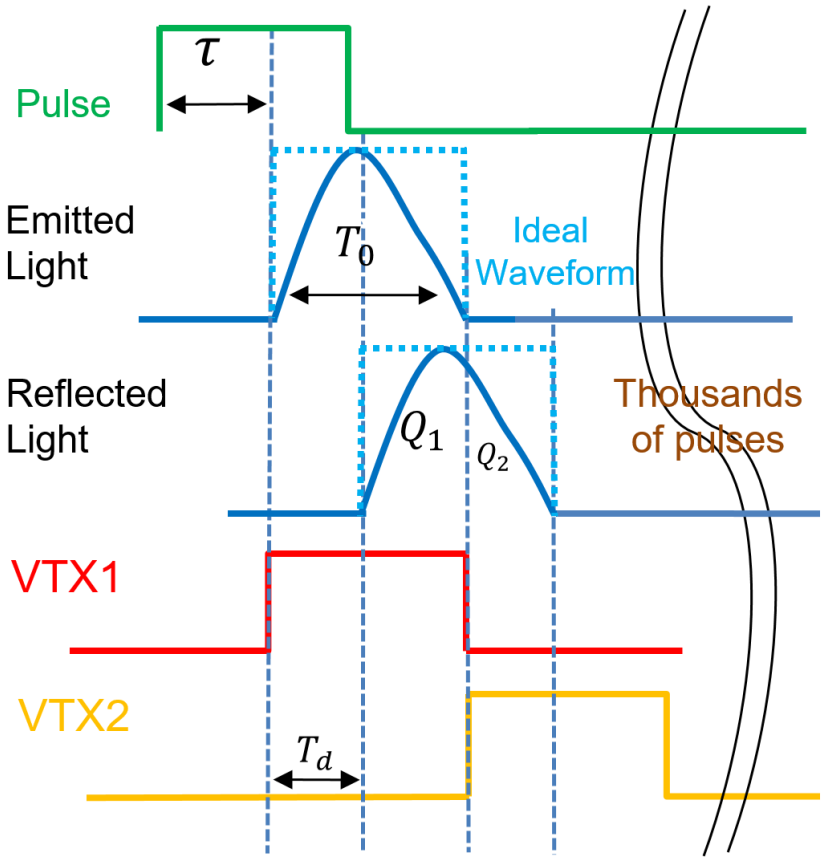


Figure 3.6: Logic timing.(Demonstration of logic timing for one pulse, the sensor may need to accumulate hundreds or even thousands of pulses to due to the measure range)

introduce considerable effects to the measurement. In this section, we will explain how we address the problems above based on our hardware system.

3.3.1 Hardware Pulse Delay Analysis

As Figure 3.6 shows, there is an unknown but constant time delay τ between the pulse signal and emitted light. This is because the MOSFET driver has an input capacitor and needs time to charge. This will not be a problem for ordinary usage, but at nanosecond level, a 300$[pf]$ input capacitor requires several nanoseconds of charge time at an input current of 1$[A]$. The MCU I/O interface cannot provide such current, hence the MOSFET driver introduces almost 20$[ns]$ delay. Thanks to our 4.608$[Ghz]$ high resolution timer, we can estimate and calibrate this delay. An inaccurate estimate of the time delay $\tilde{\tau}$ will cause the measurement Q_1 to have a bias of ΔQ and Q_2 to have a bias of $-\Delta Q$. According to the ideal waveform of constant

emitted and reflected light, there is a linear relation $\Delta Q \propto \tilde{\tau} - \tau$. Plugging ΔQ into Equation 3.1, the biased measurement $\tilde{T}_d$ is:

$$\begin{aligned} \tilde{T_d} &= \frac{Q_2 - \Delta Q}{(Q_1 + \Delta Q) + (Q_2 - \Delta Q)} T_0 \\ &= T_d - \frac{\Delta Q}{Q_1 + Q_2} T_0 \end{aligned} \tag{3.6}$$

Therefore, the true T_d can be obtained with:

$$T_d = \tilde{T_d} + kT_0, \quad k \propto \tilde{\tau} - \tau. \tag{3.7}$$

Because T_0 is a known constant, this shows that the bias in the estimation of the pulse delay $\tilde{\tau} - \tau$ only results in a linear error and can be corrected by adding a constant offset, which value is easy to obtain by using an oscilloscope. The $4.608[Ghz]$ high resolution timer can handle this since our offset resolution is approximately $217[ps]$. The use of the high resolution timer will help calibrate the data at hardware level by slightly adjusting the logic timing of the circuits. As Figure 3.7 shows, we are perfectly adjust the delay from the pulse signal to the PLD driver signal, namely, the time when the laser emits the light attribute to the usage of high resolution timer in our hardware system.

3.3.2 Errors from Non-constant Reflected Light

Considering the reflected light to be non-constant, Equation 3.1 fails and we have the inequalities:

$$\frac{Q_2}{Q_1 + Q_2} \leq \frac{T_d}{T_0}, \quad T_d \in [0, T_\sigma) \tag{3.8}$$

$$\frac{Q_2}{Q_1 + Q_2} > \frac{T_d}{T_0}, \quad T_d \in (T_\sigma, T_0] \tag{3.9}$$

where T_σ is the peak of the reflected light power. First we need to prove the uniqueness of the T_d which corresponds to the voltage outputs. Without loss of generality we assume the reflected power is $f(x)$ and $G(x)$ and satisfies $G(0) = 0$. We have:

$$Q_1 = G(T_d) = \int_0^{T_d} f(x)\,\mathrm{d}x \tag{3.10}$$

$$Q_1 + Q_2 = G(T_0) = \int_0^{T_0} f(x)\,\mathrm{d}x \tag{3.11}$$

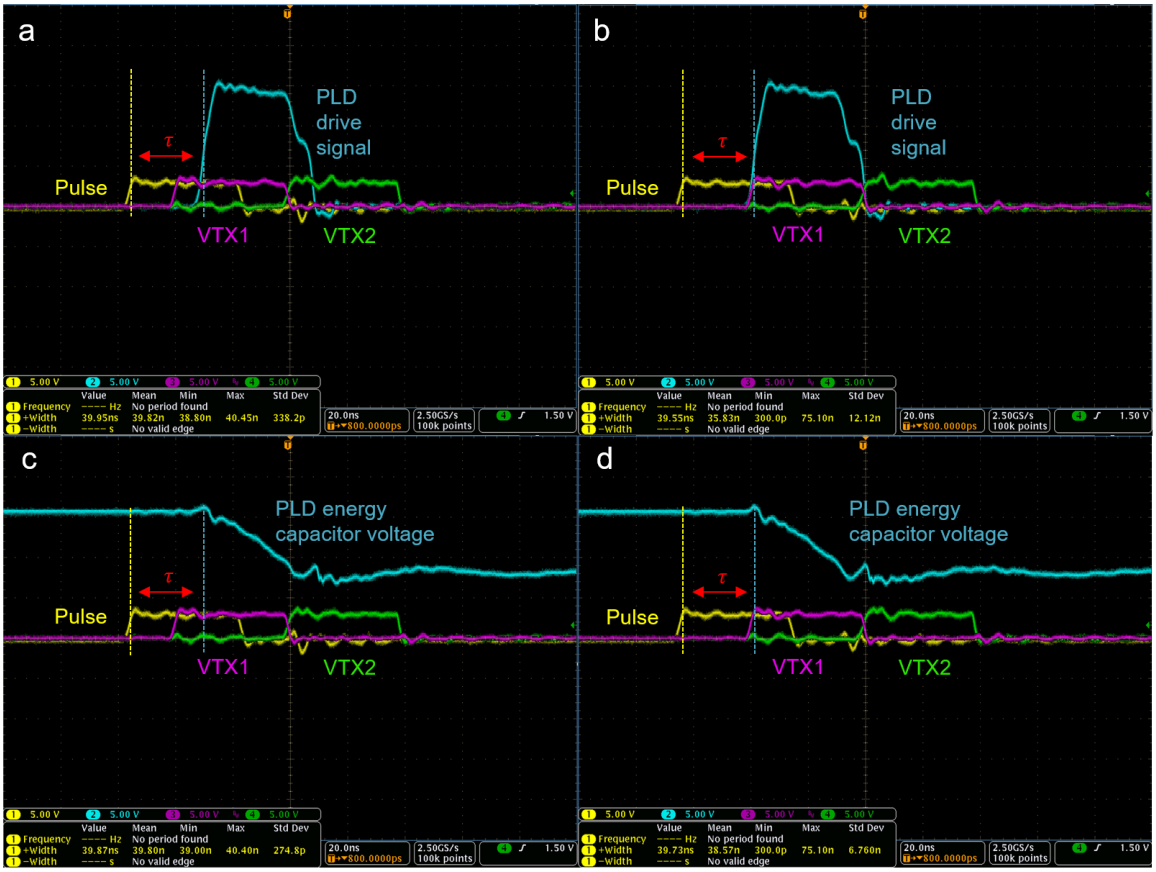


Figure 3.7: Adjust the logic timing.(This figure shows the hardware signal using oscilloscope; Signals pulse, VTX1 and VTX2 are controlled by the high resolution timer; PLD(Pulsed Laser Diode) drive signal refers to position of hardware 3.5 g; PLD energy capacitor voltage refers to the position of hardware 3.5 d, when the PLD emits light and consume the power, potential in d drops.)

Because $f(x) > 0$ in its domain $[0, T_0]$, $G(x)$ is monotonic continuous and have the extrema at 0 and T_0. Thus, we can prove that $\exists! T_d \in [0, T_0]$ which satisfies:

$$1 - \frac{T_d}{T_0} = \frac{Q_1}{Q_1 + Q_2} = \frac{G(T_d)}{G(T_0)} \tag{3.12}$$

This means that the correspondence relationship between T_d and $\frac{Q_1}{Q_1+Q_2}$ exists and is unique with respect to the reflected waveform because the pulse width T_0 is a fixed constant. Considering Equation 3.8, and Equation 3.9, the model can be rewritten as:

$$T_d = \frac{Q_2}{Q_1 + Q_2} T_0 + \mathcal{P} \tag{3.13}$$

where intuitively, $\mathcal{P}$ is only dependant on $\frac{Q_2}{Q_1+Q_2}$ given the reflected waveform from the laser diode. However, the amplitude of the waveform Q_1+Q_2 varies from pulse to pulse and depending on surface material, the coefficient of reflection is also different even at the same distance. Thus $\mathcal{P}$, accounting for these factors, introduces non-linearity to the system. Considering this, we perform software calibration by collecting the measured data to fit a calibration function.

3.3.3 Comprehensive Calibration

Directly fitting the data to an unknown model is not intuitive and difficult for reviewing the error in the system when considering non-linearity. We first collect the data with constant discharge amount for each measured distance so that a comparison can be made by using different regression model.

We first record only one data point for each certain distance. The reflected light intensity range is fixed by controlling the amount of pulses to be emitted. For simplicity, here we only consider the calibration of the central one pixel. Assuming that the distance data set is $\mathcal{L}^{(n)}$ and

$$\forall L^{(i)}, L^{(j)} \in \mathcal{L}^{(n)} (i \neq j),$$

$$L^{(i)} \neq L^{(j)} \tag{3.14}$$

For every element, note as $L^{(i)} \in \mathcal{L}^{(n)}$, the corresponding outputs of the sensor are noted as $V_1^{(i)}, V_2^{(i)}$(Since the reset voltage V_{R1}, V_{R2} are constant, here $V_1^{(i)}, V_2^{(i)}$ stand for the discharging voltage of the two capacitors). The model could be written as two components:

$$\widetilde{L^{(i)}} = \widetilde{\text{linear}}(V_1^{(i)}, V_2^{(i)}) + \overline{\mathcal{P}} \tag{3.15}$$

where the linear part is the data correction model given in Equation 3.5:

$$\widetilde{\text{linear}}(V_1^{(i)}, V_2^{(i)}) = \frac{\mathcal{K}(V_2^{(i)} - V_{\text{le}})}{(V_1^{(i)} R_{\text{sen}} - V_{\text{le}}) + (V_2^{(i)} - V_{\text{le}})} - D_{\text{ofs}} \tag{3.16}$$

The leakage voltage V_{le} is related to the pulse times, and could be obtained by exposing the sensor under pitch-black environment and read the output without giving and light pulses. According to [78], this offset is partially caused by the deep generated electrons diffusing phenomenon. We obtain the sensitive ratio R_{sen} by exposing the sensor under constant ambient light and then calculate the output voltage ratio of the two capacitors. Then the problem to be solved could be transformed into an

optimization problem:

$$\underset{\widetilde{\text{linear}},\overline{\mathcal{P}}}{\text{argmin}} \frac{1}{n}\sum_{i}^{n}(L^{(i)} - \widetilde{L^{(i)}})^2 \tag{3.17}$$

$$s.t. \quad L^{(i)} \in \mathcal{L}^{(n)}.$$

According to our observation of the collected data, the relationship is more curved than linear, hence a combined model is made using a polynomial curve:

$$\widetilde{L^{(i)}} = \widetilde{\mathcal{P}}(\mathcal{X}^{(i)}) \tag{3.18}$$

where

$$\mathcal{X}^{(i)} = \frac{(V_2^{(i)} - V_{\text{le}})}{(V_1^{(i)}R_{\text{sen}} - V_{\text{le}}) + (V_2^{(i)} - V_{\text{le}})}$$

In the next step we collect the data with multiple pulses. We denote the distance data set as $\overline{\mathcal{L}^{(n)}}$. For every distance data element $\overline{L^{(i)}} \in \overline{\mathcal{L}^{(n)}}$, the corresponding calculated data from the sensor output are not unique, assume that $N^{(i)}$ data is collected at $\overline{L^{(i)}}$. Consider Equation 3.18, for $\forall j \in N^{(i)}$ we have:

$$\overline{L^{(i)}} = \widetilde{\mathcal{P}}(\mathcal{X}_j^{(i)}) \tag{3.19}$$

Estimating $\widetilde{\mathcal{P}}(\mathcal{X})$ is equivalent to solving the optimization problem:

$$\underset{\widetilde{\mathcal{P}}}{\text{argmin}} \frac{1}{n}\sum_{i}^{n}\frac{\mathcal{K}_i}{N^{(i)}}\sum_{j}^{N^{(i)}}(\overline{L^{(i)}} - \widetilde{\mathcal{P}}(\mathcal{X}_j^{(i)}) - \sigma)^2 \tag{3.20}$$

Where σ has an infinitesimal value that is used to ensure the single match since when we collect the data, different distances can match to the same $\mathcal{X}$ due to the leakage voltage, ambient light conditions. $\mathcal{K}_i$ is the weight vector that is related to the measure distance which could adjust the calibration results for different measure range.

3.3.4 Propagation Delay Caused by System Temperature

There is one more factor that introduces offset to the system measurement. The calibration methods we state above are based on a ideal condition that the temperature remains constant so that the propagation delay from the MOSFET gate driver can be ignored. It is common for MOSFET driver to have the propagation delay increase

$1 \sim 2[ns]$ with respect to the increase of temperature of tens degrees, which will introduce corresponding time offset to the measurement of T_d. To address this problem, we simply assume that the propagation delay is only related to the temperature since the MOSFET supply voltage which is another factor that affects propagation delay is constant. Thus the relationship between the T_d and the system temperature can be regressed by the collected data.

3.4 Software Configuration for Calibration

3.4.1 Sensor Firmware Configuration

Algorithm 1 Framework of Laser Sensor Firmware for Calibration

Hardware Initialization:
CPU Initialization: System Clock, NVIC, DMA and System Watchdog;
Peripheral Initialization: ADC, DAC, High Resolution Timer, Serial Communication and GPIO.
Parameters:
High Resolution Timer channel delay buffer D_n: This buffer determines the accurate time delay among laser pulse, VTX1, VTX2 and VTX3.
Interrupts Vector V_n: each $V_i \in V_n$ represents an interrupt and each interrupt is bound to a callback function.
Pulse Number P: The emission and accumulation amount of the sensor from host computer calibration tool.

1: In the main loop, The CPU processes the laser operation(in 2) and waits for interrupts in this mode, when the ADC callback is triggered, jump to 4, when the serial receive callback is triggered, jump to 3;
2: If $P > 0$, $P = P - 1$, operate one pulse and accumulate logic timing according to delay buffer D_n; if $P = 0$, $P = P - 1$, read the raw data from ADC-DMA and send to host computer through serial-DMA; if $P < 0$, enable serial receive call back;
3: Receive indications from host computer, renew the value of pulse number P, reset the sensor and start a new frame;
4: ADC convert and ADC-DMA transmit completed, raw data are directly transmitted to host computer through serial-DMA;

The laser sensor hardware firmware configurations are documented in 1. For this mode, the firmware can be considered as a hardware translation for the data request from host computer. The sensor system receive the indications from the host computer through serial port of pulses to emit and accumulate and return the raw data of voltages of the two capacitors inside the receive sensor. We collect the

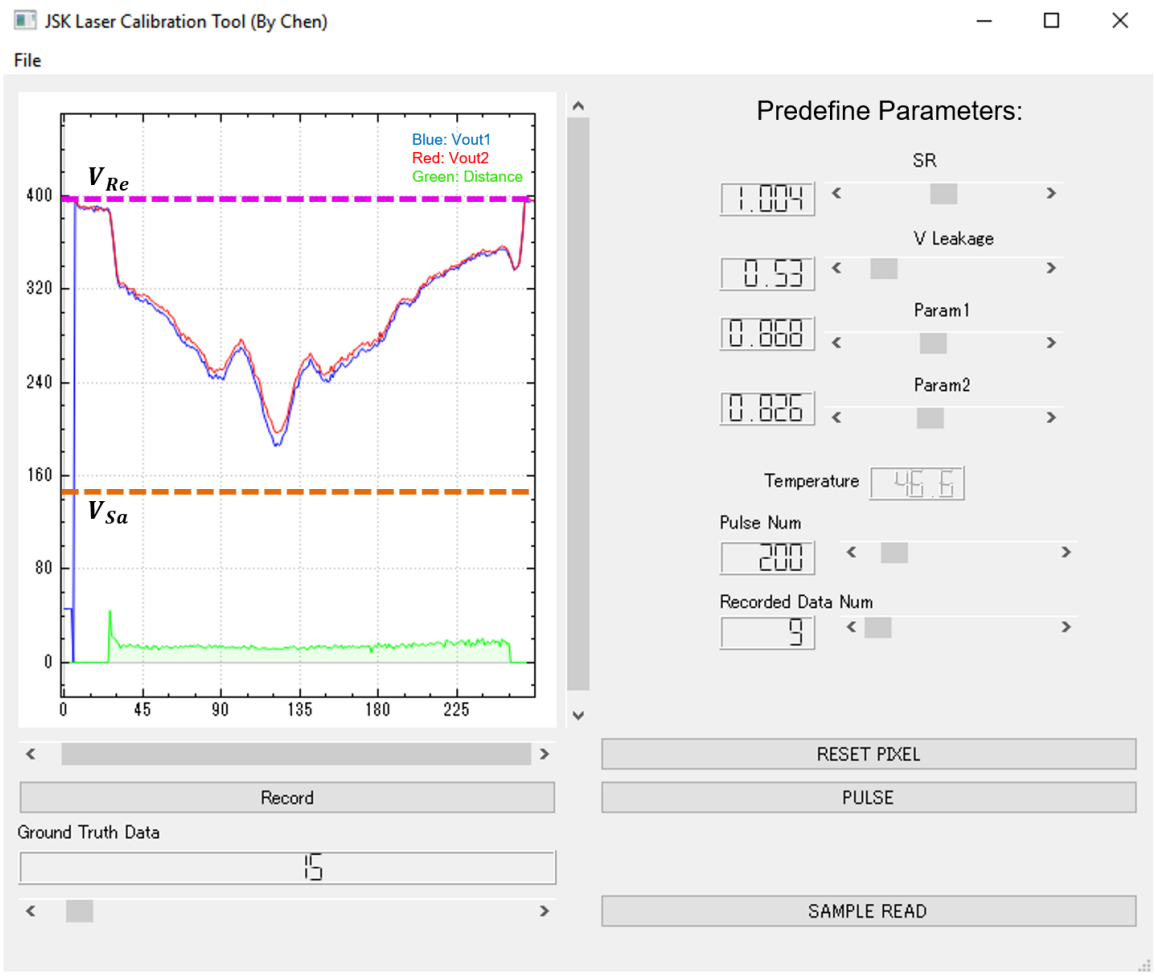


Figure 3.8: Calibration Software Tool(V_{Re} and V_{Sa} are the pixel reset voltage and saturation voltage; Blue and red lines are the voltage raw data for the two capacitors; Green line is the distance data using stored calibrated parameters; To save transmission bandwidth, the first several data pixels of blue line are used to carry temperature data since the first and last 8 pixels are invalid for distance information.)

calibration data using this firmware configuration as well as measure some of the parameters like the voltage leakage rate by simply disable the laser emitter. This mode can not be used in applications since it uses fixed pulses and does not consider the saturation problem.

3.4.2 Laser Sensor Calibration Tool

We developed a calibration tool for our tiny laser sensor. It runs in host computer and communicate with the sensor through USB. The UI configurations are based on Qt opensource library [102]. As Figure 3.8 shows, the left part plots the raw data collected

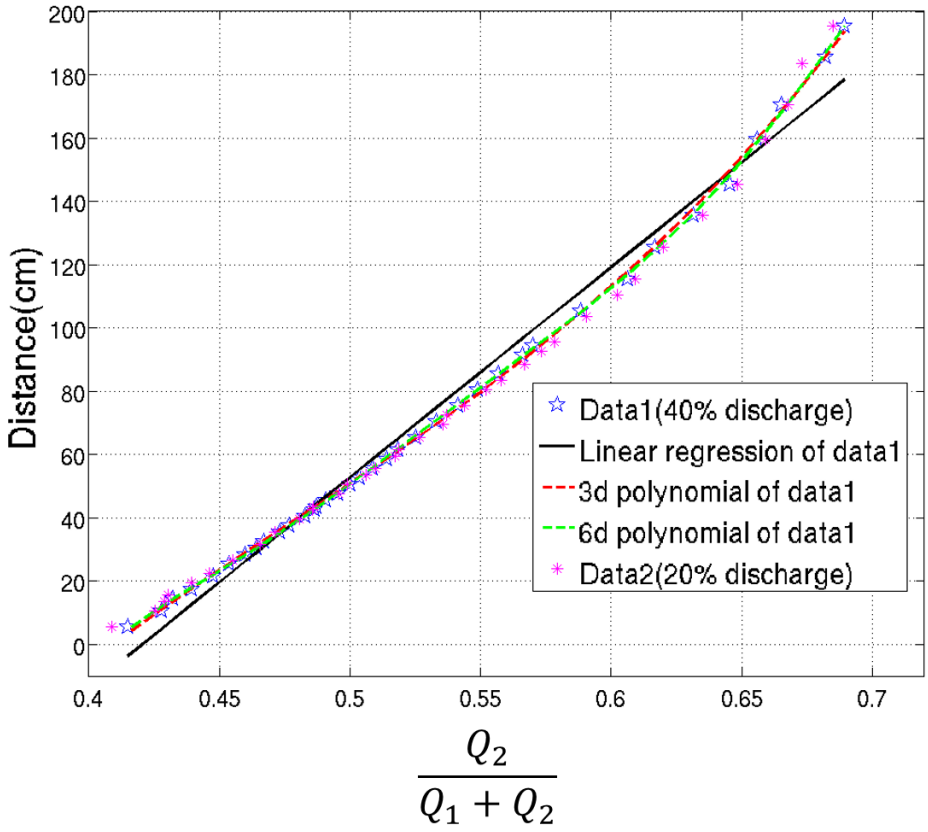


Figure 3.9: Data regression for fixed discharge data: Capacitors discharge $\approx 40\%, 20\%$ for data1 and data2 respectively. (Calibration condition: Center point measurement of white paper; $FOV = 30°$; Data range from $5cm \sim 200cm$, Indoor light condition)

from the sensor and the distance data calculated using the temporary parameters. By simply dragging the scrollbar we can change the parameters to be send to the sensor firmware like the pulse number. The predefined parameters like Sensitive Ratio(SR) and voltage leakage rate are used when calculate the distance data. To collect the calibration data, firstly a ground truth distance data is assigned by dragging the left bottom scrollbar with respect to the distance from sensor to the calibrate surface. The second step is to set the pulse number by dragging the corresponding scrollbar. Changing the pulse number will lead to the change a discharge amount of the two capacitors and results in the change of readout voltage V_{out1} and V_{out2} as red and blue lines show. The V_{Re} and V_{Sa} stand for the pixel reset voltage and saturation voltage of the two capacitors respectively. In the next section we will introduce how we collect the data and perform the calibration.

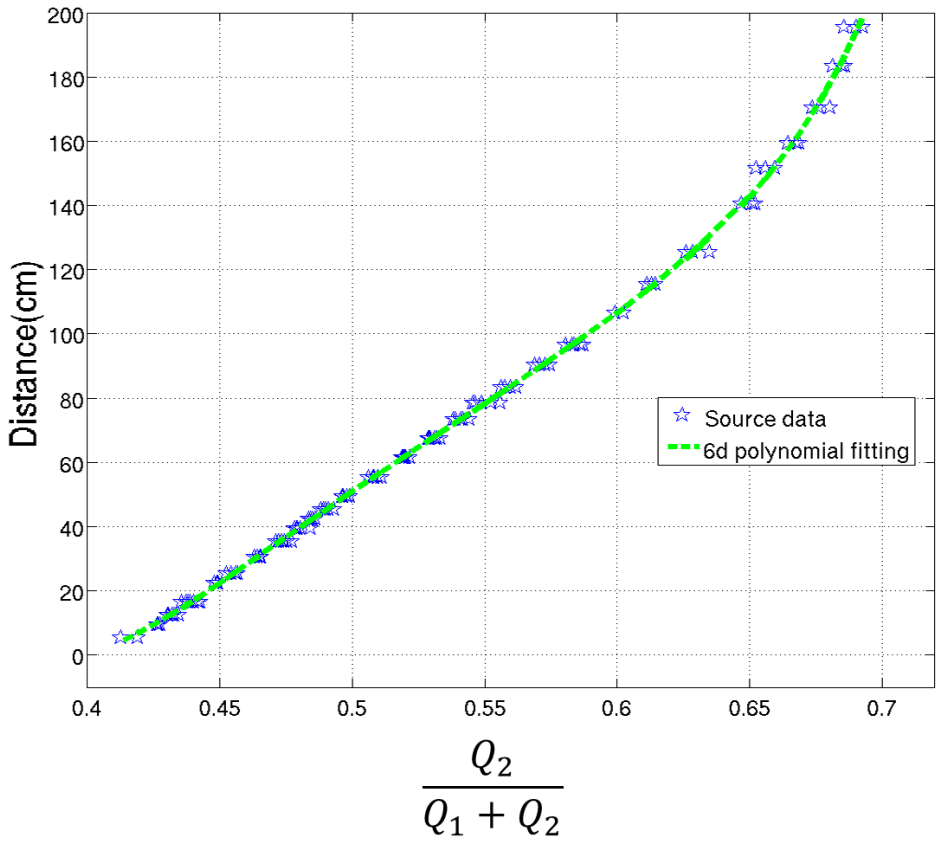


Figure 3.10: 6 degree polynomial regression for various discharge data. (Same condition as Figure 3.9)

3.5 Experimental Results

First we fit the initial model formulated in Equation 3.16 based on the laser output data we collected. As Figure 3.9 shows, using the collected data with 40% discharge level and a range of $5[cm] \sim 200[cm]$, the linear regression models, shown as black lines, are not sufficient for acceptable calibration results, with variance of 29.3814 and standard deviation being more than $5[cm]$. However, polynomial fitting methods with degree of 3 and 6 can achieve $[1.6863, 1.2986]$ and $[0.7424, 0.8616]$ in variance and standard deviation respectively. This is because the discharging amount $Q_1 + Q_2$ is not exactly the same for all the measurements, and the non-constant reflected light which are noted in Eq .3.8 and Equation 3.9 also introduces non-linearity. Considering this, the traditional linear models are abandoned and we choose the polynomial fitting methods to solve the optimization problem in Equation 3.18.

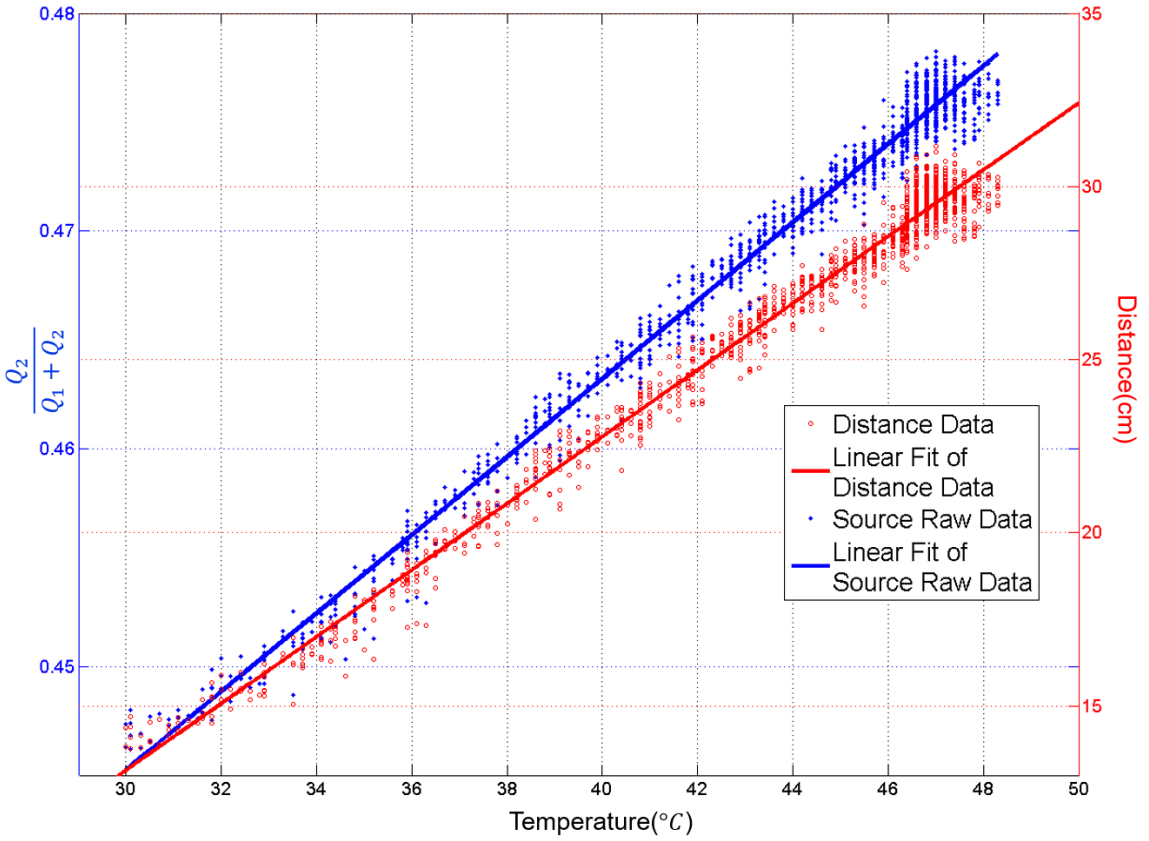


Figure 3.11: Sensor dynamic temperature calibration. (The measurement distance is 30[cm]; The calibration parameters for the distance data of this figure are obtained by saturated temperature, namely around 48[$^\circ C$]; The two data points look similar because the polynomial performs linear around the given domain)

In Figure 3.9 we also evaluate another dataset plotted in purple for comparison, which is collected with capacitor discharge level of 20%. The degree of 6 polynomial fitting error for data 2 is $[1.442, 1.20]$, which is slightly higher than data 1. This is consistent with the suggestions in the datasheet [100] that we should discharge the capacitors for a longer time such that the noise is well accumulated and easier to remove.

The results of multi-discharge data calibration Equation 3.20 are shown in Figure 3.10 with variance and standard deviation being $[3.4618, 1.8606]$. Note that the weighting term $\mathcal{K}$ is temporarily set to an all-one vector such that solving Equation 3.20 is a pure polynomial regression.

The temperature offset is shown in Figure 3.11. We adopt the build-in temperature sensor of our MCU processor and a dissipation silicone is used to connect the

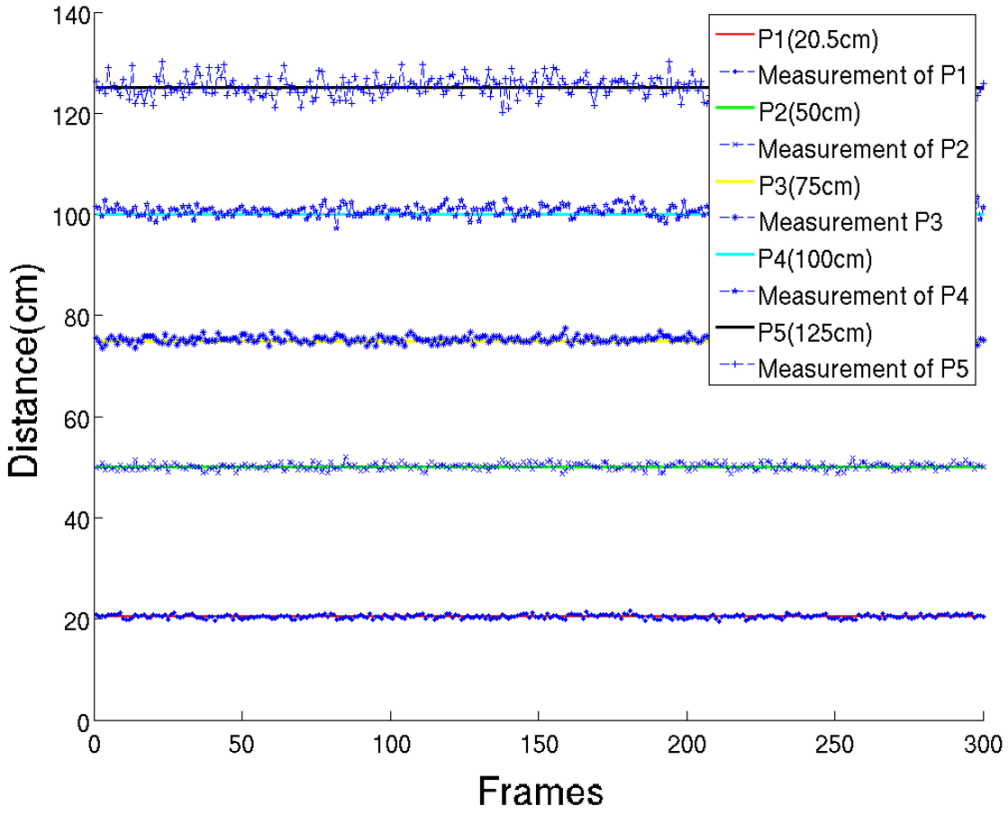


Figure 3.12: Sensor dynamic measurement noise. (Two errors are introduced, the calibration error bias and the system dynamic variance; Same condition as Figure 3.9)

Table 3.1: Measurement noise

Ground Truth	Mean	Var	SD	Bias+SD (cm)
P1($20.5cm$)	20.4628	0.1314	0.3625	-0.0372 ± 0.3625
P2($50cm$)	50.1766	0.3800	0.6164	0.1766 ± 0.6164
P3($75cm$)	75.2637	0.4077	0.6385	0.2637 ± 0.6385
P4($100cm$)	100.6640	1.0982	1.0480	0.6640 ± 1.0480
P5($125cm$)	125.4245	3.8388	1.9593	0.4245 ± 1.9593

MCU to the MOSFET driver so that the temperature is the same. More than 1500 measurement data is collected at indoor environment of normal temperature situation. As the sensor begin to work, the temperature rises and the reaches saturation of about 48[$°C$]. We compared both the ratio of the discharge amount from the two capacitor and the measurement distance with respect to the system temperature. Since the calibration parameters are obtained at saturation temperature, the data distance measurement increases and reaches the groundtruth distance of 30[cm] as temperature increases. We applies linear regression to describe temperature offset because the propagation delay is supposed to the proportion to the temperature as the MOSFET driver datasheet [103] suggested.

Measurement noise is shown in Figure 3.12. We collect the dataset at 5 ground truth target distances and for each distance 300 measurements are recorded. As shown in the figure, with increasing target range, the data waveform becomes rugged. This is because as the measurement range increases, the reflected light becomes weak, and the capacitors inside discharge less than that in a shorter measure range. Under such conditions, small noise from the circuits and ambient light will have a greater effect on the results. Table 3.1 shows the bias and variances for each measurement

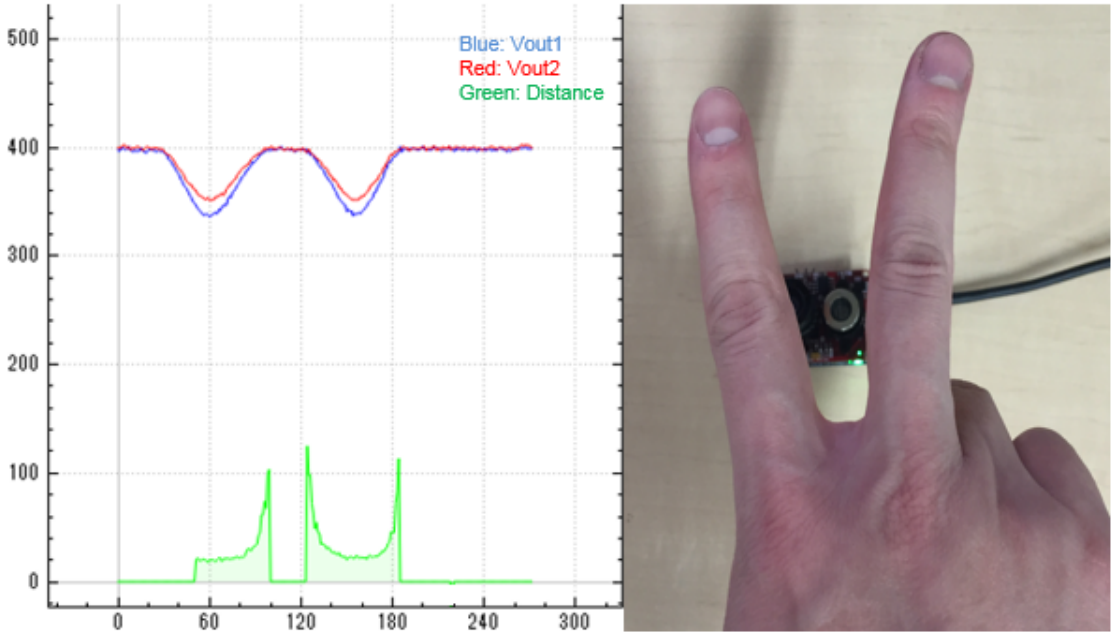


Figure 3.13: Demo of scanning fingers.(Y axis demonstrates the discharge amount of capacitors(red and blue line, unit:$\sim 8[mv]$) and the measurement distance(green line, unit:[cm]); X axis is number of sensor pixels.)

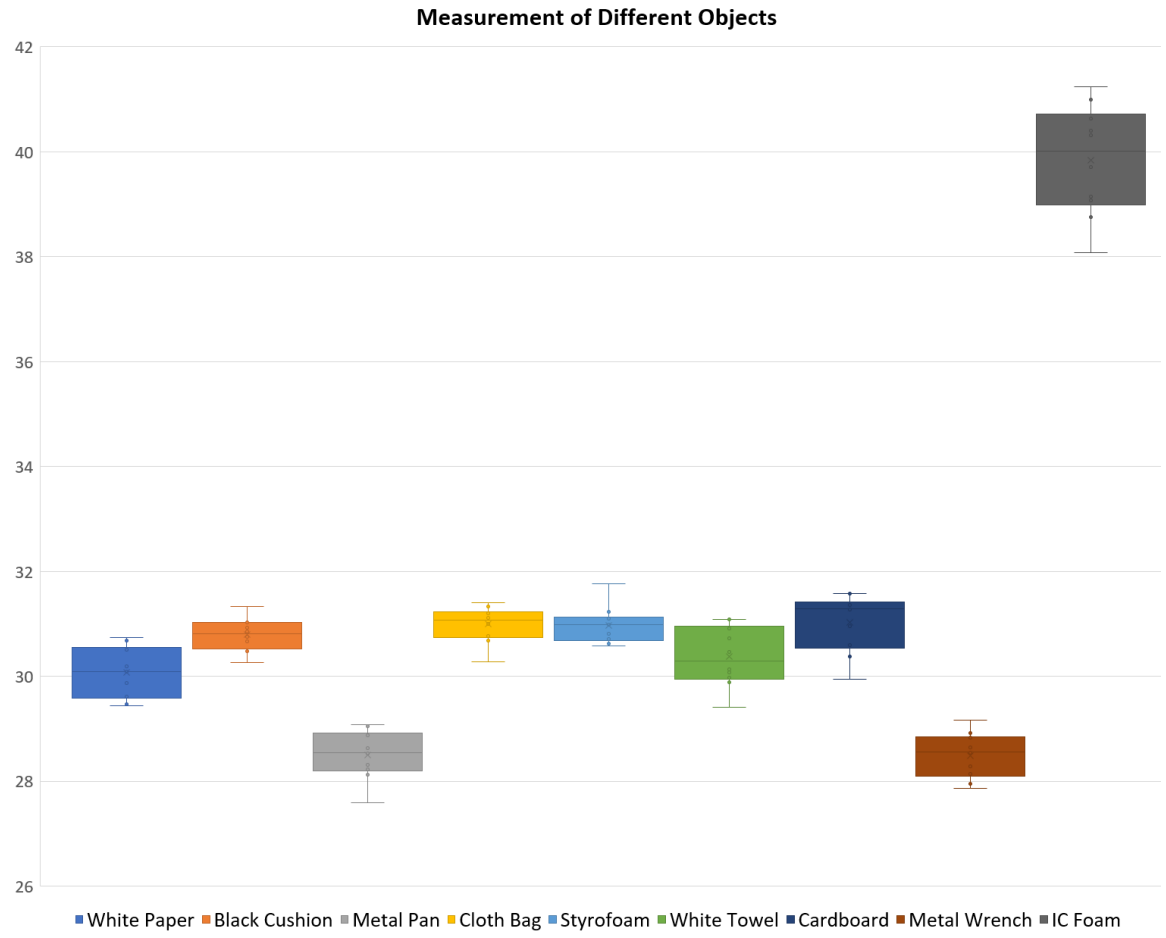


Figure 3.14: Measurement result of different materials with different reflectance. (Calibration condition: $FOV = 30°$; Fixed distance $D = 30cm$; 10 measurements for each object; Indoor light condition)

are reduced to $1[cm]$ after calibration, and the standard deviation increases if longer measurement distance is required.

Figure 3.13 gives an example of the data collected by the laser sensor. The x axis is the pixel coordinate and the y-axis shows the distance (cm) in green and capacitors voltage ADC units (bit) in blue and red.

The experiment result of multiple objects measurement is shown in Figure 3.14. We tested the white paper, black cushion, metal pan without gloss, blue cloth bag, blue Styrofoam, white towel, cardboard, metal wrench and IC foam. The measurement bias, expectation, stand derivation and reflectance rate of each material are listed in Table 3.2. For every kind of material, we perform 10 measurements of a fixed measurement distance $30cm$ under indoor light condition and obtain the mean

Table 3.2: Statistic measurement result of different Materials

Category	Mean	SD	Bias+SD (cm)	Reflectance ($mv/pulse$)
White Paper	30.0705	0.4505	0.0705 ± 0.4505	6.92
Black Cushion	30.7909	0.3001	0.7909 ± 0.3001	4.6955
Metal Pan	28.4962	0.4358	-1.5038 ± 0.4358	25.0071
Cloth Bag	30.9971	0.3220	0.9971 ± 0.3220	5.1638
Styrofoam	30.9759	0.3283	0.9759 ± 0.3283	4.8961
White Towel	30.3708	0.5380	0.3708 ± 0.5380	5.6881
Cardboard	31.0308	0.5229	1.0308 ± 0.5229	4.7981
Metal Wrench	28.4895	0.4062	-1.5105 ± 0.4062	37.125
IC Foam	39.8339	0.9896	9.8339 ± 0.9896	0.5699

distance of the measurements, the standard derivation and the reflectance. The reflectance rate is calculated using the readout data of the discharged capacitor and the pulsed numbers. Since the more pulses the laser emitted, the more the capacitor will discharge, simply divide these two data will obtain the relative reflectance rate at a certain distance. According to the result, since we calibrate the sensor using only white paper which has a constant reflectance, the objects with different material surface result in a bias compared to the white paper. For high reflectance material surface like metal, the proximity measurement range is limited since only several pulses can lead to saturation of the capacitor. On the other hand, objects like IC foam reflect little light and as a consequence, introduces imprecise measurement results.

3.6 Summary

In this section we propose the design of a tiny line laser range sensor. We specify the hardware architecture of the sensor and methods to calibrate the sensor to achieve accurate results. The advantage of our design is that we keep the sensor tiny in size and light in weight without sacrificing measurement accuracy. This is attributed to the high performance circuit modules we use for calibration. We model errors in the sensor system and propose an efficient and intuitive algorithm to calibrate

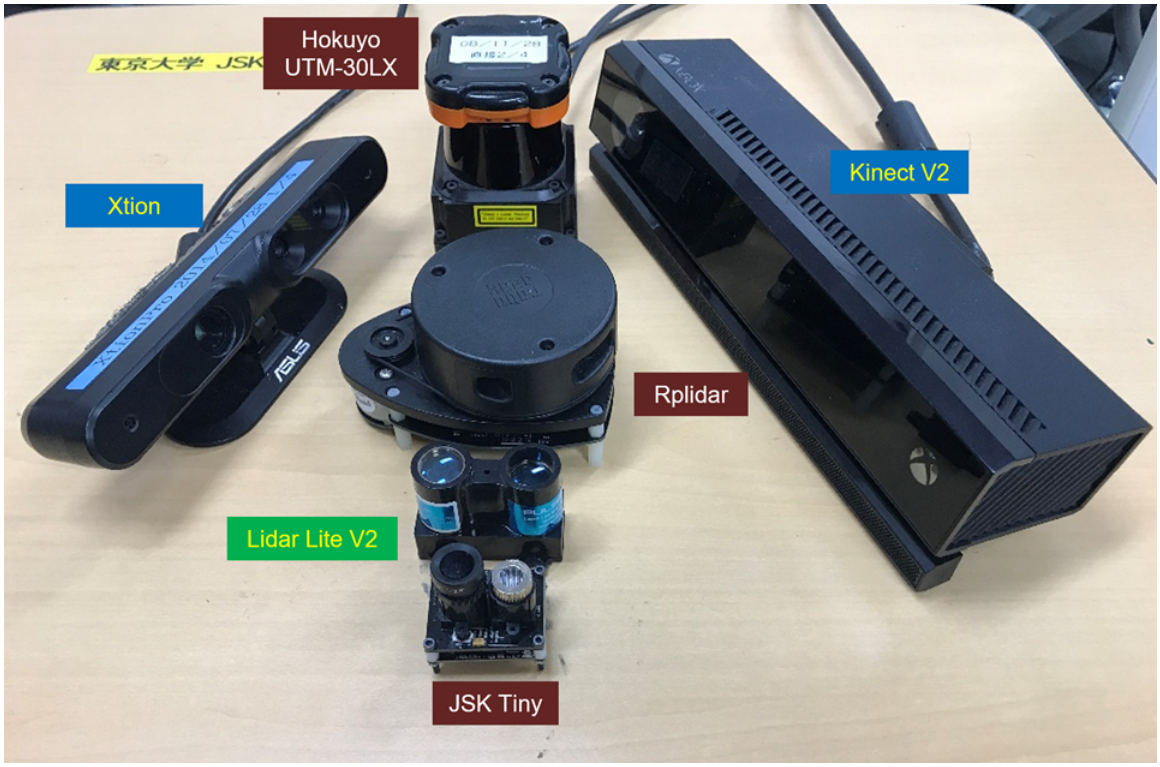


Figure 3.15: Popular Range Sensors in Robotics.

the system. According to the experiments, our sensor achieves measurement biases and repeatable accuracy of less than 2[cm], which is acceptable for a lot of range sensing applications. The comparison of popular robotic range sensors are shown in Figure 3.15 and Table 3.3. Our sensor system provides 2D range data and achieves an acceptable accuracy and precision in its given measurable range and certain light condition with a very small size and weight. The small size, low cost, and light weight make this unconventional sensor a valuable tools for robotic applications that have strong requirements on sensor size and cost, such as UAV safe landing, tiny robot range sensing, and robot hand-eye configuration as well as tilt 3D scan as Figure 3.16 3.17 shows. In the later chapter we will shown how our sensor contributes to the a task-oriented robot system as a local sensing sensor.

This tiny laser line sensor system has several limitations respect to its hardware design, measure principle and calibration approaches. Firstly, since we mount the laser emitter and receive sensor collinear, the base-line and light diffraction effect occur at the edge of the measure object. This effect will lead inaccurate range data at the edge region. A filter can be applied to remove the these inaccurate region while brings the data loss. Secondly, it can not handle the object with high reflectance

Table 3.3: Comparison of Popular Range Sensors. (Some of the data are from [73][104] and the accuracy and precision and under certain condition)

Categories	Principle	Data Type	Size[mm]	Weight (with cable)	Accuracy vs Precision	Cost
Hokuyo UTM-30LX	ToF	2D Line Scan	$87 \times 60 \times 60$	$\approx 220g$	$< 30mm$; $< 10mm$	>$3000
Asus Xtion	Structured Light	3D Depth Image	$180 \times 50 \times 30$	$\approx 220g$	$\approx 2\%$; $< 1mm$	$\approx$ $200
Microsoft Kinect V2	ToF	3D Depth Image	$249 \times 67 \times 66$	$> 1000g$	$< 20mm$; $< 1.5mm$	$\approx$ $200
Rplidar	Triangulat-ion	2D Line Scan	$98.5 \times 60 \times 55$	$\approx 180g$	$\approx 2\%$; –	$\approx$ $400
Lidar-Lite V3	ToF	1D Point	$48 \times 40 \times 20$	$\approx 25g$	$< 25mm$; –	$\approx$ $200
JSK Tiny	ToF	2D Line Image	$37 \times 35 \times 30$	$\approx 25g$	$\approx 1\%$; $< 20mm$	$\approx$ $150

surface because the range data becomes invalid when the corresponding pixel reach saturation. Reducing the laser emission power can improve this problem but at the same time limited the maximum measurable range. Thirdly, when multiple sensors are used at the same time, the range data in the overlapping region is not correct without synchronization. Lastly, as Table 3.2 shows, since we use white paper to do the calibration, the range data accuracy differs from the reflectance of the object surface. Although in a longer range the dynamic noise is larger than this bias offset, it is theoretical possible to consider the object surface reflectance in the calibration in the future work.

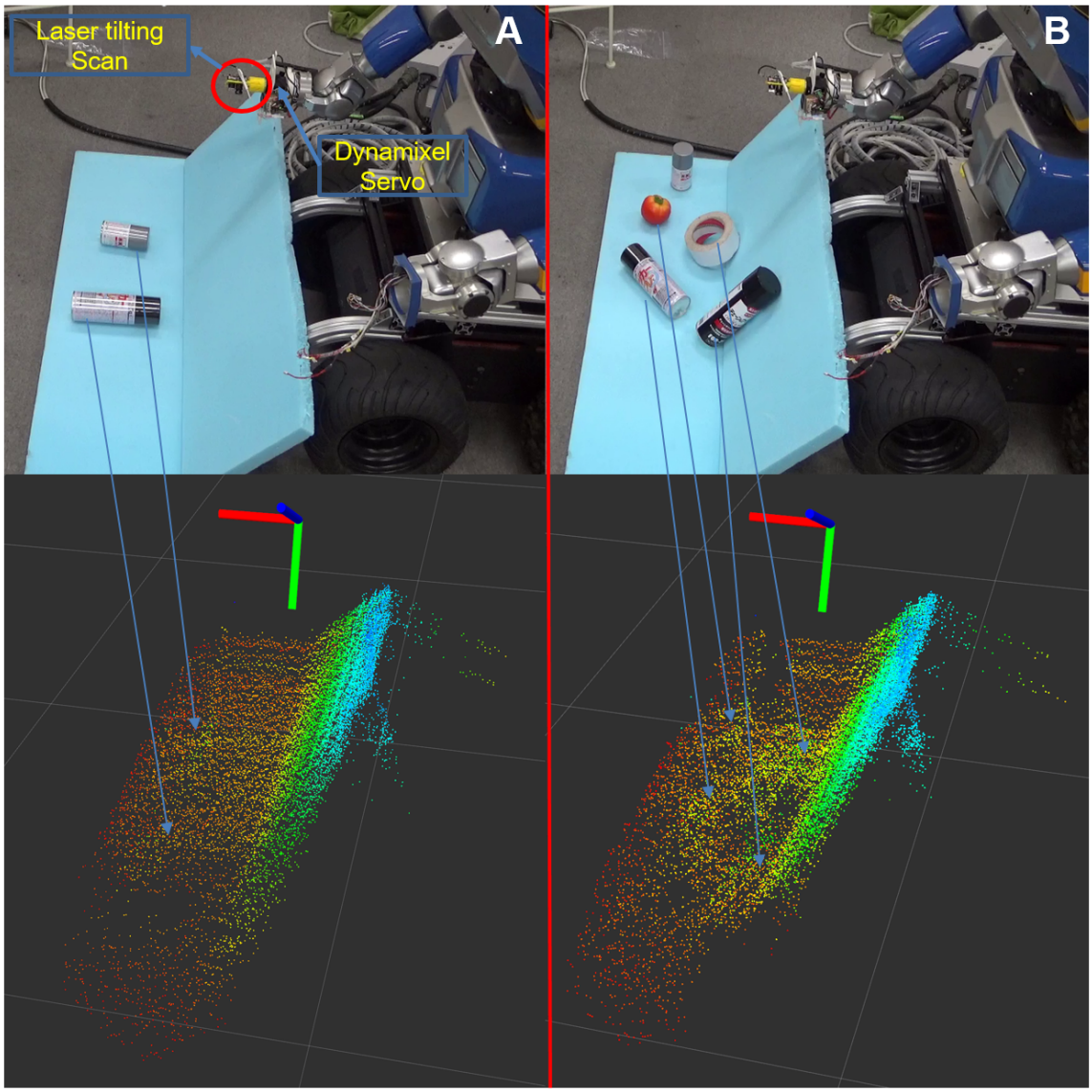


Figure 3.16: Tilt laser scan of 90°. (A dynamixel servo is applied for tilting; Every frame contains 256 pixels of distance data and 50 frames are accumulated to get the pointcloud; The different color indicates the different distances toward the laser; Grid size is $1[m]$.)

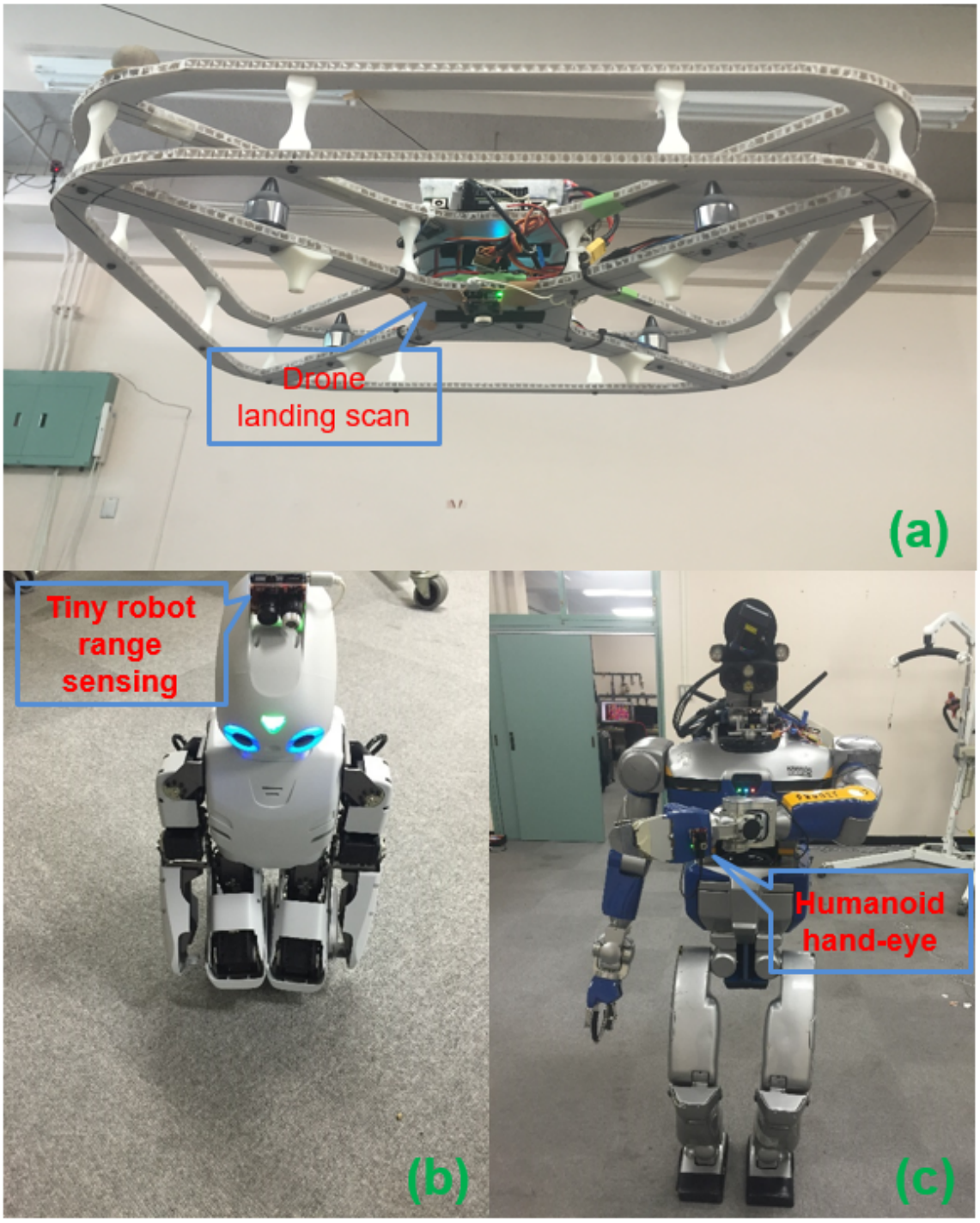


Figure 3.17: Applications: (a) Ground scanning for Drone, providing line distance data. (b) Feasible range sensor for tiny robot Darwin. (c) Range sensing hand-eye for humanoid robot HRP2.

Chapter 4

Active Local Verification with Reasoning-Based Vision Recognition in Robot Tomato Harvesting

4.1 Robot In Agriculture

Agricultural industrialization calls for autonomous and intelligent robot systems for improving the harvesting efficiency with accuracy. This kind of systems will help farm factories overcome the dilemma between human labour costs and product quality problems. Traditional mass harvesting solutions, which apply trunk shakers or combine harvesters, have been elaborated in recent decades[105][106]. But as the use of almost all of these mechanized harvesting apparatus leads to impact and damage to the harvested crops, and is not suitable for soft and delicate crops like tomatoes, strawberries and apples. Therefore, for application on crops like these, the use of selective harvesting solutions, where crops are picked one-by-one carefully like a human worker, is preferred.

The difficulties of selective harvesting ascribe to the requests of not only detecting where the crop is, but also estimating the posture of the crop and the pedicel to cut. In the real world environment, crops like tomatoes are densely gathered and occluded by branches, which make it impossible to directly detect the pedicel. Thus in our work we gain inspiration from the fact that, even when humans cannot directly

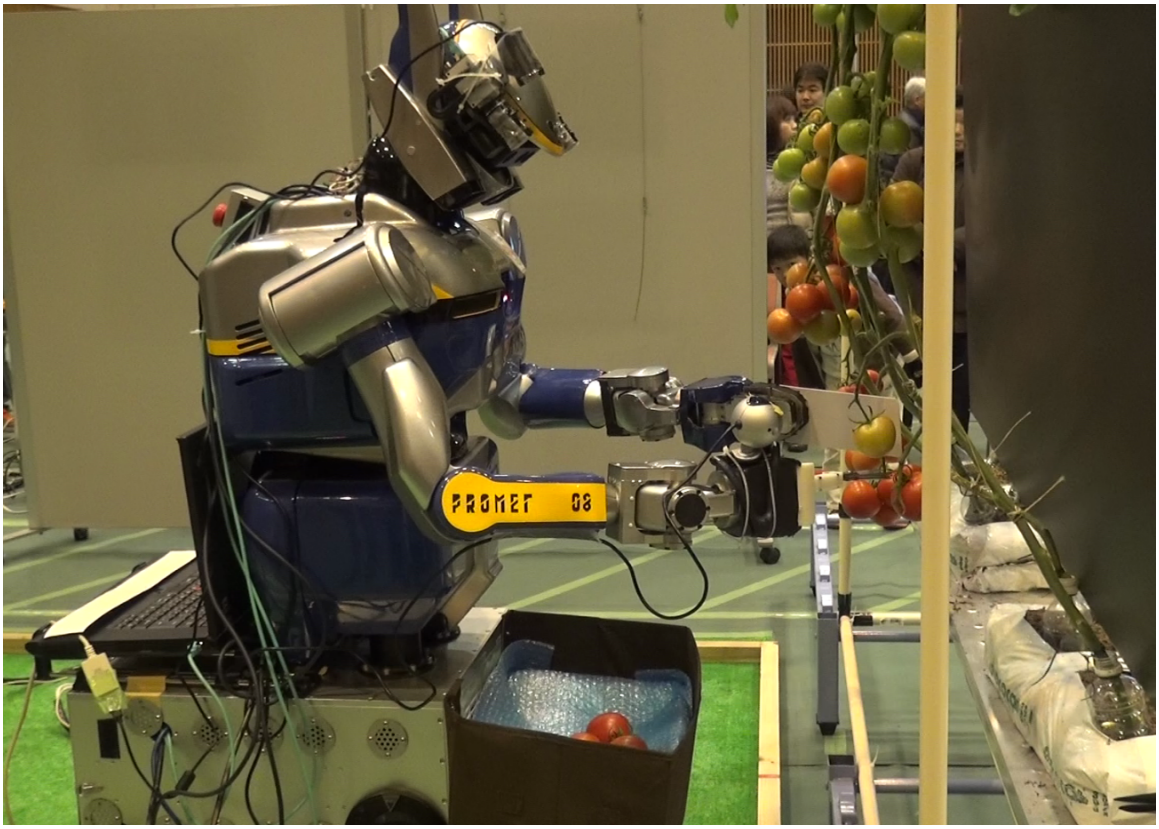


Figure 4.1: Participation in tomato harvesting robot competition 2014. 2

observe the pedicel, we can guess where the pedicel is from what we see according to the understanding of scene and laws of physics.

We make an effort to combine reasoning method with vision cognition for scene understanding. We propose a novel approach for extracting geometric models of tomatoes and estimating the pedicel direction of each tomato by physical reasoning, based on observation and knowledge of the laws of physics governing the crops.

Our system consists of two main parts:

1) Agriculture-Support Humanoid Robot: As shown in Figure 4.1, we use the upper body of HRP2(Humanoid Robotics Project 2) humanoid robot and a VMAX omni-direction mobile moving base as our robot platform. The robot has 7 DOF in each hand and 2 DOF in head. Two RGB-D sensors are installed on the head and hand of the robot to obtain better observations. Special scissors that can both cut and hold the picked crop by its pedicel at the same time are installed on

the robot's grippers. We use this platform to participate in the Tomato Harversting Robot Competition 2014 and won the champion[2] .

2) Reasoning based Vision Cognition: In this work, we mainly focus on handling crops like tomatoes which tend to be gathered together in one branch, making it difficult to directly detect the pedicel for each individual crop. First, we apply HSI color filter along with Euclidean cluster to the pointcloud and choose the closest and feasible branch for picking. Second, the robot moves the hand RGBD camera to obtain a better observation of the crop's primitive model (in this case, a 2.5D semi-sphere pointcloud). Then, according to the geometric reasoning assumption of each tomato, a ransac based sphere segmentation algorithm was performed to get the 3D location and approximate radius of each primitive. After that, these primitives are filtered and grouped into physically stable objects according to gravity and interaction forces between contacting primitives. By assigning magnitudes to the interaction forces, we can deduce the direction of each tomato's pedicel and generate a feasible path for the hand to harvest the tomato.

This section is related to two research fields in the literature.

4.1.1 Agriculture Harvesting Automation State of Art

Previous research studies on selective crop harvesting has made huge efforts on not only crop detection, but also design of the end-effector and gripper. Systematic work by Baeten.J *et al.*[107] proposed an autonomous apple harvester with a carefully designed gripper and an image based vision detection system. Their robot can pick apples on trees at a relatively high successful rate, but since the pose and the stem of the apples are not considered, the gripper's picking movement is sometimes not predictable. Another impressing work proposed by Shigehiko *et al.*[108] introduced a strawberry harvesting robot equipped with five light sources, which consisted of 120 light-emitting diodes (LED) each. In addition, three well aligned CCD cameras are used to obtain the 3D position of the strawberry and peduncle location. With a gripper that could cut and hold the peduncle at the same time, their harvesting robot could achieve a successful picking rate of greater than 79%. There have also been interesting works for harvesting clustered tomatoes based on binocular stereo vision[109]. [110][111] introduced gripper designs and their manufacturing methods. For example, [111] analyzed their scissors-like grippers that could be used to both cut and hold the pedicel of tomatoes, which is also equipped by our humanoid robot.

[2]Tomato Harvesting Robot Competition Website: `http://www.lsse.kyutech.ac.jp/~sociorobo/tomato-robot`

In comparison to the above, our system is based on a humanoid robot which application is robust and could be improved easily. Picking crops that gather together in branches is a difficult task and our reasoning-based vision approach can provide a distribution of estimated pedicel directions.

4.1.2 Vision Meets Cognition

Tomatoes trees possess a natural geometry structure that several tomatoes gather together as a branch, which makes it difficult for vision detection and selective picking. However, a physical reasoning and geometrical reasoning based vision framework [86][87] can help solving this difficult problem.

The focus of our reasoning method is on branches where tomatoes are densely gathered, and we attempt to reconstruct the geometrical and physical relationships of each crop within the branch using the primitive model according to a simple but powerful fact that every model should comply with the laws of physics. Thus we create a probabilistic model to describe the estimation of the pedicel direction. The results could be used both for directly picking and provide initial guesses.

4.2 Harvesting Humanoid System

In this section, we introduce our humanoid platform including the robot specifications, grippers and the calibration of sensors.

4.2.1 Robot Platform

Humanoid robots are designed to carry out a wide range of tasks. Recently, impressive research studies have shown demonstrations such as cooking, cleaning, folding clothes, handing over objects smoothly and carrying heavy objects[112, 113, 114, 115, 116], namely, human daily life supporting tasks. In this work, we extend the application of humanoid robot to agriculture supporting tasks.

Table 4.1 and Figure 4.2 show the basic specification of our robot HRP2W. Compared to the original HRP2 humanoid robot which is equipped legs, using a VMAX as the moving base is much more convenient and power saving in this application. In addition, VMAX can be easily altered to cooperate with the popular rail-like system that is widely used in green house factories.

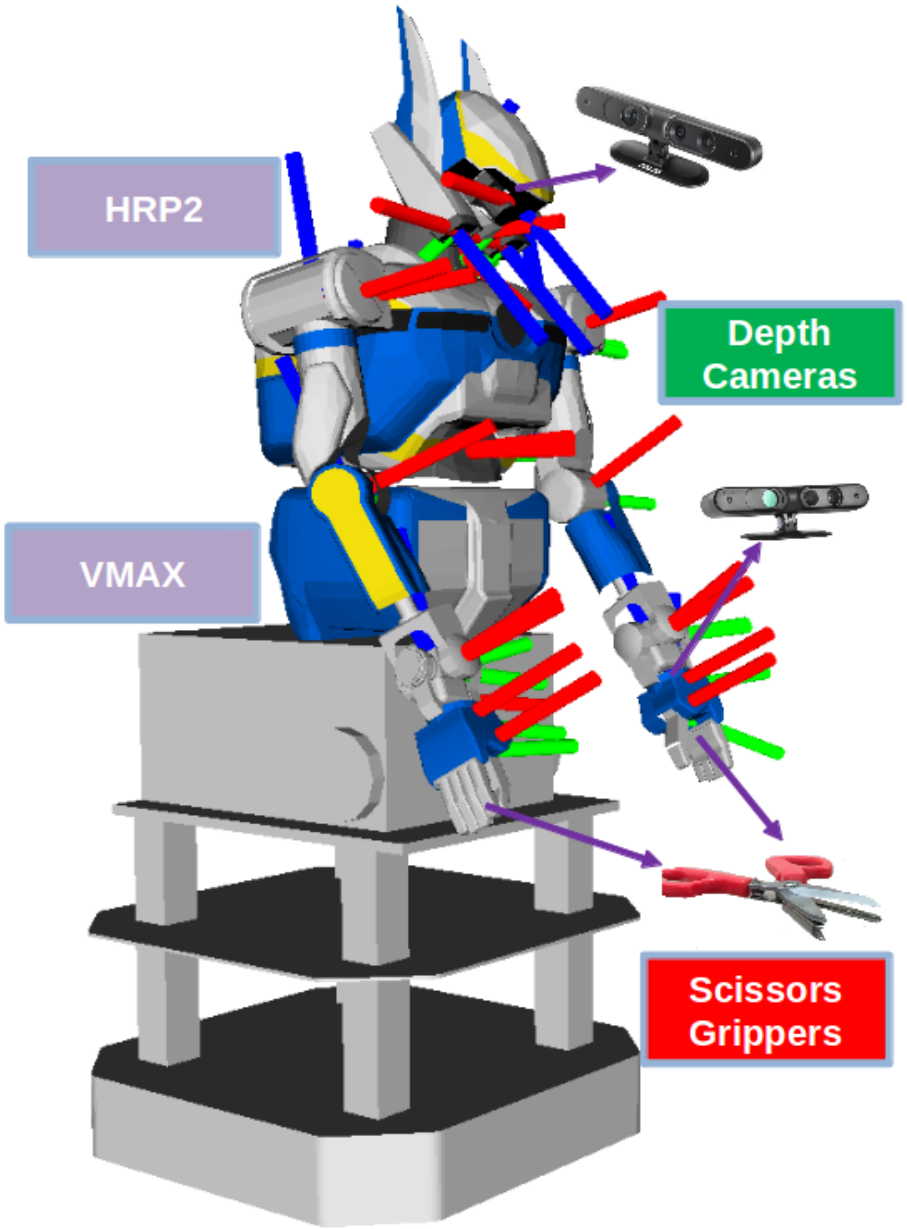


Figure 4.2: Humanoid HRP2W. (The axis demonstrate ROS TF(transform) of each joint and camera of the hrp2 upper body)

4.2.2 Harvesting Gripper Evaluation

We equipped our robot with a special scissors with two layers of blade, one for cutting and one for holding. As shown in Figure 4.3, we designed attachments between end-effector(robot finger) and scissors, and according to [111], the average pedicel cutting force is about $137.27N$. Despite of species differences, we set this as the reference, $|F_{ref}|$. Consider the geometric relations of a pair of scissors, the effort arm length and force are given by L^e, F^e, and the resistance arm length and force are denote as L^r, F^r. A simplified lever model could be applied to the scissors:

$$F^e L^e = F^r L^r \tag{4.1}$$

Assuming that the average pedicel diameter is approximately $3mm$, we experimentally determined that our mechanism can cut if L^r is within the range of

$$L^r \in [2.2, 5.1](Unit : cm)$$

Within this range of L^r, L^e slides within the range of

$$L^e \in [7.1, 9.9](Unit : cm)$$

Given the minimum amount of required cutting force F_{ref}, assuming a linear model between L^e and L^r, by rearranging Equation 4.1 we obtain:

$$F^e \geq \frac{F_{ref} L^r}{-0.9655 L^r + 12.0241} \tag{4.2}$$

Since it could be easily proven that $d\frac{F^e}{dL^r} > 0$ under the domain, $F^e \geq 98.6\ N$ is required. And the maximum finger joint torque of HRP2 robot is about $13.9\ N \cdot M$.

Table 4.1: Specification of HRP2W

Name	DOF	Sensors	Global Property	
HEAD	2	Xtion	DOF	24
ARM	8×2	Carmine	$Length \times Width \times Height$	$65cm \times 65cm \times 165cm$
CHEST	2	None		
VMAX	4	Hokuyo Laser	Weight	$150kg$

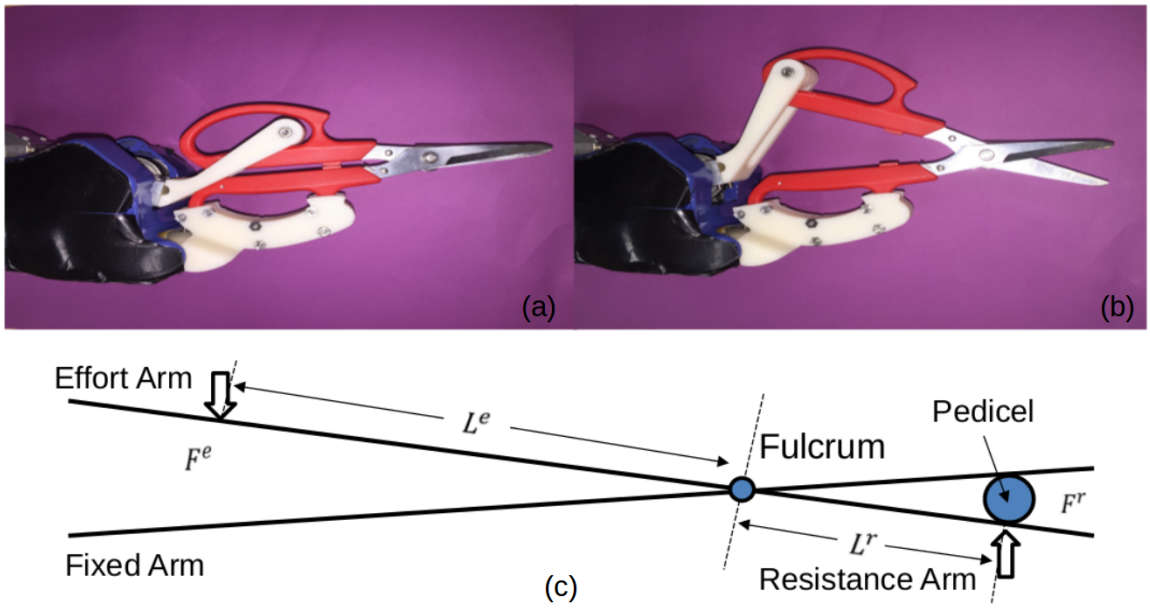


Figure 4.3: Illustration of gripper model. (a): Gripper closed, (b): Gripper opened, (c): Gripper lever model

By designing an attachment with distance to the joint about 8.5cm, which provide correspond effort force 163.53 N, our gripper is capable of cutting a general toughness pedicel.

However, according to our tests, we still obtained approximately 10% failed attempts. This may be attributed to the cutting angle, the smoothness of the blade and also partial cutting of the pedicel. As claimed in [111], depending on the cutting location and angle, the maximum pedicel cutting force sometimes could reach 245 N, which is beyond the limit of the robot's finger joint. We solve this by changing the cutting strategy and repeatedly making attempts.

4.2.3 Installation of Hand-Camera

Our robot is equipped with two RGBD sensors: Xtion and Carmine, which are fixed at the head and the hand respectively. A hand-camera is crucial for detection when tomatoes are densely gathered in branches with occlusions. In addition, hand-camera is more flexible since the view could be changed at any time. However, in terms of pointcloud data accuracy, both camera depth calibration and hand-camera to head-camera calibration is indispensable.

Depth Camera Calibration

Although the devices are calibrated during manufacturing by their own algorithm and are adequate for casual use, for our application, a more accurate depth calibration considering our application range is preferred. [117] summarised depth camera calibration methods considering depth distortion. Suppose in disparity space, let d be the distorted raw disparity data from camera sensors. The calibrated disparity is given by:

$$d_c = d + D_\sigma(u, v) \cdot \exp(a_0 - a_1 d) \tag{4.3}$$

where $\exp(a_0 - a_1 d)$ is the scale weight describing distortion intensity. For our application, since we are not going to cover the whole visible range of our camera, the distortion intensity could be simplified to a linear model and independent of $D_\sigma(u, v)$. We calibrate the camera in z depth space with a simplified model:

$$z_c = z + Z_\sigma(u, v) + \theta(z) \tag{4.4}$$

where $\theta'(z) = C$ and C is a constant. We detect chessboard corners using calibrated RGB camera data and used these points as the ground truth. By acquiring enough points throughout our application range, in this case between $0.8 \sim 1.5m$ for Xtion and $0.35 \sim 0.6m$ for Carmine, a quadratic model is fitted to $Z_\sigma(u, v)$ and linear model to $\theta(z)$.

Calibration of Hand-Camera

To calibrate the hand-camera to the head-camera, for every timestamp we need to calculate a rigid transformation, denote as $\widetilde{T_r}$, with:

$$x_r = \widetilde{T_r} x = \widetilde{R_r} x + \widetilde{t_r} \tag{4.5}$$

where $\widetilde{R_r}$ is the rotation and $\widetilde{t_r}$ is the translation. Since the joint angle vector could help solving the transformation between head-camera to the end joint where the hand-camera was installed, $\widetilde{T_r}$ consists of two transformations:

$$\widetilde{T_r} = \widetilde{T_j} \cdot \overline{T_c} \tag{4.6}$$

and $\widetilde{T_j}$ could be acquired from the robot's mechanical parameters. Only the constant transformation $\overline{T_c}$ needs to be calculated.

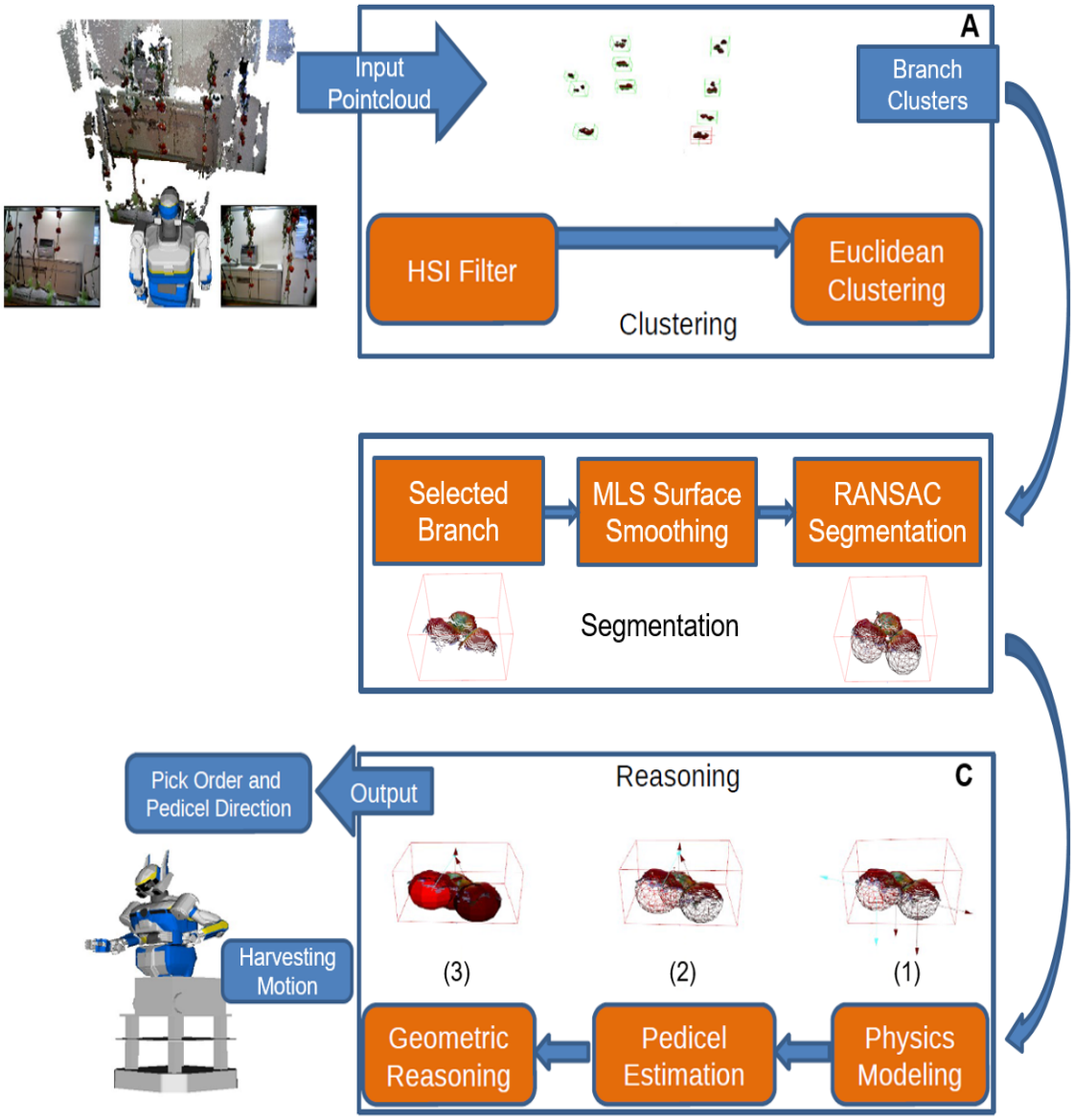


Figure 4.4: Vision algorithm flow diagram. (A) Preprocessing of pointcloud, red bounding box is the selected (closest) branch. (B) Tomato model segmentation, acquiring primitive models. (C) Physical and geometric reasoning, force model, pedicel direction estimation and selecting which to pick. (Arrows in C-(1) demonstrate the direction of gravity, interaction forces and in C-(2,3) indicate the pedicel vector)

We first make an initial guess of $\overline{T_c}$ and $\widetilde{T_r}$ manually, noted as $\overline{T_c^*}$ and $\widetilde{T_r^*}$ respectively for the hand-camera joint and the head. For points observed by the hand-camera $x_i \in \mathcal{P}_{hand}$, we define the pointcloud after applying the transformation $\widetilde{T_r^*}$ to be

$$\mathcal{P}_{hand}^{\widetilde{T_r^*}} = \{x_j | x_j = \widetilde{T_r^*} x_i\} \tag{4.7}$$

and for the pointcloud observed by the head-camera, $\mathcal{P}_{head}$, we could obtain $\overline{T_c}$ using the registration method:

$$\underset{\overline{T_c^*}}{\operatorname{argmin}} \, Dist(\mathcal{P}_{head}, \mathcal{P}_{hand}^{\widetilde{T_r^*}}) \tag{4.8}$$

Simple ICP algorithm would lead to a good result for acquiring the transformation $\overline{T_c}$.

4.3 Reasoning based Vision Cognition

This section discusses reasoning-based vision recognition approach for detecting the tomatoes on a branch and estimating the pose of each tomato from the pointcloud data. This section elaborates three parts, tomato branch clustering, model segmentation using Ransac and physics reasoning for pedicel. Figure 4.4 demonstrates the pipeline of our approach.

4.3.1 Tomato Branch Clustering

Like human tomato harvesting, our algorithm first extracts and chooses one branch to focus on, and then picks tomatoes one by one. In this part, an HSI filter is first applied onto the raw pointcloud data to obtain tomato candidate points. Then we use a euclidean clustering algorithm to group points and their neighbours into one cluster. The robot will choose the feasible and closest branch to pick.

We choose the HSI parameters $-15° \leq H \leq 15°, 100 \leq S \leq 250, 25 \leq I \leq 180$. Admittedly, the change of lighting condition will affect the result of HSI filters with offline parameters. However, our application focuses on indoor lighting condition where the ambient light is constant, corresponding to harvesting in greenhouse in the evening without natural light.

Next we apply euclidean clustering to the filtered pointcloud. Euclidean clustering technique can organize points into clusters with respect to the distance feature. For

$\forall p_i, p_j \in P_{hsi}$, clusters $O_i = \{p_i \in P_i\}$ and $O_j = \{p_j \in P_j\}$ are obtained by:

$$min||p_i - p_j|| \geq d_{threhold} \tag{4.9}$$

which means points are set to different clusters if the minimum distance is above $d_{threhold}$. This method is effective if points are spatially continuous and there does not exist gaps in a cluster. We set $d_{threhold} = 6\ cm$ and it works well for tomato branch clustering, as shown in Figure 4.4.A.

4.3.2 Ransac-based Primitive Model Segmentation

Primitive shape of tomatoes, apples, oranges and cherries can be described as a sphere in 3D. Generally speaking, our work is based on this assumption. To estimate a sphere, two popular methods are introduced. An $n - D$ sphere $\mathcal{S}^n$ is defined as a sphere in n dimension space with points satisfy the following equation:

$$||(p - c)^T(p - c)|| = r^2 \tag{4.10}$$

where c is the center of the sphere and r is the corresponding radius. For:

$$\forall p_{i,j} \in (\mathcal{S}^n \cap \mathbb{R}^n)$$

we have:

$$||(p_i - c)^T(p_i - c)|| = ||(p_j - c)^T(p_j - c)|| = r^2 \tag{4.11}$$

Since every two distinct points provide one linear equation of $n + 1$ unknowns, it requires $n + 1$ points to obtain the solution of the sphere model. In our case, $n = 3$, which means at least 4 distinct points are necessary, and all the points should not be aligned in the same plane.[118].

The other method is introduced in [119]. This method uses only 2 points for the calculation of a sphere model in $\mathbb{R}^3$ space, but need to acquire the normal of each point. Denote two distinct points $p_1, p_2 \in \mathbb{R}^3$ with corresponding normals $\vec{n_1}, \vec{n_2}$. The line segment between two normals is defined as $\mathbb{L}$. We can obtain the shortest line segment $\mathbb{L}_{min}$ between normals $\vec{n_1}, \vec{n_2}$:

$$\underset{\mathbb{L}}{\operatorname{argmin}}\, Dist(\vec{n_1}, \vec{n_2}) \tag{4.12}$$

then the center c of the sphere is simply set to be the middle point of shortest line segment $\mathbb{L}_{min}$, and the radius of the sphere is defined as $\frac{(p_1-c)+(p_2-c)}{2}$.

For the effectiveness of sampling based fitting algorithms, we consider Pointcloud $\mathcal{P}$ of size N within a primitive model Ψ consist of n points. If any k points can lead to a model candidate ψ, the probability from one sampling procedure that $\psi \subseteq \Psi$:

$$P(k) = \binom{n}{k} / \binom{N}{k} \tag{4.13}$$

consider that $(n \gg k)$, $P(k)$ could be note as:

$$P(k) = \frac{n!(N-k)!}{N!(n-k)!} \approx (\frac{n}{N})^k \tag{4.14}$$

Thus, a successful model estimation probability after s times sampling failures is given as:

$$P(k, s) = 1 - (1 - P(k))^s \tag{4.15}$$

The first method is intuitive and general, but sensitive to noise. The second method's robustness heavily depends on the normal estimation results. There are no big difference in effectiveness since the estimation of normal also cost time. Pointclouds that are acquired by RGBD camera can achieve a certain accuracy in certain scale, but since the measurement method is based on structured light theory, the detail of the object like textures and edges would sometimes introduce noises. Therefore, it is necessary to smooth the surface of the pointcloud. Considering this, we take advantage of the Moving Least Squares (MLS) [120] surface reconstruction method for smoothing pointclouds with noise. MLS tries to recreate the missing parts of the surface by higher order polynomial interpolations between the surrounding data points and eliminate sharp convex and concave parts on the surface.

We obtain all the sphere models of tomatoes on the branch that satisfy the radius constraint by iteratively sampling. Once the algorithm finds a candidate with acceptable inlier points, the model will be stored and the inlier points would be removed from the pointcloud, note that the semi-sphere models with sphere plane farther than the centroid of the sphere will not be regard as a model in this process since it is obvious the camera could only see the front half of the sphere. The algorithm stops until the remaining number of points are small enough. Figure 4.4.B briefly demonstrate segmentation procedure.

4.3.3 Physical and Geometric Reasoning

Unlike previous work [86] which was based on volumetric primitives, our reasoning algorithm is designed to deal with higher level reasoning. Given the segmented primitive sphere models which represent each tomato, denoted as $\Theta(c, r)$, where c is the sphere center and r is the corresponding radius. For a branch where n spheres are detected, we denote the branch:

$$\mathcal{B} = \bigcup_{i}^{n} \Theta(c_i, r_i) \tag{4.16}$$

where $i \in \mathcal{N}, i \in [1, n]$. Then we assign a connected graph to $\mathcal{B}$, noted as $G_{\mathcal{B}}$. Taking every sphere $\Theta(c, r)$ as a node, and in terms of connectivity, any two distinct nodes should satisfy:

$$\forall \Theta(c_i, r_i), \Theta(c_j, r_j) \in \mathcal{B}, (i \neq j)$$
$$| \; ||c_i - c_j|| - (r_i + r_j) \; | \leq d_e \tag{4.17}$$

where $||c_i - c_j||$ is the euclidean distance between two sphere centroids. An error of d_e is imposed because tomatoes are not perfect spheres and there may also be errors from the sensors and segmentation results.

Then we assign a physical correlation to each edge of $G_{\mathcal{B}}$. Accordingly, two attached tomatoes should provide interaction force to each other which are equal and opposite according to Newton's Third Law:

$$\Sigma F_{\Theta(c_i,r_i),\Theta(c_j,r_j)} = -\Sigma F_{\Theta(c_j,r_j),\Theta(c_i,r_i)} \tag{4.18}$$

The direction of the force could be simply calculated by the center of two models and denoted as $\vec{e}_{c_i c_j}$. However, the size of force could not be directly measured. We consider assigning a distribution to the size of interaction forces. According to statistics, we assigned a Gaussian distribution for the size of interaction forces with respect to two parameters. One is the distribution weight ϵ, denote as:

$$\epsilon \sim N(\mu, \sigma^2), \epsilon \in [0, 1) \tag{4.19}$$

and according to our measurements, we set $\mu = 0.3$ and $\sigma^2 = 0.25$. The second aspect is the weight of the "dominant" node which has the least number of edges connecting to other nodes. Dominant nodes tend to locate peripherally on the branch. The weight, caused by gravity is related to the radius of sphere, noted as $\theta \cdot r$, where θ is

a constant coefficient. Therefore, the interaction force is given by:

$$\vec{F_{i,j}} = \epsilon \cdot \theta \cdot r \cdot \vec{e}_{c_i c_j} \tag{4.20}$$

And we could easily deduce that $F \sim N(0.3\theta r, 0.25\theta^2 r^2)$, $F \in [0, \theta \cdot r)$. As each edge in the graph $G_{\mathcal{B}}$ provides interaction force to connected nodes, the correlation for every node in this graph is given by:

$$\vec{\widehat{F}}_i = \sum_j^n \vec{F_{i,j}} \tag{4.21}$$

Note that $\vec{F_{i,j}} = \vec{0}$ if $i = j$ or nodes $\Theta(c_i, r_i), \Theta(c_j, r_j)$ are not connected.

As each tomato on the branch is kept in balance with each other and does not fall, this suggests a balance equation of:

$$\vec{G_i} + \vec{\widehat{F}}_i + \vec{F_{i,p}} = 0 \tag{4.22}$$

where $\vec{G_i}$ is the gravity of model and can be noted as $\theta \cdot r_i$, where vector $\vec{F_{i,p}}$ is the force from the pedicel that holds and keeps the crop from falling down. $\vec{F_{i,p}}$ is a multivariate Gaussian distribution due to the edges of the node. Since every node follows a same Gaussian distribution of force magnitude, we can easily calculate the pedicel vector expectation $\vec{F^e_{i,p}}$. Figure 4.5 shows a demonstration of the estimation of pedicel direction. We calculate and visualize the pedicel vector distribution on the sphere ($\phi \in [-\pi, \pi], \theta \in [0, \pi]$). For a node with two edges $-\vec{F}_{i,1}, -\vec{F}_{i,2}$ which are assigned with the same Gaussian distribution, combined with gravity-support vector $-\vec{G}_i$, we could obtain the distribution of $\vec{F_{i,p}}$. Blue color indicates low probability and red indicates high probability.

Then we perform geometric reasoning and try to find which tomato to pick first. Since the connected graph and the pedicel direction is obtained, the next task is to determine which tomato is the easiest to pick. Experience tells us tomato become ripen from the top to the bottom on the branch. This is because tomatoes located at top of the branch is closer to the root. Therefore, we created a penalize function to each primitive model considering the location, edges and the direction of pedicel:

$$C_i = \vec{\sigma} \cdot [-Ep_i, N_{\Theta(c_i, r_i)}, < \vec{F^e_{i,p}}, -\vec{G_i} >]^T \tag{4.23}$$

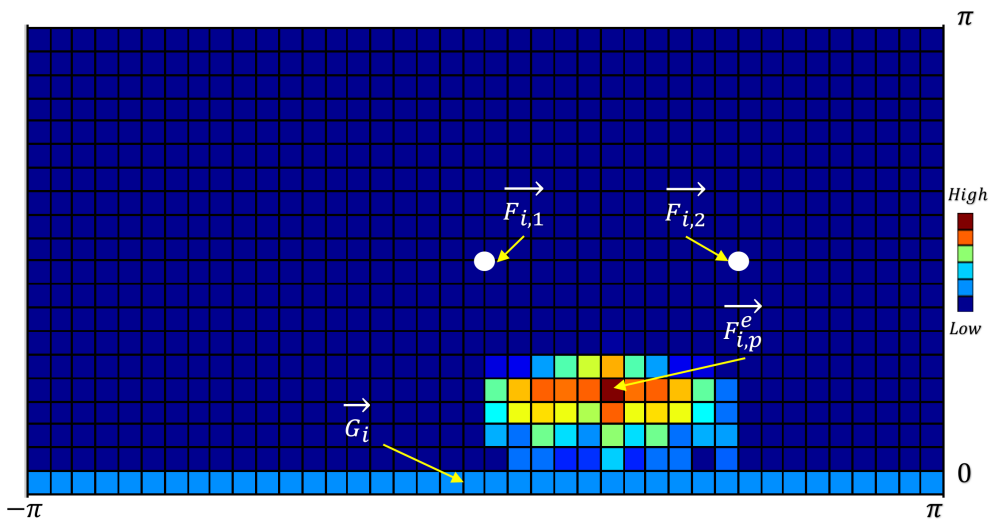


Figure 4.5: Illustration of pedicel direction distribution. (The landscape of pedicel distribution is obtained using Equation (4.22) over sphere angles ϕ and θ. Here $\vec{\widehat{F}}_i = \vec{F_{i,1}} + \vec{F_{i,2}}$, and combine with $\vec{G_i}$, probability of $\vec{F_{i,p}}$ rises from dark blue to red)

note that $\vec{\sigma}$ is a weight vector, Ep_i is the Gravitational potential energy, $N_{\Theta(c_i,r_i)}$ is number of edges of the node $\Theta(c_i, r_i)$ and $< \vec{F^{e}_{i,p}}, -\vec{G_i} >$ is the angle between the expectation pedicel vector and gravity-support vector. As Figure 4.4.C shows, the brightness of color demonstrates the picking order of each tomato in the geometric reasoning output.

Since our algorithm provides a distribution of the pedicel direction, when means a guess of the result, our picking motion should be designed to cover the potential area as much as possible, which is given by:

$$\underset{D}{\operatorname{argmax}} \iint_D P(\vec{F_{i,p}}) \cdot \vec{e_{i,p}} ds \tag{4.24}$$

$$\begin{aligned} s.t. \quad & Dist(D, c_i) > r_i, \\ & D \leq D_{max}. \end{aligned}$$

where $P(\vec{F_{i,p}})$ is the probability function and $\vec{e_{i,p}}$ is the unit vector, which means moving the gripper to cover the areas where $F_{i,p}$ draws a higher probability with respect to the radius of the tomato r_i and the gripper's covering range D_{max}. To solve this, we simplified the problem by considering a constant probability of the

gravity-support vector. Our robot will first move to cover the gravity-support vector and then move towards the vector of pedicel expectation.

4.4 Local Sensing Based Picking Verification using Tiny Laser Sensor

The limitation of the reasoning method is that basically we are based on the probability model and only gives a guess of the target, thus it is very necessary for robot to obtain a real observation. According to our experiment of tomato harvesting, due to the precision of recognition(segmentation, sensor error) and control offset, sometimes the real pedicel will be slightly away from the scissor blades(around $2[cm]$ in one tomato case). As the scissor opens only $2[cm]$, it can not hold the stem and cut it. The pedicel can not be directly detected by our RGBD camera because it is too thin and sometimes occluded, thus local sensing is very necessary for such task-oriented system.

4.4.1 Tiny Laser Sensor Continuous Measurement Mode

In the robot applications, we have to make our tiny laser cover all the measurement of objects with different distance and reflectance in the measurement range where fixed pulse number is no more suitable. We propose our continuous measurement mode of both raw data and distance data only configurations. Since the continuous measurement mode is designed to address the problems of insufficient incident light accumulation and accumulation saturation, a dynamic process methodology is necessary. Under the condition of insufficient light accumulation, the objects with further distance become undetectable and on the other hand the close objects will lead to pixel saturation problem. Consider this, we increase the pulses from 0 to a given pulse number to cover the target measurement range and readout the data under a certain pulse interval. Therefore when the pixel is lead to saturation, the previous readout data is used to calculate the distance.

The difference between raw data mode and distance data only mode is that the distance calculation is embedded in the laser in distance data only mode while the raw data mode requires host computer to do that and at the same time the raw data mode can provide intensity information. The raw data mode provide more information but at the same time requires computational resources from host computer and occupies more transmission bandwidth thus the raw data mode is relatively slower than dis-

tance data only mode. Algorithm 2 shows the distance data only measurement mode. In every timer interrupt, the laser emit one pulse and accumulate the incident light, when the timer counter reaches any of the value in predefined accumulation pulse buffer P_n, a non-destructive readout is performed and the raw data will be processed using the calibrated polynomial parameters in the main loop during the interval of accumulation pulse buffer. The distance data is renewed when a new process request comes and if any pixel arrives saturation, the previous data is reserved. After all the readout finish, we call DMA to transmit the calculated distance data.

Algorithm 2 Framework of distance data continuous measurement mode

Hardware Initialization:
CPU Initialization: System Clock, NVIC, DMA and System Watchdog;
Peripheral Initialization: ADC, DAC, High Resolution Timer, Common Timer, Serial Communication and GPIO.
Parameters:
High Resolution Timer channel delay buffer D_n: This buffer determines the accurate time delay among laser pulse, VTX1, VTX2 and VTX3.
Timer counter T and Timer frequency f: emission and receive once during the timer period;
Accumulation Pulse Buffer P_n: where at each $P_i \in P_n$, pulse accumulated to send the raw capacitor data from ADC;
Calibration polynomial parameter vector K: calibrated polynomial coefficient.
Interrupts Vector V_n: each $V_i \in V_n$ represents an interrupt and each interrupt is bound to a callback function.

1: In the main loop, The CPU processes the raw data from ADC-DMA to get the distance data using calibration parameter vector K and waits for interrupts in this mode, when the timer callback is triggered, jump to 2, when the ADC callback is triggered, jump to 3;
2: $T = T + 1$, open one shot high resolution timer under delay buffer D_n; if $T = 1$, jump to 4, if $T = P_i$, jump to 5, if $\forall P_i, T > P_i$, jump to 6;
3: ADC convert and ADC-DMA transmit completed, raw data are stored in memory and being processed in the main loop 1;
4: Initialize the receive sensor, trigger ADC to obtain the reset capacitor voltage;
5: Since $T = P_i$, the P_i pulses are accumulated, send distance renew calculation flag;
6: Transmit the calculated distance buffer through Serial-DMA and start a new measurement by set $T = 0$.

4.4.2 Laser Data Real-time Clustering using DBSCAN

In a lot of applications, the purpose of local sensing is targeting at verify and adjust the vision recognition results with precise and prompt response. The objective of

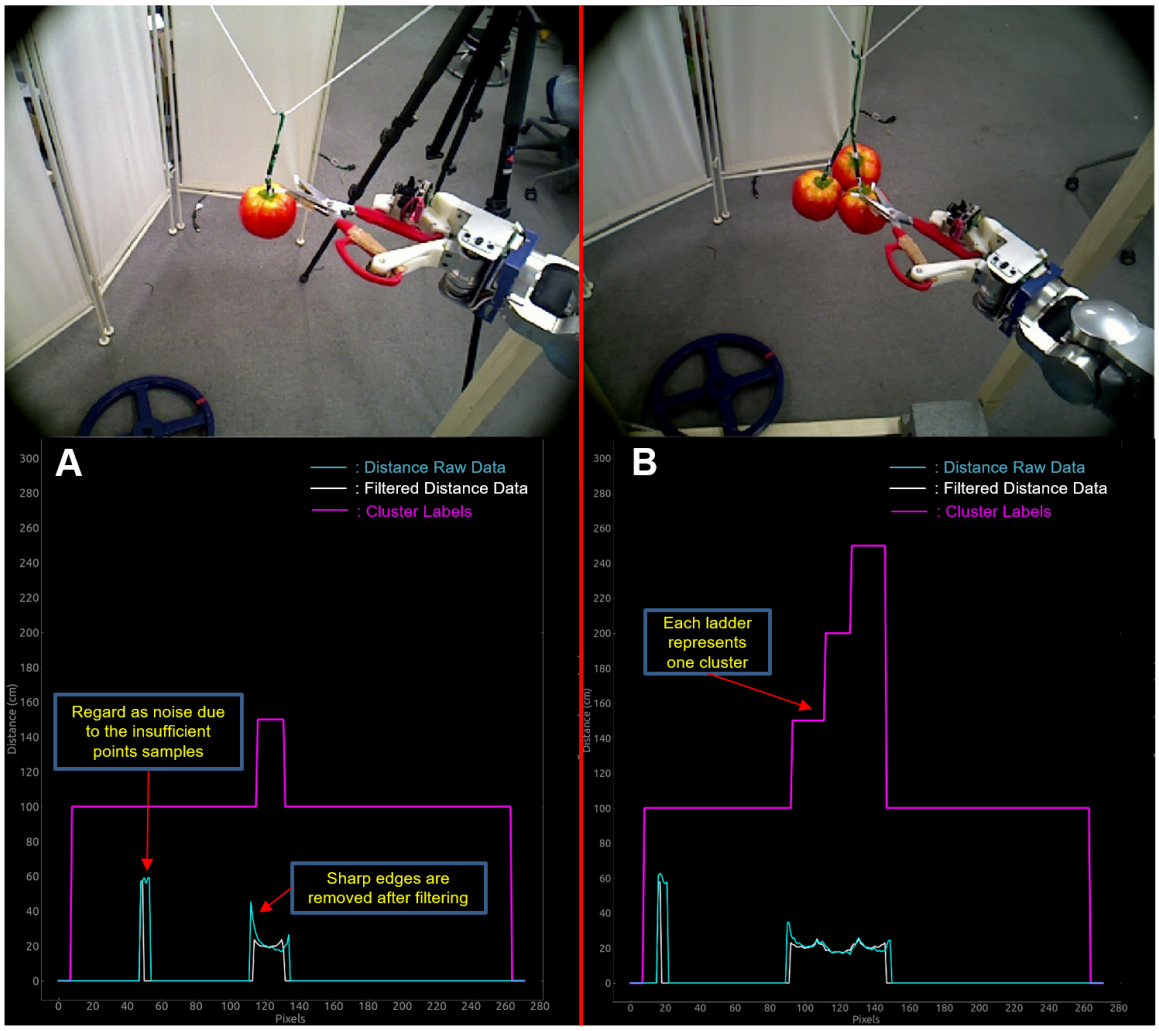


Figure 4.6: Tiny laser sensor pedicel detection. (A and B demonstrate the detecting of pedicels of simple and complex situations respectively. Blue line is the distance raw data; White is the distance data after filtering. Purple line represents the final clustering results where each ladder means one cluster.)

local sensing in this application is to detect and relocate the position of the pedicel near the scissor blade during the harvesting pose since the vision recognition and reasoning results could introduce offset errors. Thus we equipped our system with real time unsupervised clustering algorithm based on DBSCAN(Density-Based Spatial Clustering of Applications with Noise) [121]. As the raw line data of the sensor may contain distance information of other parts of the tomato branch, the clustering algorithm can separate the data points since the size of the pedicel is almost fixed(around 0.6[cm]). The DBSCAN is an unsupervised clustering algorithm and provide the number of clusters, the points belong to each cluster and the noise point. It is very suitable for our sensor to remove the noises of the object edge caused by the diffraction and neighbour pixel incident light. After obtaining the clusters, we simply check the size of each cluster and find the one that fits the pedicel prior and give the relative position to adjust the angle vector of the robot.

Figure 4.6 demonstrates the pedicel detect results using tiny laser sensor. Firstly, according to the raw data, the sharp edges caused by the base-line effect is addressed by applying a smooth filter. The base-line between the emitter and receiver causes the reflected light unreachable to the receiver at the edge parts of the object. Combined with the light diffraction phenomenon, it causes the inaccurate measurement at the edge parts of the object. This effect becomes severe when the measure object is small and close, which will lead to the object shape unrecognizable. Thus, a filtering procedure is necessary for the initial processing.

According to the observation, the emitter side edge of the measure object have a distance drag that up to tens of centimetres. If several objects gather together, only the edges of the side object suffer from the severe distance drift. Therefore, we first apply a DBSCAN clustering method with large epsilon value and bigger sample points amount restriction to produce the initial clustering of big clusters. This process helps in handling the situation where several objects gather together and the edges between these objects can be maintained. Then a smooth filter is applied to each clusters, removing the points near the edges which have large variances inside cluster first and smooth the rest points. Lastly, a second time DBSCAN with a smaller eps is applied to obtain the objects within the big clusters that produced by the first DBSCAN. Figure 4.6 explains the algorithm that the blue line is the distance raw data and the white is the distance data after filtering. Purple line represents the final clustering results where each ladder means one cluster.

4.5 Experimental Results

Our system is evaluated in terms of successful harvesting rate and detection rate for both artificial tomatoes and real tomatoes on tomato trees with different geometric complexity.

- Successful Harvesting Rate: A successful harvesting means that the robot picked the tomato and put it into the tray-box without human intervention. But for now in the experiment, humans need to send motion orders to the robot because our system is not completely autonomous yet, it requires humans to judge whether one harvesting behaviour is successful or not since the system has not equipped the corresponding functional modules. If a tomato harvesting procedure fails, we will perform the second try, and if both attempts failed, we will perform human support, namely, manually add a little offset to the gripper(almost $\leq 2cm$) or cut the pedicel again due to the gripper's performance.

- Detection Rate: Here we assume the branch is detected successfully and this rate corresponds to the percentage of tomatoes that are detected and assigned as a sphere in the tar get ranch.

- Geometric Complexity: Branch complexity represents the difficulty of harvesting since the geometrical poses varies dramatically and tomatoes may be occluded. Artificial tomatoes could better demonstrate this diversity.

The harvesting motion of our robot could be divided into four steps as shown in Figure 4.7. First the robot will localize the target branch in robot coordinate and generate the trajectory to move the hand-camera to a predetermined location relative to the tomato. Only two camera views are used in our case (note that both cameras are based on active sensing measurement, and depth data tends to be unstable in overlapping areas). In the second step, our algorithm segments every tomato and constructs the connecting relationships, then the tomato to pick and the corresponding pedicel direction will be computed through physical and geometric reasoning. In the third step, the robot will make an attempt to cut the pedicel. Currently, in this process human interference is required if the robot fails to obtain the correct position (according to our experiment, error less than $2cm$) or fail to cut the pedicel. We will provide a slight offset to the gripper or continue another cutting process by remote control. The final step is placing the picked tomato to the tray box, and then returning to the initial pose.

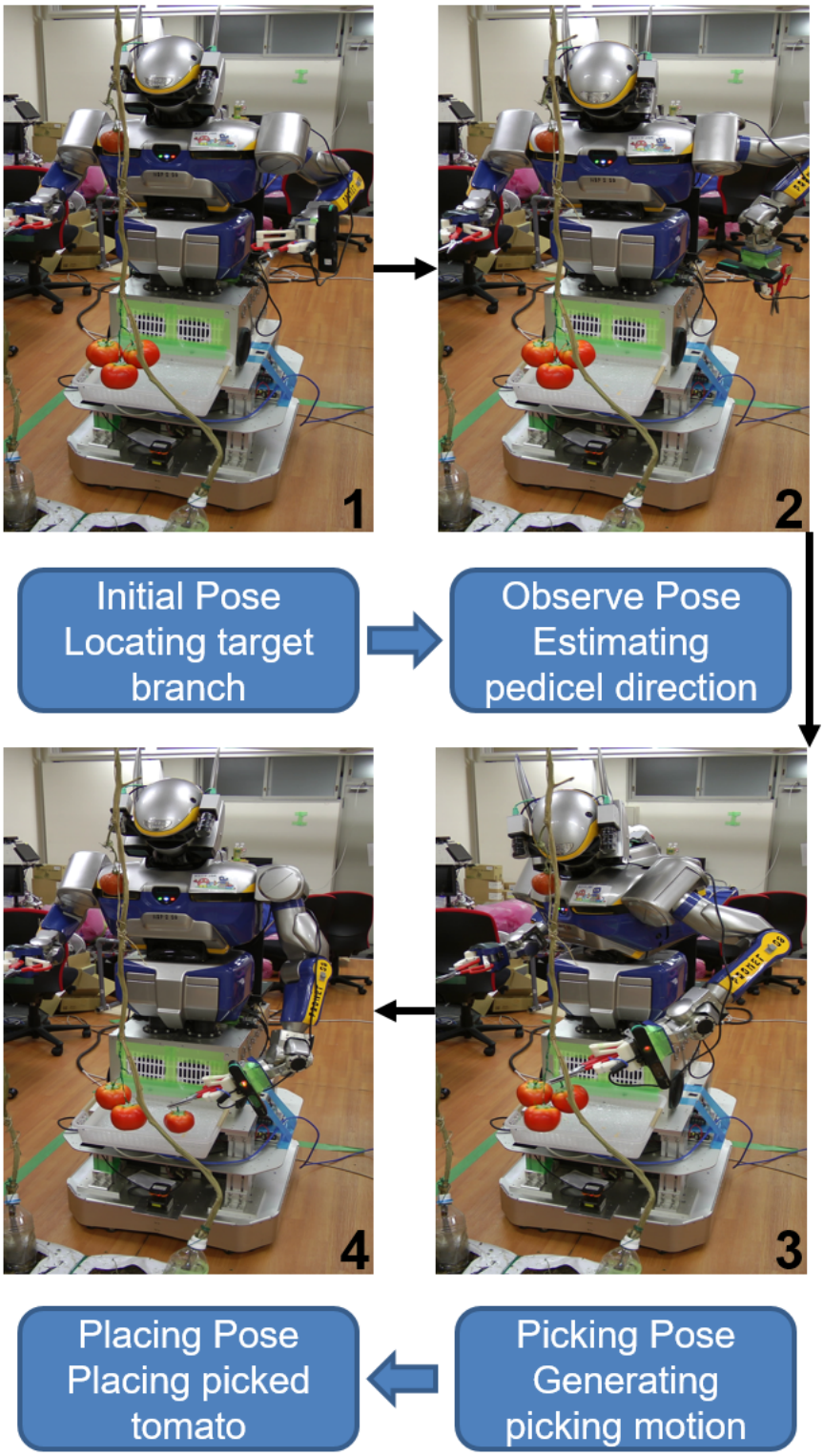


Figure 4.7: Harvesting motion. (Motions 1-4 are sent to the robot sequentially. In step 3, human interferences are required in case of picking failure and picking verification is applied using our tiny laser sensor.)

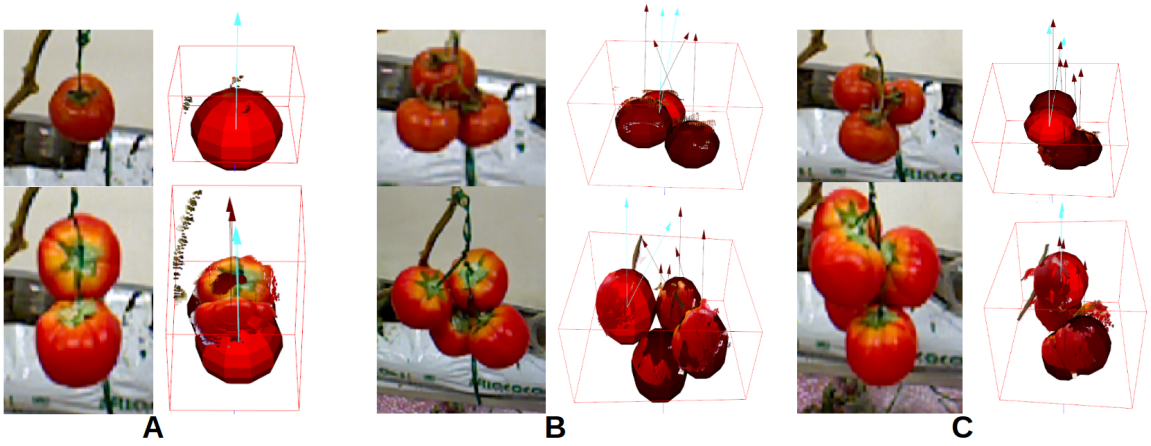


Figure 4.8: Tomato branches of different geometric complexity. A: Simple situation with 1 to 2 tomatoes, no occlusion. B: Normal situation with 3 to 4 tomatoes, slight occlusion. C: Complex situation with more tomatoes, densely gathered and occluded. (Arrows represent the pedicel vector and gravity-support vector, blue ones belong to the first tomato to be picked)

Table 4.2: Harvesting experiment. 3

Categories	Trials	SD	SP(1)	SP(2)	HS	CF
Gripper Tests	71	-	63	3	-	5
4.8-A	36	36	26	4	3	3
4.8-B	57	49	30	10	11	6
4.8-C	43	33	17	9	13	4

Figure 4.8 illustrates the classification of branch by geometric complexity. We grouped all the branches into three categories - simple, normal and complex - according to our observation of real tomatoes in tomato trees, and harvesting results are shown in Table 4.2. SP(1) and SP(2) indicate how many trials were successful by the first or second attempt respectively[3]. Every trial represents task of harvesting the detected tomato. In our test, since we also test on artificial tomatoes, the pedicel is substituted by wires with similar hardness compared to the real pedicel. Meanwhile,

[3] SD: Successful Detected
SP: Successful Picked
HS: Human Supported
CF: Cut Failure

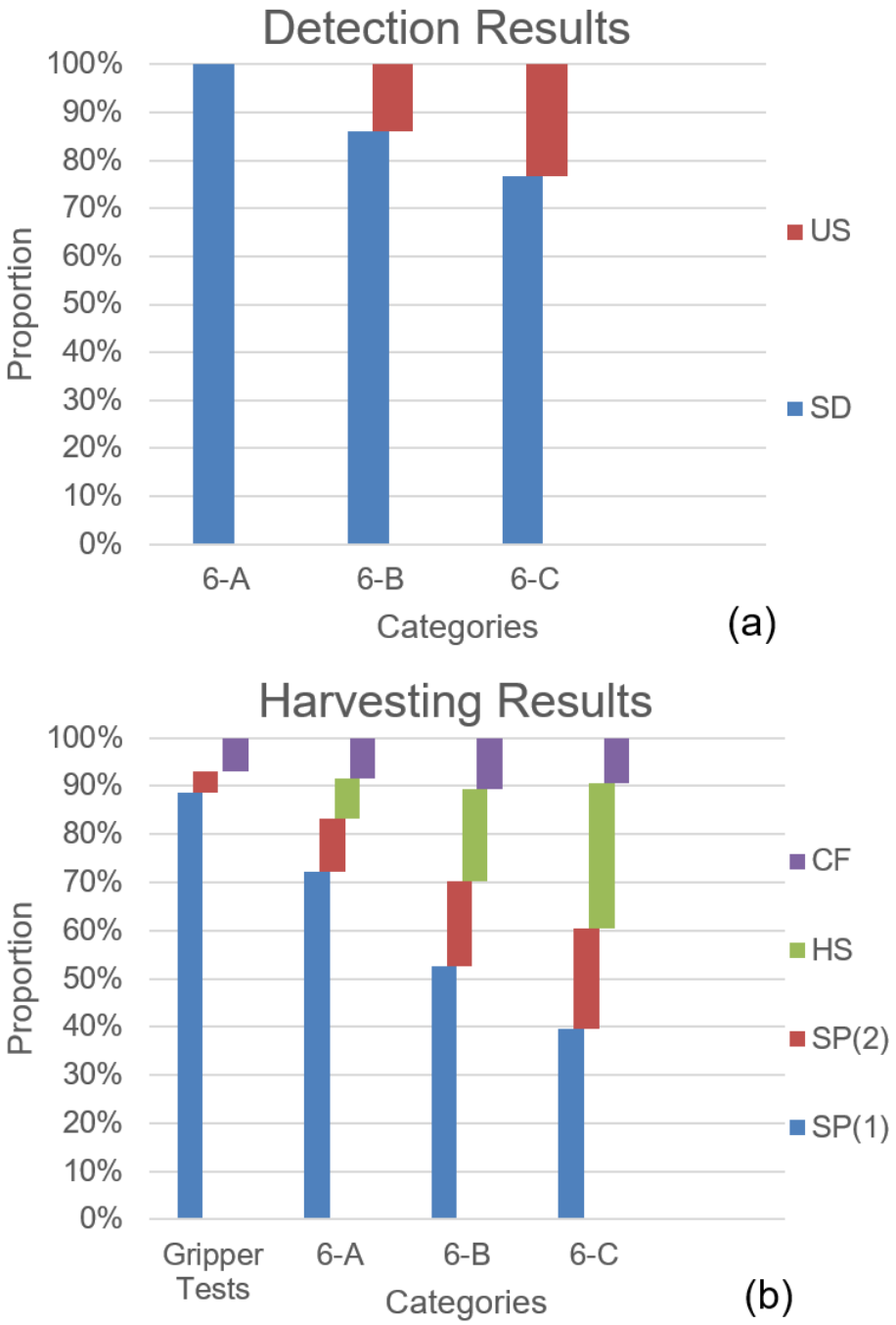


Figure 4.9: Detection and harvesting results. (a)Detection results: The percentage of detected tomatoes out of the total number of tomatoes on target branch. (b)Harvesting results: Gripper tests are performed separately to cut the real tomato pedicel. SP(1) indicates successful picking by first try, SP(2) indicates successful picking by the second try, note successful picking is SP=SP(1)+SP(2).

Table 4.3: Laser guide picking experiment

Categories	Trials	Successfully Aligned	Wrong Aligned	Out of View
Random Offset Alignment	61	56	2	3
Recognition Error Alignment	39	26	11	2

we perform gripper tests separately just for testing the successful cutting rate of real tomato pedicel. It can be inferred that multiple cutting attempts would contribute towards increasing the successful harvesting rate.

The local sensing based picking verification using our customized tiny laser distance data can help dealing with the offset and recognition error. We randomly assign a offset of less than $5cm$ to the recognition result and once the robot reaches the observation pose, the laser guide local sensing function is called in our algorithm and the effectiveness of the offset alignment experiment is shown in Figure 4.10 (A) and Table 4.3. We perform 61 offset alignment trials of above condition and 56 trials successfully align the scissors to the pedicel while 2 cases mistaken the nearby pedicels as target and 3 times the laser gets out of view. Note that it is more likely to miss the pedicel in the view for one tomato because there is no nearby pedicels and even the nearby pedicels can cause to wrong alignment, the algorithm seems capable of finding the nearest pedicel for most of the time.

In addition, the local sensing framework can also deal with the recognition error of the vision algorithm. Figure 4.10 (A) indicates the correct detection of all three tomatoes thus the pedicel direction is correctly estimated. However in Figure 4.10 (B), because of the sampling based algorithm may sometimes fail to extract all the possible candidates of tomatoes which lead to the incorrect detection, the vision result contains the error of the pedicel direction and position. Based on the tiny laser distance information, the robot can verify whether the picking attempt will successfully have something to cut in the observation pose. If there is nothing detected in the picking attempt area, the robot will enter laser guide mode to locate the nearby pedicels to pick as Figure 4.10 (C) demonstrates. For this experiment, we tested 39 trials of recognition error alignment. Note that the recognition error are mainly like Figure 4.10 shows that the target tomato is correctly detected but the pedicel position and directly is inaccurate. In the experiment, applying the local verification method we achieve 26 successful alignment. There are 11 wrong alignments that the nearby

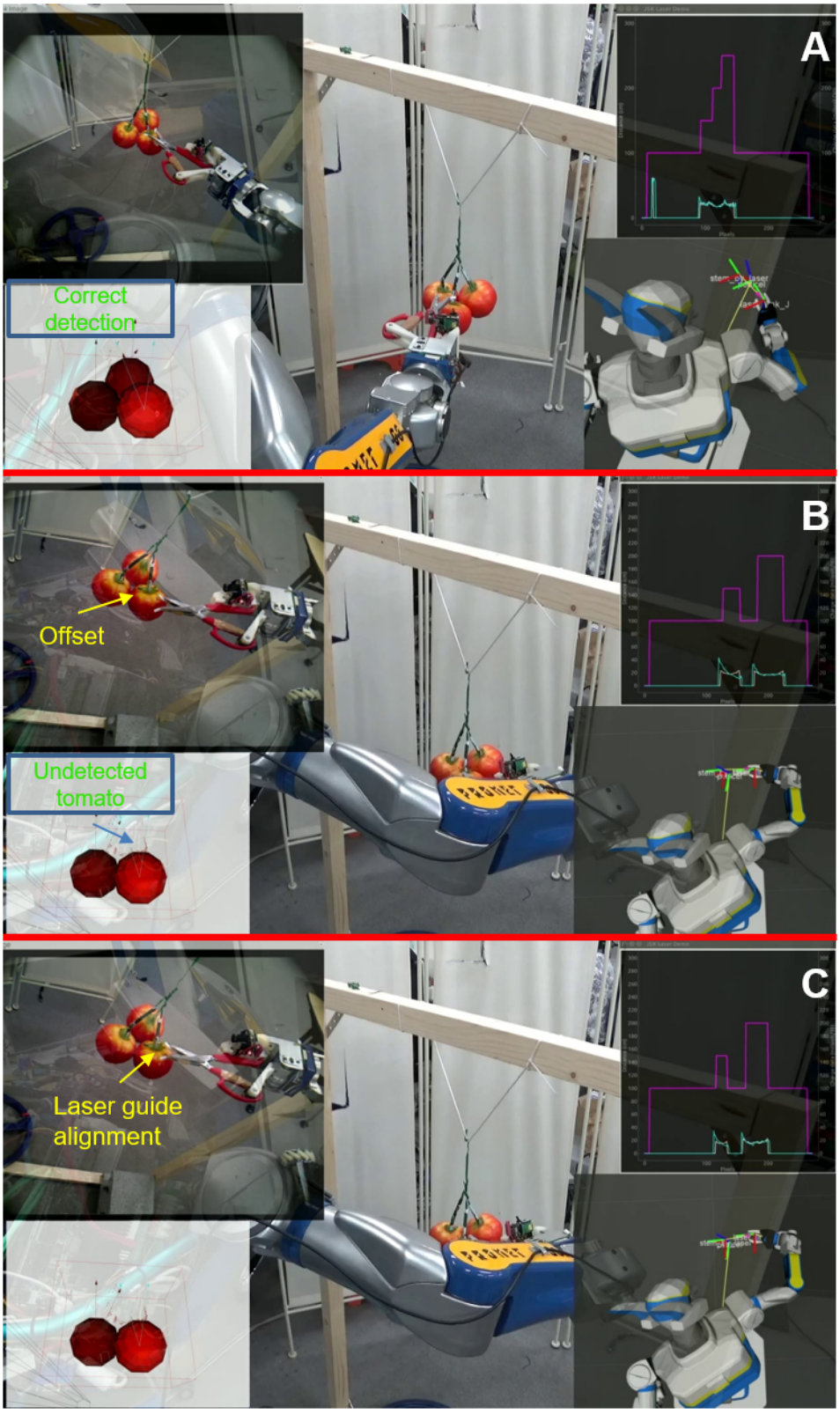


Figure 4.10: Laser Guide Picking. (In A all the tomatoes are detected; In B a offset appears due to the recognition error; In C a laser guide motion compensation is performed.)

pedicels are detected and the edge of tomatoes are recognized as pedicels. This is because the direction of the pedicel is inaccurate which lead to the nearest pedicel uncertain in the recognition error case.

4.6 Summary

In this chapter, a harvesting humanoid robot system is proposed and we presented a novel vision cognition approach that enables the robot to harvest tomatoes gathered in branches through physical and geometric reasoning to have an initial guess of the pedicel of apply our light weight laser system to do perform the harvesting based on local sensing. In the experiment we showed the feasibility and effectiveness of our system framework of combining reasoning based high level vision and local verification based laser sensor guiding picking motion. In addition, the effectiveness of our tiny TOF laser line sensor system and its flexible usage are illustrated in this task-oriented robotic application.

ACKNOWLEDGMENT

This work is a cooperative research with DENSO corp.

Chapter 5

Development of Metallic Tools Grasping Robot System Based on Reflectance Feedback Using Tiny Laser Sensor

5.1 Participation in MBZIRC and Metallic Tools Grasping Task

The MBZIRC [4] is short for Mohamed Bin Zayed International Robotics Challenge, which is held in Abu Dahabi, United Arab Emirates(UAE). The MBZIRC is a open challenge for the world-wide robotics researchers and the selection of teams are evaluated by the submitted proposals by the organization committee.

There are four challenge tasks in MBZIRC, challenge 1 and 3 are focusing on UAV(Unmanned Aerial Vehicle) robot, challenge 2 is aiming at UGV(Unmanned Ground Vehicle) robot and the last one requires all the robots to work together. We submitted our proposal [122] and obtained the qualification with a sponsorship of $350,000$ USD from the MBZIRC committee. Along our preparation for MBZIRC, we also submitted two progress reports [123][124] to the committee according to the obligations of the sponsored team. In the proposal and progress reports, the general

[4] MBZIRC(Mohamed Bin Zayed International Robotics Challenge)
Website: `http://www.mbzirc.com`

framework and methodology of our system, the solutions to the difficulties of each challenge we have met and the preparation time-line are stated [5].

In this chapter, we introduce the robot systems we build for participation in the MBZIRC challenge 2 of robot platform design, hardware development, task-oriented vision approach and the results we achieved along the preparation. In addition, we proposed a local sensing framework toward to wrench grasping task using the customized tiny laser sensor we developed to obtain the reflectance of the wrench surface. By providing both the range and intensity information, with both information we can infer the reflectance of the object surface, we demonstrate the effectiveness of the robot system in grasping the target wrench with a precise alignment of gripper and wrench.

5.2 A Task-oriented High Power Field Robot Platform For Robotic Challenge

In this section, we describe a task-oriented field robot platform with humanoid upper body and mobile wheeled base. Our goal is design a robot platform that can handle outdoor tasks rapidly and stably which requires the collaboration between the upper body and mobile moving base. Therefore, we developed a powerful mobile base with two $350W$ motors and can run at a speed of $4m/s$. The payload of the mobile base can up to more than $50kg$. We modelled the kinematics of the mobile wheeled base and created the control system of both hardware and software. Like ordinary HRP2 humanoid robot, we enable all the features of HRP2G upper body and also add the geometry model of the mobile base into the euslisp simulation viewers. The two parts are connected through ROS in software level which makes it much easier to maintain the whole system. In the MBZIRC task-oriented experiment, we equipped this robot with a multi-sense sensor and GPS sensor in the head, a velodyne laser on the base and a customized gripper.

There are a lot of integrated robot system researches in recent decades. The widely used wheeled robot Pioneer enable researchers to integrate a robot arm with several DOFs[125] for manipulation and visual servo feedback grasping[126]. However for now, Pioneer robot is a little bit small and the payload is only about $17kg$ which is somehow not very adequate for some field tasks. More recently Clearpath com-

[5] Progress Videos:
`http://www.jsk.t.u-tokyo.ac.jp/~xychen/mbzirc/first_report_all.mp4`
`http://www.jsk.t.u-tokyo.ac.jp/~xychen/mbzirc/second_report_all.mp4`

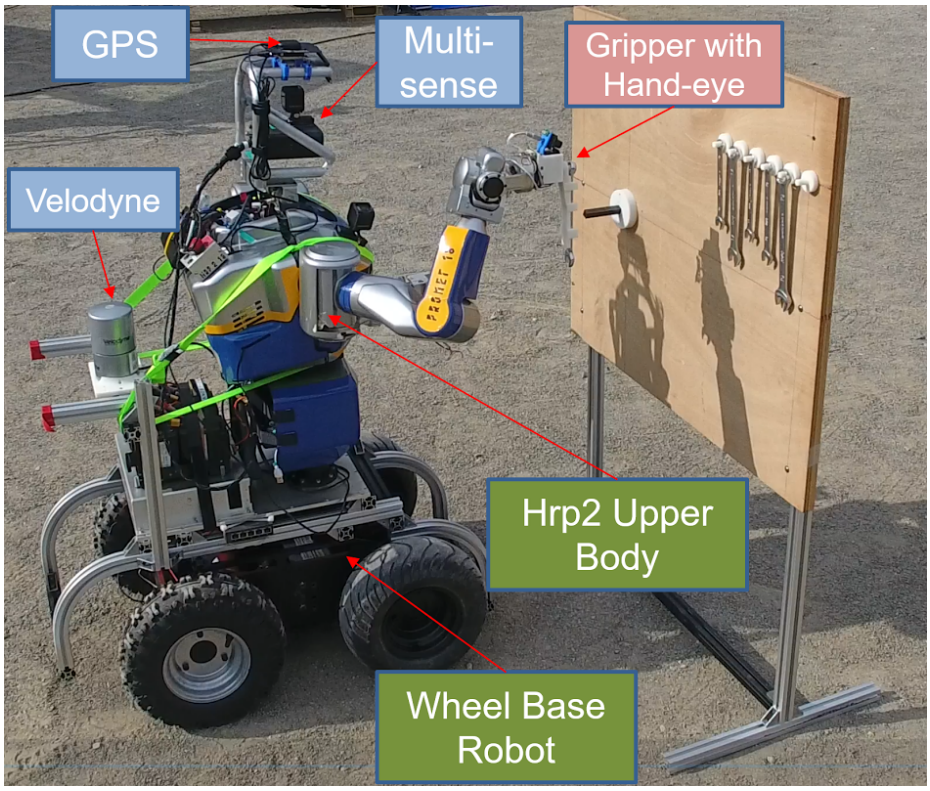


Figure 5.1: HRP2G Robot Platform Practice in MBZIRC 4 Tasks.

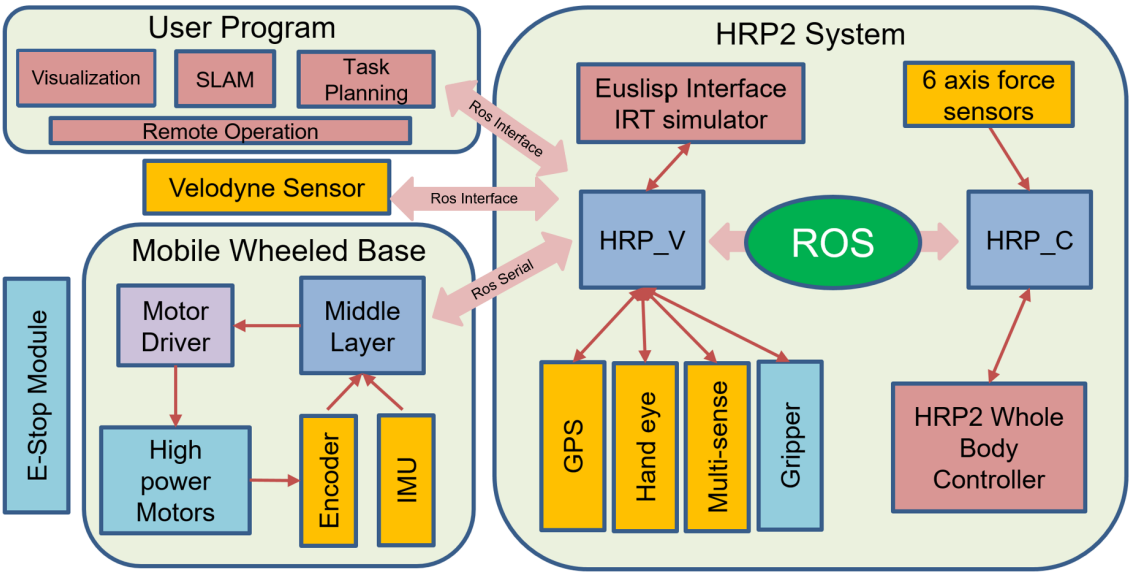


Figure 5.2: HRP2G Robot System Pipeline

pany has announced their powerful field robot Husky with a payload of $75kg$ which enable it to carry a small industrial robot arm like UR5, and capable of more field tasks[127][128].

The system integration research of humanoid robot with a mobile wheel base is not a methodology as popular as a robot arm. However, the importance of this research remains clear that people tend to have a higher acknowledgement of humanoid robot because they looks like human and people also trust that the humanoid can handle human's tasks. In the previous works of our lab, we developed this kind of robot platform named HRP2W mainly for indoor applications, like home-assistant(cooiking, cleaning, moving objects)[129][130], also greenhouse agriculture harvesting[2]. But for outdoor use, we need a stable and powerful mobile base with big and soft tyres to work in the unpredictable field.

5.2.1 Hardware and Software Architectures

As Figure 5.1 shows, the humanoid platform HRP2G for MBZIRC consists of two main parts, the HRP2 upper body and the mobile wheeled base. The normal HRP2 upper body has 20 DOFs and for the purpose of installing gripper, we remove the finger joints of both hands and install the gripper we designed in right hand, the left end effector is empty now and we plan to install a sensor in the left hand. The HRP2 upper body is no difference with the normal HRP2 robot both in hardware and software. The development environment is under Euslisp [6] programming language with ROS(Robot Operation System) and IRT interfaces.

Table 5.1: Specification of HRP2G

Name	DOFs	Sensors	Global Property	
HEAD	2	Multi-Sense	DOF	21
ARM	7×2	6-Axis Force Sensor	$Length \times Width \times Height$	$90 \times 60 \times 130(cm)$
Gripper	1	Hand Eye		
WAIST	2	None		
Mobile Base	2	Velodyne IMU	Total Weight	$\approx 80kg$

[6] `https://github.com/euslisp`

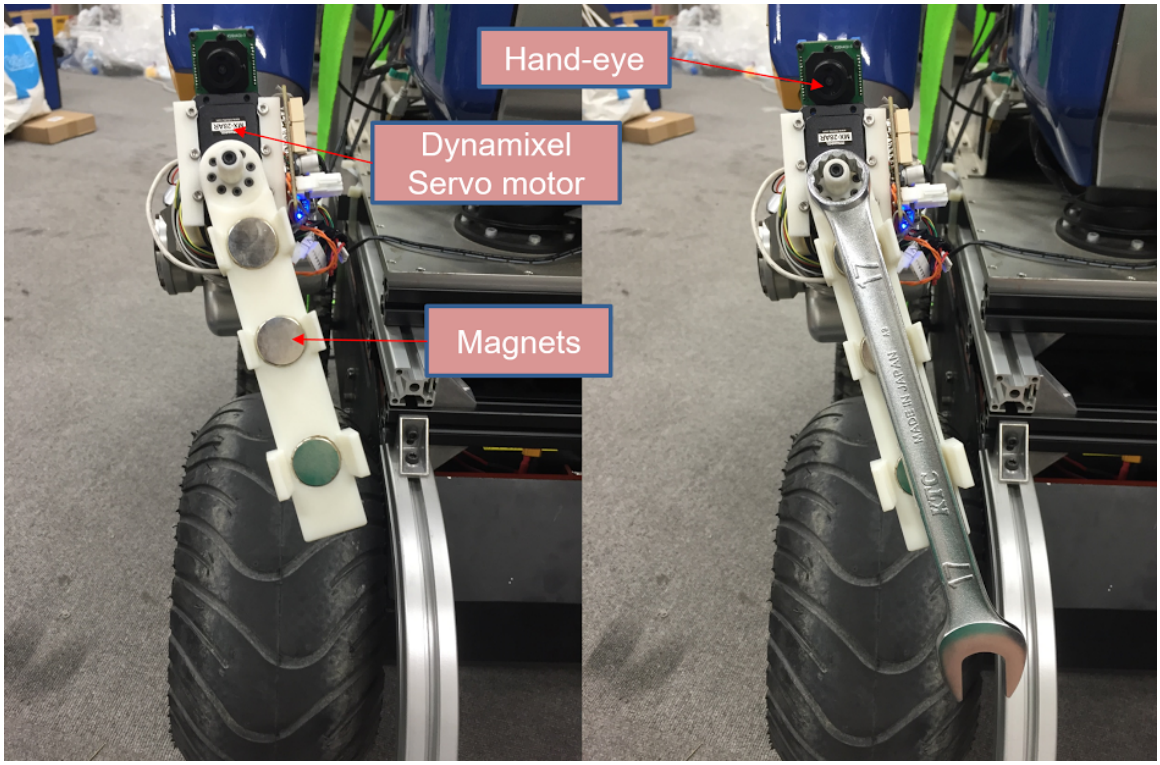


Figure 5.3: HRP2G Gripper for MBZIRC tasks

For the mobile wheeled base we install two $350W$ brush motor and high power motor driver. We design a middle layer MCU(Micro Control Unit) board to control the motors via the feedback of the encoders and IMU in the PCB board. The communication of MCU board to the humanoid computer is modified to be completely under ROS software architecture, the mobile base represents as a single node in the ROS and continuously publish the state of the base and take action indications from the motion planning layer.

As Table 5.1 shows, despite of the factory configured sensors of HRP2 and mobile base, the robot equips several type of sensors. We install the Multi-sense stereo camera on the head of HRP2, collaborating with the head and WAIST movement, the visible range is more than 180 degree. A 32-line Velodyne laser range sensor is installed on the mobile wheeled base with an adjustable height to cover $180 \sim 360$ degree of view. As for many field tasks, it requires robot to do accurate manipulations, we also install hand-eye on the end effector of HRP2. A GPS is used for the outdoor localization for the robot and a wireless E-STOP is integrated in the mobile wheeled base.

For the task-oriented application, we create a gripper that can pick the iron wrench and fix it into the shaft. As Figure 5.3 shows, we put a Dynamixel servo motor at the hand joint and design 3D-print models with magnets installed. The hand-eye is fixed on the hand joint and we are using the data to align the gripper toward the wrench. The Dynamixel servo motor is used for align the wrench and when the robot is trying to fix the wrench to the shaft, Dynamxiel can also provide torque to turn the shaft. At this moment the permanent magnets are used and we are planing to use electronic-magnet later so that the robot can both pick and release the wrench.

We build a hardware middle layer circuits board to control the mobile wheeled base and also communicate with the upper computer. A 32 bits MCU is used as the control unit, we select the newly published STM32F7 series MCU with ARM cortex-M7 cpu architecture. The MCU is very powerful and it can achieve 462 DMIPS computation ability. We also installed 9-axis IMU sensor(MPU9250) in the board and attribute to the high performance of MCU, we can obtain a pose estimation rate of $1khz$. In the PCB board we also add encoder counter circuits, power management circuits, CAN bus interface, ADC(analog to digital converter) interface, motor control PPM interface and serial to USB module. The encoder data and IMU data are collected by the MCU as the control feedback and MCU controls the motor according to the indications from the upper computer. A radio receiver is connected with the PCB board and human can take over the control by simply press a button in the radio controller that paired to the radio receiver.

5.2.2 Mobile Wheeled Base Kinematics and Optimized Feedback Control

As our mobile wheeled base uses only two motors, the front wheels and the corresponding rear wheels are connected by the chain, the kinematics is the differential drive model[131]. We applied the skid-steer drive model to describe the kinematics of the mobile base.

Consider the control inputs of the system are only the velocity of the left wheel and the right wheel, which could be noted as:

$$M_{input} = \begin{bmatrix} V_l \\ V_r \end{bmatrix} \tag{5.1}$$

where V_l, V_r are the corresponding velocity of left and right wheels.

The output of the model is the state of vehicle, note as q, for simplicity we only consider 2-D situation in our model. Then we have the model output:

$$M_{output} = q = \begin{bmatrix} v_x \\ v_y \\ w \end{bmatrix} = f_m \begin{bmatrix} V_l \\ V_r \end{bmatrix} \tag{5.2}$$

where v_x, v_y are the linear speed of the vehicle in self coordinate, namely the lateral velocity and the forward velocity respectively. w represents for the angular velocity in 2-D situation, namely the yaw velocity.

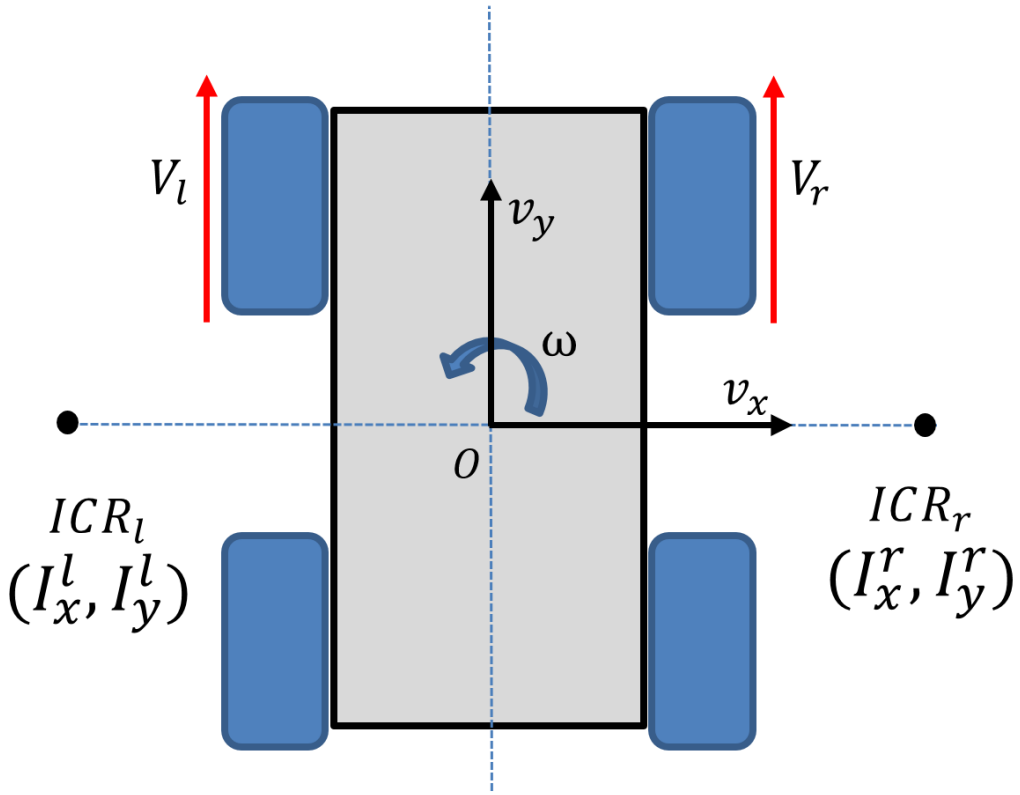


Figure 5.4: Idea ICR Model

To get the vehicle kinematics model f_m, we need to consider the ICR(Instantaneous Center of Rotation) of the mobile wheeled base. [132] introduced the geometry of the skid-steer model when considering ICR in the system. The correction factors are introduced to present the mechanical matters, note as (a_l, a_r), and the model f_m is

only related to the ICR coordinates and the correction factors:

$$f_m = \frac{a}{I_x} \begin{bmatrix} 0 & 0 \\ I_x & I_x \\ -1 & 1 \end{bmatrix} \tag{5.3}$$

where I_x is the x lateral coordinate of the ICR point, and Equation 5.3 is under the ideal situation for simplicity where $a_r = a_l$, $I_y^r = I_y^l = 0$ and $|I_x^r| = |I_x^l|$.

In a lot of cases, the I_x can be noted as:

$$I_x \approx \frac{V_r - V_l}{2w} \tag{5.4}$$

Since for most of the applications, only kinematics odometry wont achieve an acceptable result for localization, multi-odometer system is applied like GPS and visual SLAM. In the skid-steer model, the lateral velocity v_y become obvious when the vehicle is turning and it is very difficult to accurately estimate the value in real world environment where the terrain is always unknown.

We use the encoder and IMU data as feed back and control the velocity of each wheel. Traditional PID approach could achieve a good result in most situations. As PID law can be expressed as:

$$\Delta V(t) = K_p e_v(t) + K_i \int_0^t e_v(\tau) d\tau + K_d \frac{de_v(t)}{dt} \tag{5.5}$$

where $e_v(t)$ is the difference between the aim speed and the measurement speed, $V(t)$ is the output duty rate to the motor driver. As we need the output to react as fast as possible, we set the control rate to $1khz$ in respect to the encoder update rate.

As Equation 5.5 shows that a good estimation of the current speed and acceleration is very necessary to the control system. Direct approach suffers from the encoder resolution and quantization error. We choose the polynomial fitting method[133] to estimate the parameters.

Consider:

$$P = (p_1 p_2 p_3)^T \tag{5.6}$$

where P is the parameters vector, p_1, p_2, p_3 are proportion to the current acceleration, velocity and the encoder counter value respectively. Then we collect the encoder state

with a certain frequency, then we have:

$$X = (x_1 x_2 ... x_n)^T \tag{5.7}$$

with time matrix:

$$H = \begin{bmatrix} t_1^2 & t_1 & 1 \\ t_2^2 & t_2 & 1 \\ . & . & . \\ . & . & . \\ . & . & . \\ t_n^2 & t_n & 1 \end{bmatrix} \tag{5.8}$$

matrix X is the encoder value corresponds to the time matrix H. When $n > 2$, the linear equation is given by:

$$HP = X \tag{5.9}$$

to estimate the parameters, we apply Least Square method:

$$P = (H^T H)^{-1} H^T X \tag{5.10}$$

if we ignore the square part, the estimation turns out to be a average function of all measurements. Taking real-time into consideration, we choose 5 time interval in the estimation procedure.

5.2.3 Experimental Results of HRP2G in Test Field

In task 2 of MBZIRC, the robot HRP2G is requested to operate a valve stem using the given wrench tools on a randomly placed panel in the challenge arena as Figure 5.1 shows. Firstly the robot must locate the panel in the arena and moving toward to it and align the robot to the surface and the panel. Then the robot need to detect the wrench tools and choose the appropriate tool to use because only one wrench will fit the valve stem. Next we need to pick it up, and manipulate it to rotate the valve stem one full circle (360 degrees) and a torque of approximately 5 Nm will be required for the robot to operate the valve. The challenge is judged by three aspects: completion, autonomous and effectiveness(time)[7].

[7]MBZIRC TEAM-JSK First Report Video
`http://www.jsk.t.u-tokyo.ac.jp/~mizohana/mbzirc/task2_1.mp4`

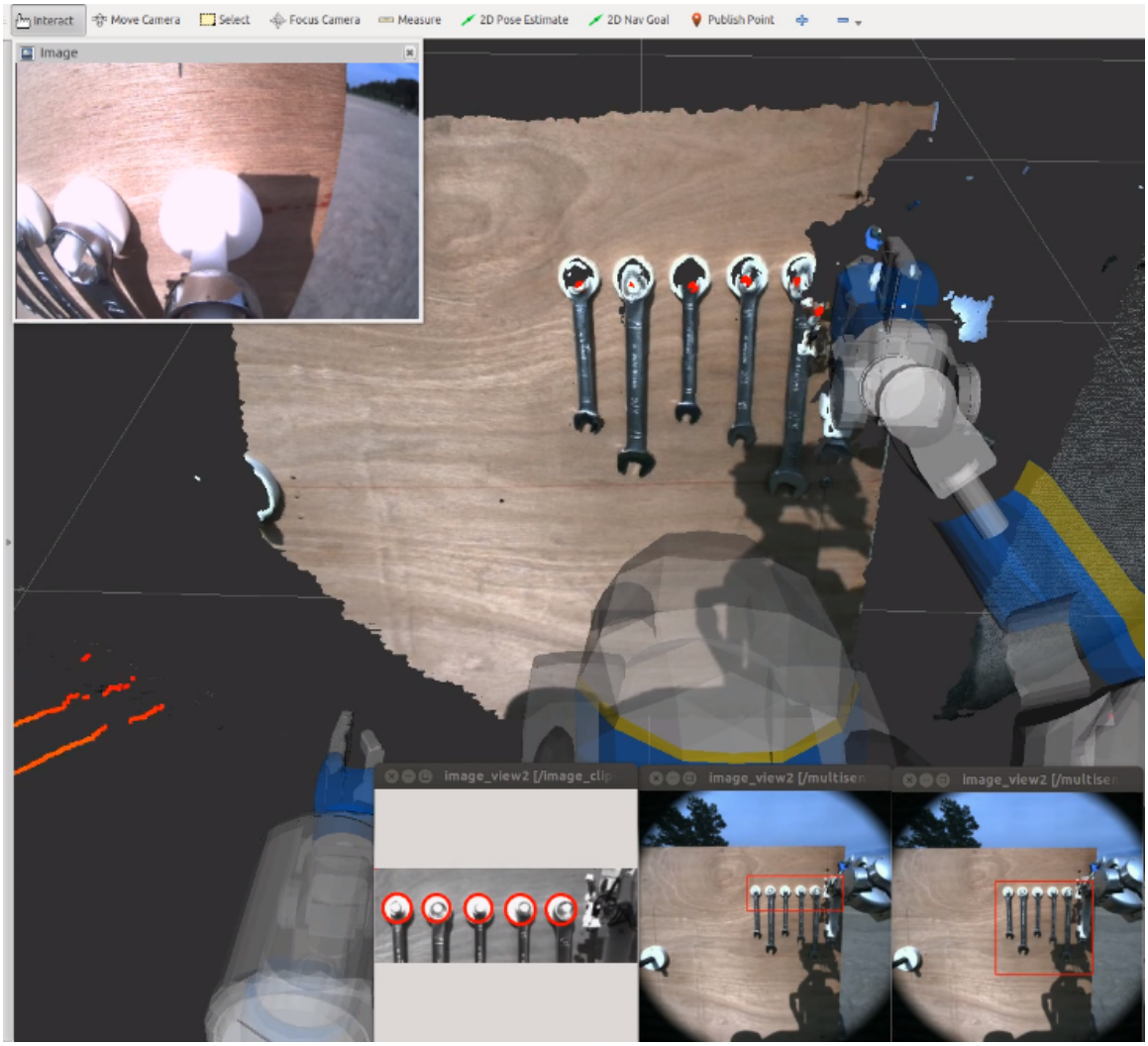


Figure 5.5: HRP2G Picking the Wrench 7

The advantage of our HRP2G platform in the challenge is that attribute to the high power of the mobile wheeled base, the robot can run at a very fast speed($\approx 4m/s$). We consider the maximum end effector load of HRP2 is about $4kg$, we design the gripper to pick the wrench and make sure the torque arm length is $> 15cm$ which results in $< 3.4kg$ force requirement for the end effector and justified the feasibility of our robot to operate the valve.

For now we already had a stable platform and hardware configurations and as Figure 5.5 shows, we also developed a detection algorithm that only need minor annotation from human. We accomplished the whole task by human remote control in about 10 minutes and for the next we are mainly focusing on making the whole system autonomous.

We perform the test as Figure 5.6 shows. The Figure 5.6(A) is the control result of the linear move test, namely, forward and back-forward. The Figure 5.6(B) is the control result of angular speed. The thick brown and pink line are the given orders of angular and linear velocity respectively, within the same ROS topic $/cmd\_vel$. The green, blue and red line are the feedback data collected by the sensors like IMU and encoders. We can see that the linear velocity control is relatively smooth and the angular velocity control is vibrating(within $0.2rad/s$). This is because the type we used now is not very flat and the wheelbase is a little bit short, also the chain drive system made it difficult for the algorithm to control angular velocity while keeping the linear velocity controlled. According to our experiments, we think the linear velocity control is acceptable and we are going to reduce the vibration of angular velocity control by changing the hardware and apply adjustable parameters control algorithm to our system.

The gripper we made works perfectly both in the preparations and the final challenge. The magnet is strong enough to pick the wrench with about $5cm$ and the gripper have a stem to lock and hold the ring part of the wrench so that even if the gripper is not well aligned with the wrench, the wrench will adjust itself to fit into the gripper. When the align error is too large, we use hand-eye sensing to adjust the offset.

We also perform the experiment on practising the MBZIRC task in field ground. Firstly the robot localizes the panel and approach the panel, then align itself and move to pick the wrench. After the wrench is successful picked the robot move again and align itself to the stem shaft and insert the wrench to rotate the stem for 360 degrees. In the experiment, we encountered several challenges. As the upper body manipulation range is limited due to the inverse kinematics of the humanoid robot,

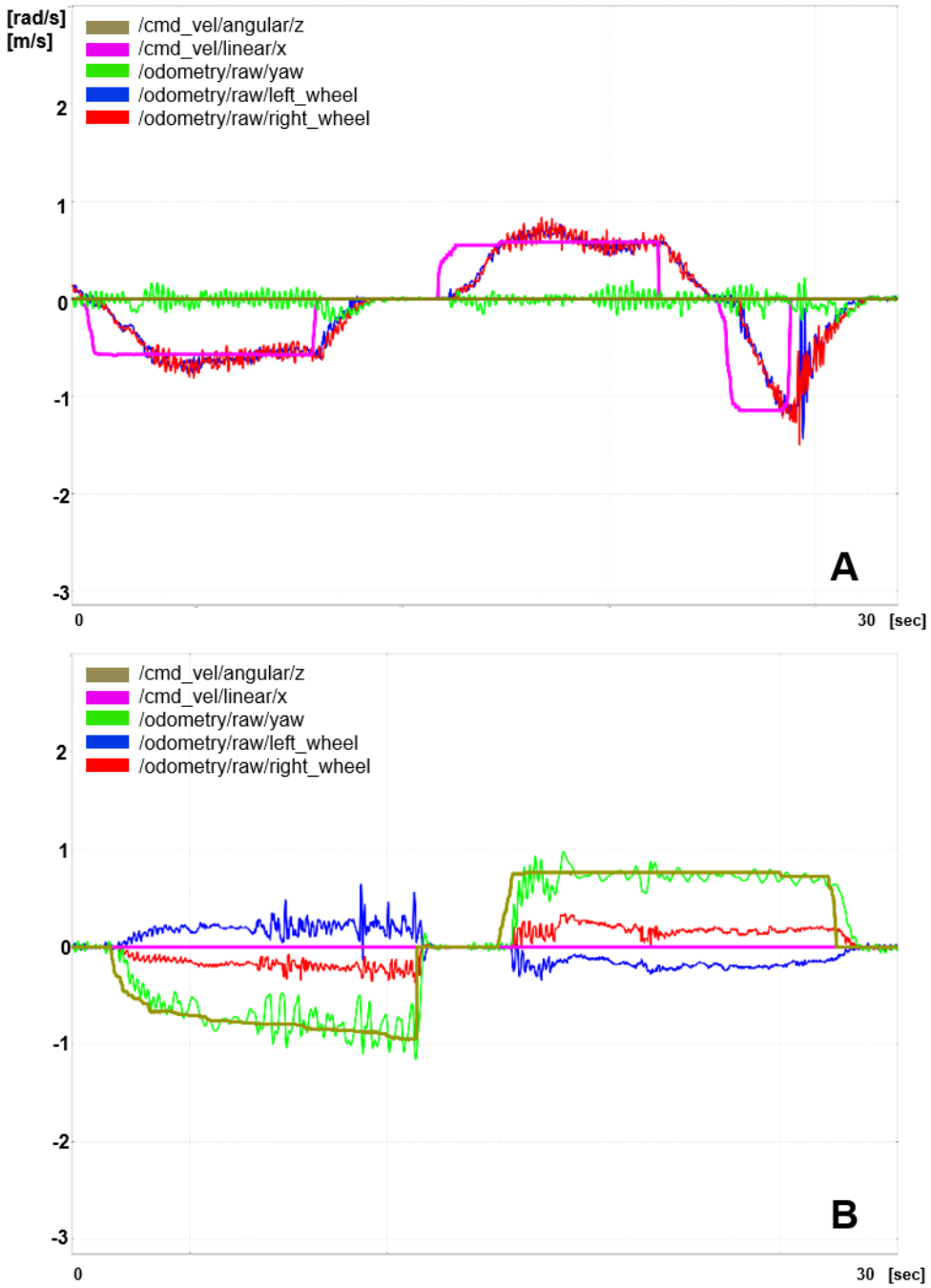


Figure 5.6: Mobile Base Control Test: Figure A shows the control of the linear velocity. Figure B shows the control of the angular velocity. The */cmd_vel* topic contains the aim angular and linear speed orders and the */odometry/raw* topic records the response data collected by the IMU and encoders(Note here we apply ROS coordinate where the forward direction is x axis in robot coordinate). In the figure, the x-axis is the time coordinate in second, y-axis is the speed in m/s and rad/s.

the mobile wheeled base must move accurately to approach the panel. If the error is a little bit large, the IK can not be solved. Temporarily we judged the situation by human beings and make decision to move the mobile base to align the robot itself and in the future we will add a full body motion planning to the robot so that the robot will move by itself when the IK can not be solved. Also in the testbed we tested the highest velocity of the mobile wheeled base and we found out that because the centroid is a little bit high, when the mobile base brake suddenly, it is very likely the robot will fall down. Same things will happen if the mobile base moves a sharp turn. To solve this we add some protection mechanism(also for large obstacles) and set acceleration limitation to the mobile base control. Neither the linear acceleration nor the angular speed in linear velocity situation is limited. However, the acceleration limitation is hard programmed and only can ensure the safety of the robot.

In the preparation trial at test field we can finish the whole task in 10 minutes with minor human remote control in the experiment. Half of time is speeded on rotating the stem because it is very hard to fix the wrench into the stem, which is also considered as the most difficult part in this task.

5.2.4 Discussion

In this section we described the high power field robot platform with humanoid upper body and mobile wheeled base. We built both the hardware and the software of this platform including the mechanism attachments, hardware PCB board, and software driver bridge. The kinematics motion and control of the mobile wheeled base are analysed and realized in the our system. Targeting at the challenge, we justified the feasibility of a specific gripper to pick the wrench tools. In the experiment, we test the control output of the mobile wheeled base in different terrain.

Although at the final challenge we decided not to take this robot platform and instead we use a single arm robot AERO which is built through a collaboration project between our lab and THK company due to the consideration of the portability, maintainability and cost, we justified our high level system architecture toward this task and demonstrate the feasibility of this approach. Note that most of the sensors, algorithms and gripper are directly applied in the AERO robot and we successfully picked the wrench autonomous at the final challenge and ranked 5 in the challenge.

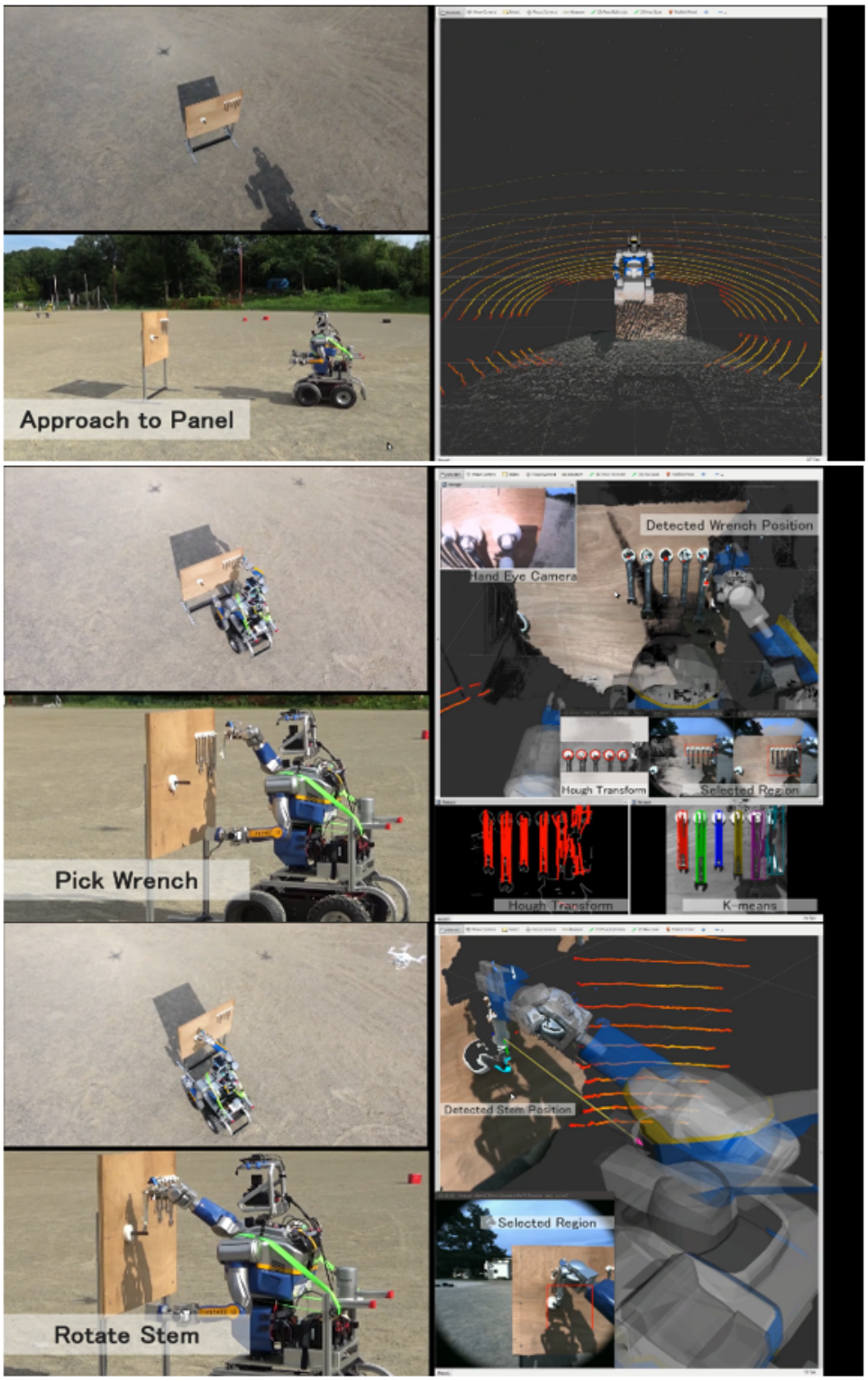


Figure 5.7: Experiment on MBZIRC task. Three steps: Approach to the panel; Pick the wrench; Rotate the stem.

5.3 Robot Wrench Manipulation Based on Customized Tiny Laser Sensor using Object Reflectance Information

For this challenge, the difficulties lie in the correctly detecting the wrenches. Since in the challenge, the ground truth size, position and orientation of wrench holder are given. All teams adopt the prior shape information of the panel since the panel corners are very easy to be detected. These corner information will provides a plane and the wrench hold position as well as the stem shaft position can be deduced. However, when facing the real world tasks, some of these information can not be always provided. Besides, sometimes the target objects can not be directly detected using the traditional sensors due the surface reflectance. In this section, we demonstrates the application of manipulation using our ultra-tiny line laser sensor to detect the metal wrench tools based on the intensity data.

As Figure 5.8 shows, the intensity data can provide the reflectance information of the objects surface. Therefore for the situation which the distance between the object and the background plane is indistinguishable 5.8 (B), the distance based clustering approach fails while performs well in situation 5.8 (A). Thus the additional information of the object surface reflectance can open a new perspective to detect the objects with the indistinguishable distance problem.

5.3.1 Raw Intensity Data Continuous Measurement Mode

To obtain the intensity data, namely the raw capacitor charge data of the tiny laser sensor, the raw(intensity) data continuous measurement mode is proposed. Algorithm 3 indicates the how this measurement mode works.

The difference between the raw data mode and the distance measurement mode which mentioned in the last chapter is that all the raw data are transferred to the host computer and the distance calculation process is performed on the host computer. The raw data are transmitted to the host computer at several accumulation pulse numbers which note as accumulation pulse buffer P_n in Algorithm 3. The pulse numbers in this buffer cover the whole measurement range from tens of laser pulses to nearly thousand of laser pulses. The measurement frequency and maximum range can be altered by change the P_n. In our experiment we set $n = 5$ and the minimum $P_i \in P_n$ is 20 and the maximum is 700. This mode requires larger communication

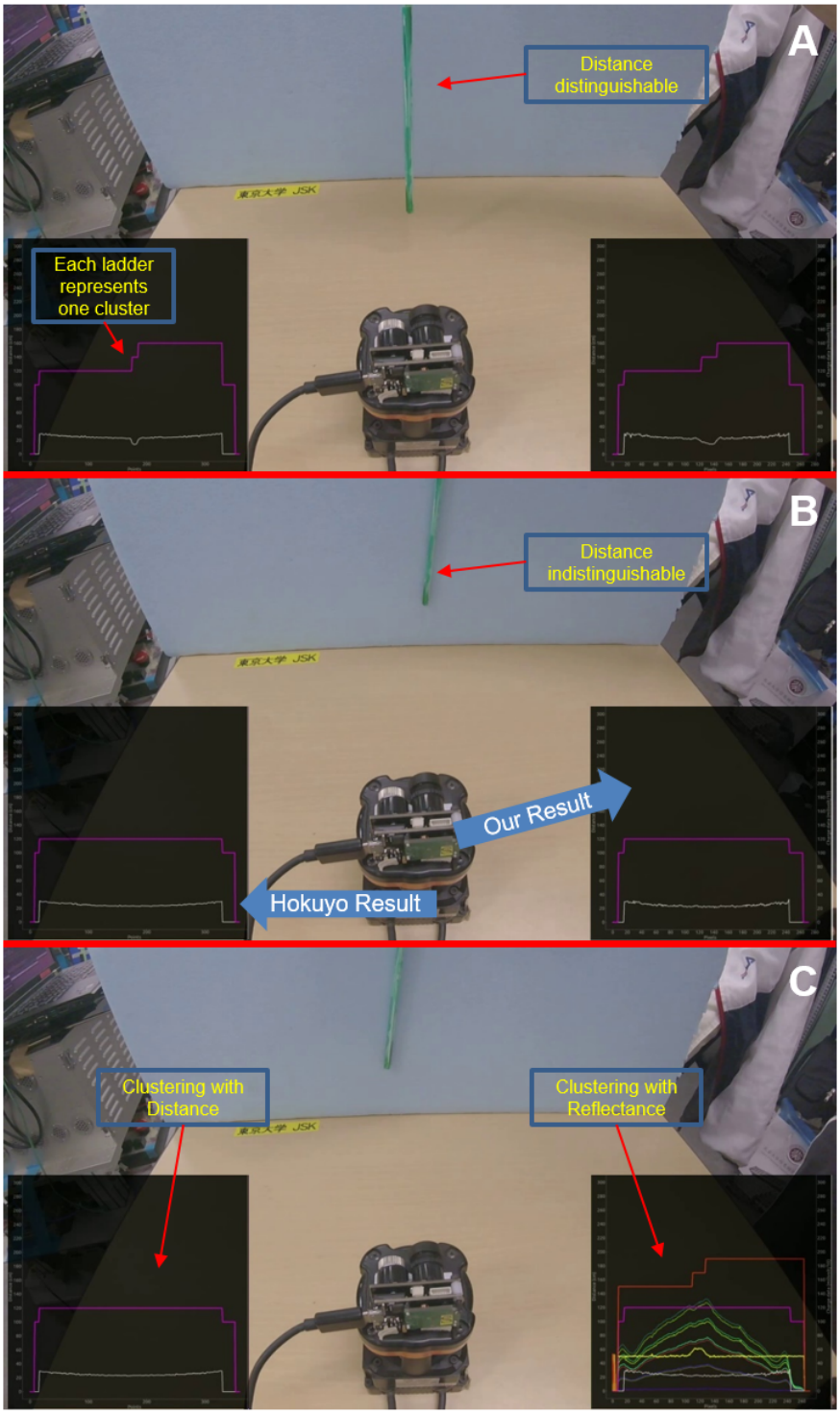


Figure 5.8: Reflectance based clustering demonstration. (In A the distance of the stick and the background is distinguishable thus both sensors can achieve correct clustering only by distance using same DBSCAN parameter; In B the distance is indistinguishable since the stick is too close to the background; C demonstrates the clustering result using the reflectance information.)

bandwidth between the sensor system and the host computer since all the nth raw data need to be sent, thus the maximum measurement frequency is limited.

With the intensity data, we performed experiments on detecting the metallic objects using both intensity and distance information that provide the orientation and position information of the target object respectively.

Algorithm 3 Framework of raw(intensity) data continuous measurement mode

Hardware Initialization:
CPU Initialization: System Clock, NVIC, DMA and System Watchdog;
Peripheral Initialization: ADC, DAC, High Resolution Timer, Common Timer, Serial Communication and GPIO.
Parameters:
High Resolution Timer channel delay buffer D_n: This buffer determines the accurate time delay among laser pulse, VTX1, VTX2 and VTX3.
Timer counter T and Timer frequency f: emission and receive once during the timer period;
Accumulation Pulse Buffer P_n: where at each $P_i \in P_n$, pulse accumulated to send the raw capacitor data from ADC;
Interrupts Vector V_n: each $V_i \in V_n$ represents an interrupt and each interrupt is bound to a callback function.

1: In the main loop, The CPU processes the raw data from ADC DMA and waits for interrupts in this mode, when the timer callback is triggered, jump to 2, when the ADC callback is triggered, jump to 3;
2: $T = T + 1$, open one shot high resolution timer under delay buffer D_n; if $T = 1$, jump to 4, if $T = P_i$, jump to 5, if $\forall P_i, T > P_i$, jump to 6;
3: ADC convert and ADC-DMA transmit completed, raw data are stored in memory and being processed in the main loop 1;
4: Initialize the receive sensor, trigger ADC to obtain the reset capacitor voltage;
5: Since $T = P_i$, the P_i pulses are accumulated, call Serial-DMA service to transmit the processed data;
6: All transmission finished, start a new measurement, $T = 0$.

5.3.2 Experiments on Local Sensing based Wrench Picking

As Figure 5.9 shows, from the laser output data we can clearly recognize the wrench position in figure 5.9(A). In figure 5.9(B), the robot slightly change the pitch of the end effector to obtain the intensity data. When the reflectance of the target surface is very high, the sensor is facing directly toward the wrench surface. Figure 5.9(C∼E) demonstrate the wrench picking, stem shaft detecting and rotating movement.

To align the gripper to perpendicular to wrench, the intensity data are monitored. The robot pitches the gripper with a small angle step and record all the intensity

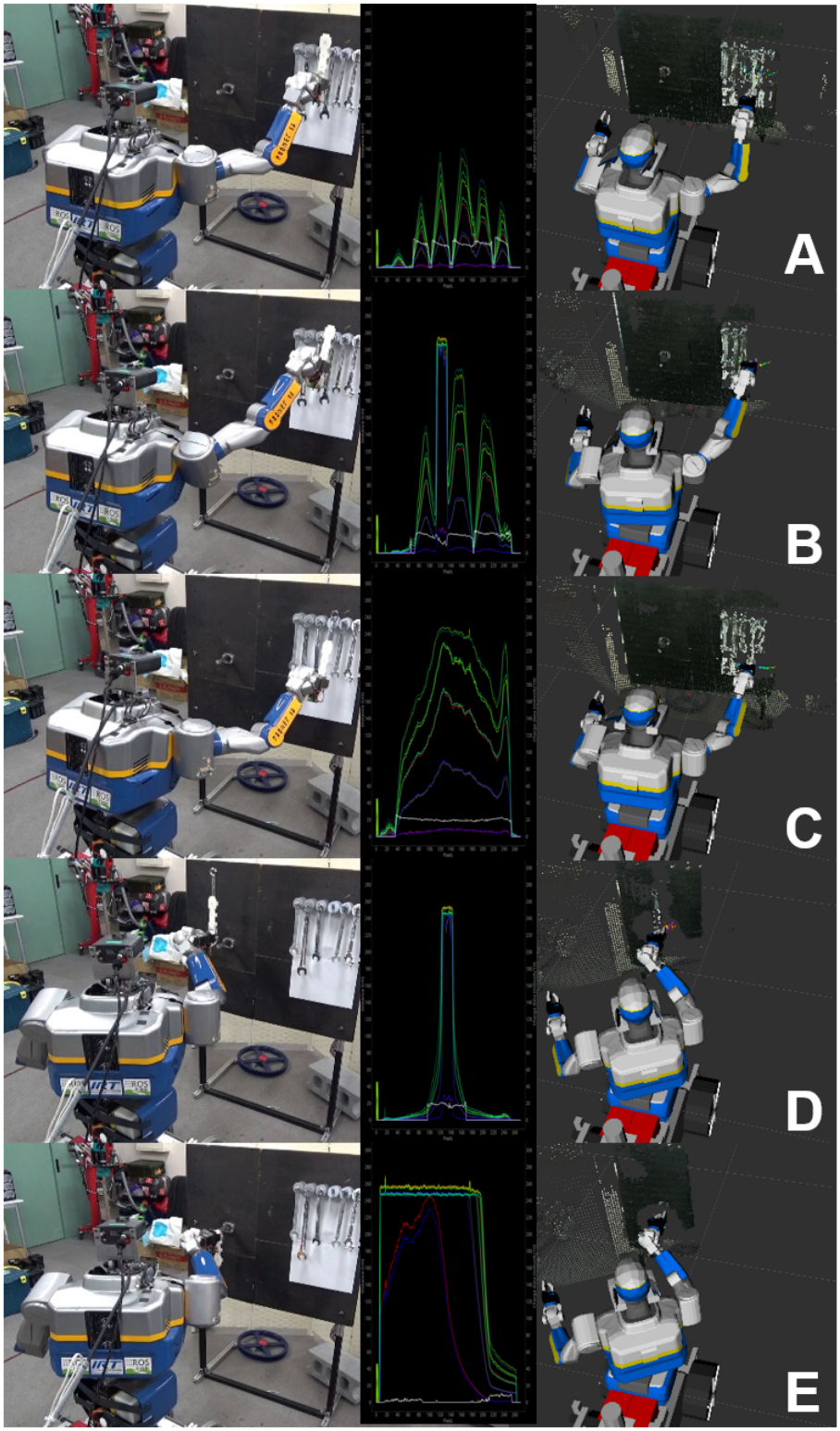


Figure 5.9: Wrench manipulation using on hand laser sensor by customized magnet gripper.(A∼E presents the wrench recognition, intensity based direction alignment, wrench picking, stem shaft detecting and stem shaft operating)

information, after pitching the angle with maximum intensity is found, which indicates the geometry perpendicular between the wrench surface and the laser sensor.

Figure 5.10 shows the experiment on wrench picking using general purpose robot hand. Both experiments share the same algorithm framework but using the general purpose robot hand is more challenging because the customized magnet gripper can easily suck the metallic wrench to the gripper while the general purpose robot hand need to align to the wrench accurately.

5.3.3 Discussion

It is a difficult problem to detect the metallic objects and obtain its range position and orientation information through fast and effective measurement. The traditional range sensors depend heavily on the incident angle to get the range information. In additional, even the contact and accurate 3D pointcloud is obtained by the non-active range sensing like stereo vision, it requires computation resources to obtain the orientation and position of the target objects as well as resolving the occlusion issue. Our proposed robot system with on hand tiny laser sensor demonstrates the its feasibility and advantages in this kind of tasks, which shows the flexibility usage of our tiny laser sensor system.

5.4 Summary

In this chapter the development of a challenge oriented robot system is introduced. Targeting at the specific problem, we design both the hardware architectures and the software algorithms. In the experiments, we demonstrate the feasibility of our systems as well as the limitations. Since in the challenge provided a lot of prior information for us to simplify the challenge, a more general problem can be addressed by our local sensing framework using the tiny laser sensor we developed that was customized to provide the intensity information. The experiments of this local sensing framework show the effectiveness of our tiny laser sensor in this general robotic task and demonstrate the customizable advantage of our sensor.

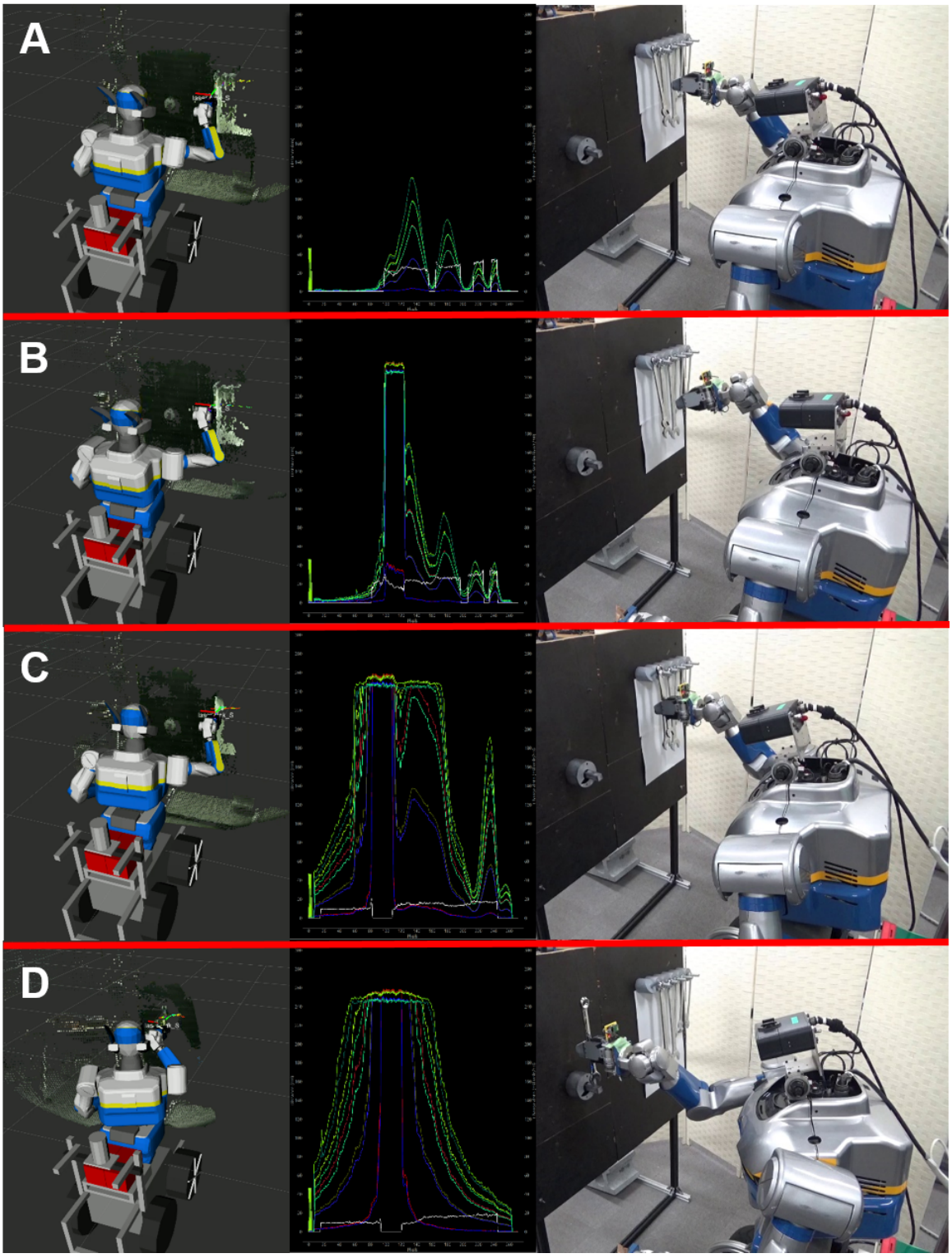


Figure 5.10: Wrench manipulation using on hand laser sensor by general purpose gripper.((A) is initial picking pose; (B) shows the robot pitches the gripper to find the orthogonal angle to the wrench surface; In (C) robot picks the wrench; (D) demonstrates the robot fit the wrench to the shaft stem.)

Chapter 6

Vision Recognition and Local Sensing Based Object Manipulation for Aerial Robot Systems

6.1 Target Detection and Picking with Vision-Based Drone Systems

Except the challenge 2 of MBZIRC, we also participated in Challenge 3. The Challenge 3 of MBZIRC requires a team of 3 UAVs(Unmanned aerial vehicle), equipped with magnetic, suction or other type of end effectors to search, find, pick and relocate a group of static and moving objects. The challenge 3 is considered as the most difficult challenge of MBZIRC since it requires multiple drones to work together and the size of the objects to be picked are only circles of $10cm$ radius which brings both complexity and precision requirements to the UAV systems.

In this section, the systems and approaches we apply in the challenge task 3 of MBZIRC are described. First we introduce the drone platform systems we build specific for this task. Next we describe the system frameworks that consist of general approaches toward this challenge. In the experiment the **Gazebo** [134] based software simulation result is showed. In the simulation, we justified our target detection algorithm and the whole task-oriented state-machine. These algorithms are then implemented in the real platform. Combined with a electromagnet gripper we designed for the drones as a end-effector to pick to object, the feasibility of the framework are

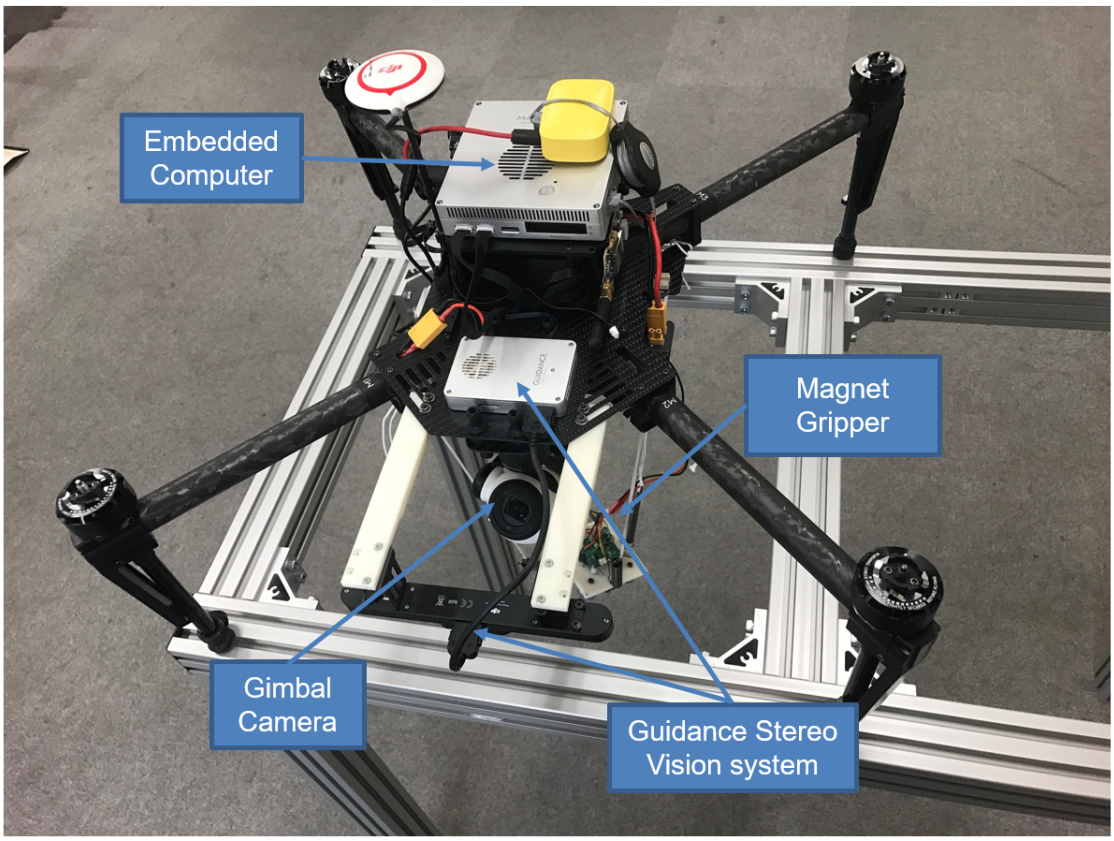


Figure 6.1: Task 3 Platform: DJI M100

demonstrated in the experiment. In the last part of this section a discussion is made to explain why our system fails in the challenge and the potential improvements.

6.1.1 Platforms

We developed two types of UAVs. We first used customized DJI M100 as our standard platform (Figure 6.1). The UAV is equipped with Nvidia Jetson TX1 and TK1 based embedded computers to run control and vision algorithms. We adopt the DJI ZENMUSE Z3 gimbal camera as our the vision sensor. The DJI M100 is accessible through the provided on-board SDK with ROS support, we can not only obtain the sensor data like GPS, IMU and barometer from the SDK, but also control the drone by simply using the API functions from the SDK. The specifications of our M100 is as Table shows.

The second type we built a multi-link based transformable aerial robot, which the prototype has been introduced in [135]. We improved the transformable aerial robot

Table 6.1: Specification of customized DJI M100 in MBZIRC Challenge 3

SPECS	Value	SPECS	Value	Sensors
Type	Quadrotor	Max Payload	$700(g)$	DJI ZENMUSE Z3 Camera, DJI Guidance, GPS, IMU, Barometer, Gripper proximity sensor
Takeoff Weight	$\approx 4(kg)$	Gripper Weight	$180(g)$	
Height	$350(mm)$	Propeller	$650(mm)$	
Diagonal Wheelbase	$650(mm)$	Motor KV	$350(kv)$	
Battery	Lipo $6s$	Hovering Time	$\approx 20(min)$	

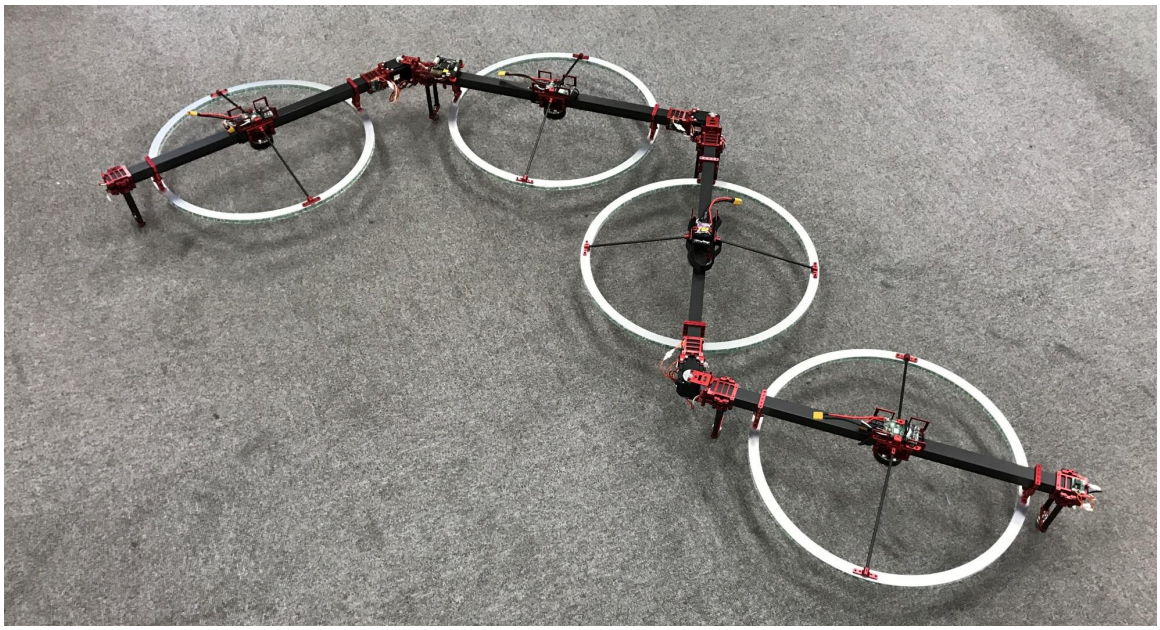


Figure 6.2: Task 3 Platform: Hydrus

called **Hydrus** (Figure 6.2) by changing the propellers to 14[inches], re-designing the mechanical parts and building the distribution circuits boards with IMU sensor on each link. Thus, the performances are improved compared to our recent work of manipulating the objects [136].

6.1.2 System Framework Design

Electromagnet Gripper design

We chose to use an electromagnet gripper based on its ease of control, and also because the circuits can be readily designed and made by ourselves. Permanent magnets are stronger but it also requires an extra mechanism to push on the object for releasing, which will make the attachment more complex, and may also require an additional motor. We also contemplated on the use of an air vacuum but the size of a vacuum is too big for our drone.

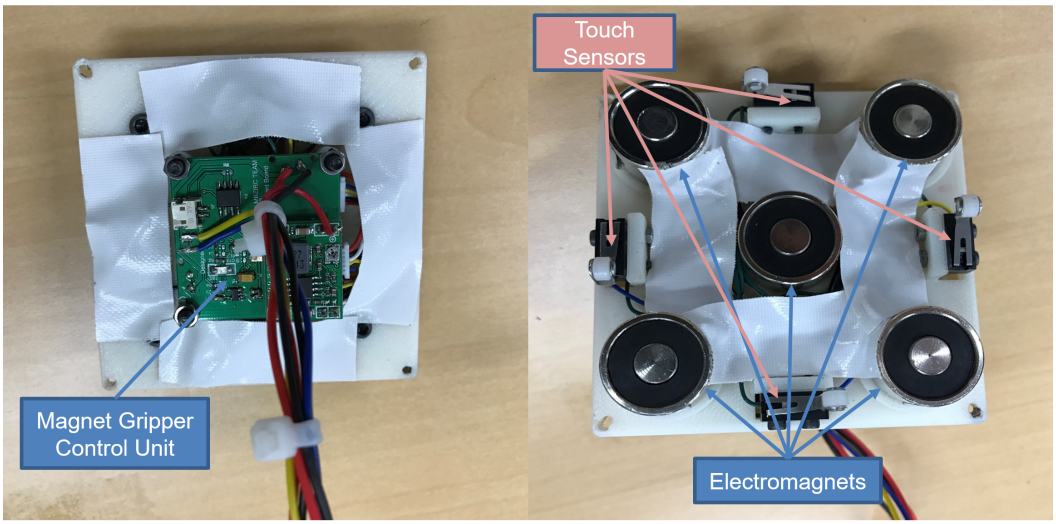


Figure 6.3: Task 3 Gripper ($8cm \times 8cm, 180g$)

We equipped 5 small electromagnets on a gripper of size $8cm \times 8cm$. Each electromagnet is capable of producing more than $20N$ of attractive force given a good object thickness. The gripper shown in Figure 6.3 consisting of 5 electromagnets can pick up an object made of iron weighing near $1kg$ with a thickness of more than $0.3mm$. The gripper is also equipped with four touch sensors to detect object–gripper contact state. In addition, these touch sensors are used to determine the position of the gripper with respect to the object.

The design of the electromagnet circuits driver is simple since it can be simply regarded as a series connection of a inductor and a small resistor. Darlington Transistor is suitable to drive an electromagnet as the current requirement is only $200ma$. Because of the inductor effect, a protection diode is necessary in the circuits to prevent backflow. We designed the circuits board with a 32 bits micro controller, a Darlington driver IC and both serial, CAN(Control Area Network) communication interface in

the gripper board. An on-board driver written as a ROS node collects the status of the magnets and the touch sensors in the gripper and reports these information to the embedded computers.

Target Detection Vision Approach

There are several limitations for drone vision processing. Firstly the sensors should be as little as possible since the flying time will dramatically reduce if the weight increase and meanwhile guarantee the quality of the images in outdoor environment. Secondly, it is impossible to have very high performance processing ability for on-board computer when considers the size and weight. Lastly the sensors that can be steadily and easily installed are preferred. Thus we adopt mono-camera as the vision sensor for our platform and use nvidia embedded on-board GPU computer TX1 and TK1 as our main processors since it is required that all the vision algorithm need to be running at real-time.

Since in our approach we decide take advantage of the global arena information, the coordinate system is under world coordinate, which introduces the projection matrix P that satisfy:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = P \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{6.1}$$

where u, v are under the image pixel plane and X_w, Y_w, Z_w are the corresponding world coordinate. According to the very basic computer vision geometry theory [137], the projection matrix P consists of several parts and can be noted as:

$$P = K\mathbf{E} = K[R|\ T] \tag{6.2}$$

where K is the camera intrinsic matrix and can be obtained by using a chess-board [138]. The $\mathbf{E}$ is the extrinsic parameter matrix and is basically a 3D rigid transformation from the camera coordinate system to the world coordinate system and consists of two separate rigid transformations which can be obtained by the gimbal control feedback and the global odometry of the drone. According to the Equation 6.1, the corresponding world coordinate of each pixel from the image plane can not be uniquely determined. We simply use the prior arena knowledge and set the height of the object to a predefined value that $Z_w = C$. Thus the 3D world coordinate of any image pixels can be acquired.

For the object detection, we tried several traditional algorithms like the shape-based hough circle transform, color-based HSI filter with clustering and feature-based edge detection. Considered the computation complexity and stability, we adopt separate HSI filters for each color of the objects and apply euclidean cluster among all the points candidates after filtering. Note sets P_r, P_g, P_b, P_y, P_o are the image pixels after HSI filter of red, green, blue, yellow and orange objects in the arena. Then all the candidate pixels after filtering P_{all} is:

$$P_{all} = P_r \bigcup P_g \bigcup P_b \bigcup P_y \bigcup P_o \tag{6.3}$$

then $\forall p_i, p_j \in P_{all}$, clusters $O_i = \{p_i \in P_i\}$ and $O_j = \{p_j \in P_j\}$ are obtained by:

$$min||p_i - p_j|| \geq d_{threhold} \tag{6.4}$$

the pixel distance threshold is proportional to the distance from the UAV to the ground which can be found in the projection matrix P since the distance resolution of two pixels increases when the camera is leaving from the ground. Thus by removing some of the clusters with little pixels, we obtained the valid clusters $\bigcup O$. The centroid of each clusters and the color domain are then computed that $\forall O_i$,

$$[Point(u_i, v_i), C_i] = \mathcal{F}(O_i) \tag{6.5}$$

where the 2D points in image plane can be projected into the 3D world coordinate thus the global position and color domain $G_i$of each detected object cluster O_i is obtained:

$$G_i = [Point(X_i, Y_i, Z_i), C_i] \tag{6.6}$$

where Z_i is predefined with respect to the height of the objects. Then a filter is assigned to all the G_i to remove the error detection and noise. The algorithm finally assigns a rank to the G_i after filtering to find the most stable target and pass the information to the motion executors.

State Machine

The state machine of this challenge is divided into 3 big states of searching, picking and placing. Figure 6.4 briefly shows the flowchart of the state machine. Firstly the searching areas are defined with respect to the UAVs we use. In each area the predefined or random generated way points are created offline. After taking off each

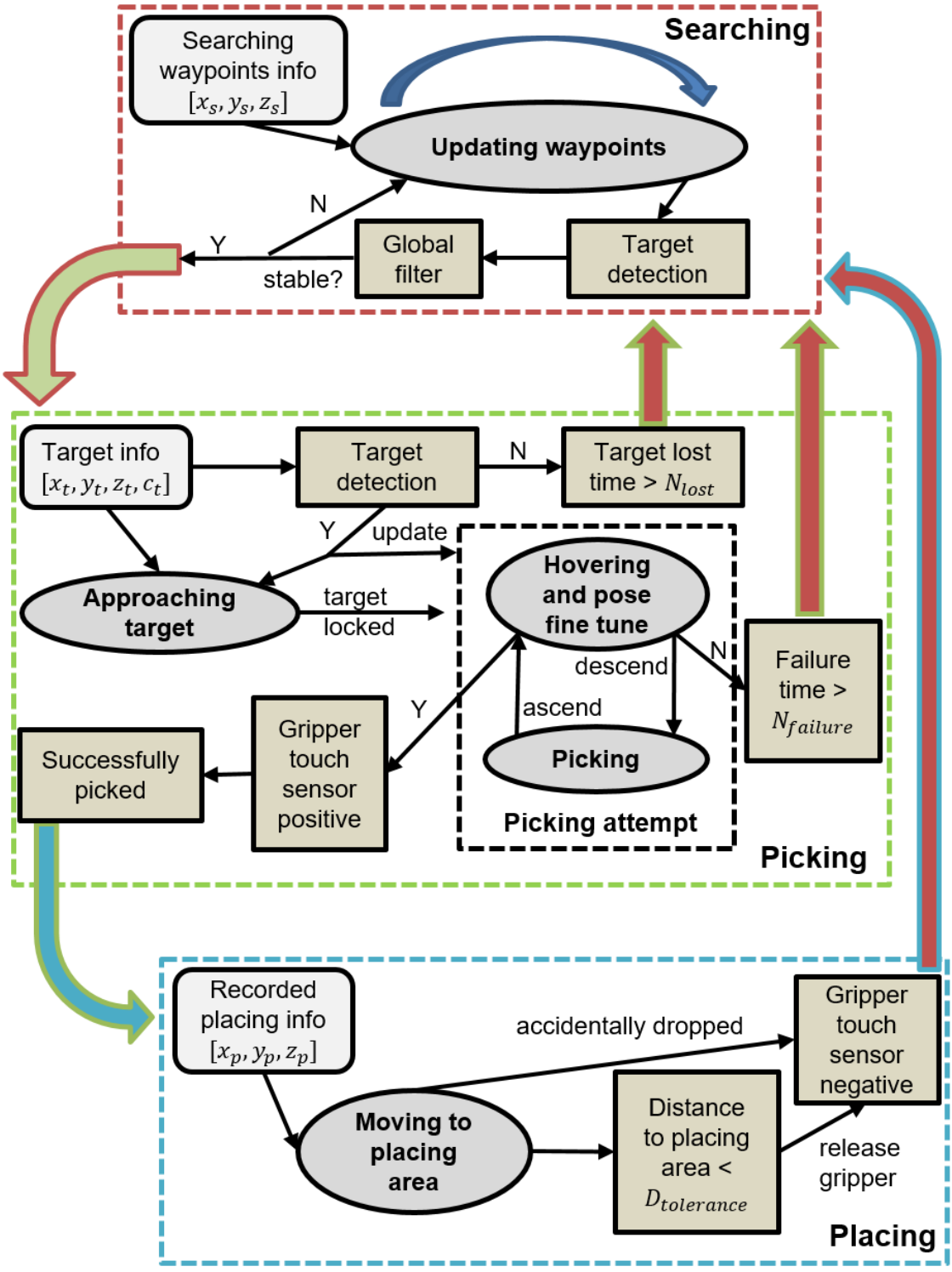


Figure 6.4: Task 3 State Machine. (Note that waypoints information, target information and placing area information are under world coordinate and c_t is the color domain of the target)

UAV follows the way points to search the whole searching area until the detection algorithm detect and find a stable target, then the UAV turn into picking state. The controller drives the UAV to approach to the target object and make attempt for picking. If successfully picked by reading the state of the touch switches on the gripper the UAV turn into the placing state. If it fails until certain times picking trials, it will give up picking and return to searching state to search for a different target object.

6.1.3 Experiments

GAZEBO Simulation

The real size arena ws created in GAZEBO simulation environment with objects of different color. We adopted the hector quadroter ROS package [139] as our simulation UAV agent and the physical interaction between the objects and UAV agent was generated by our simulation plugin node. A RGB camera sensor plugin was also created to obtain the real-time environment data in simulation. Then we implemented the vision detection algorithm, state machine and PID controller in the simulation and achieves results of about 10 static objects were successfully picked and placed into the dropping box within 15 minutes full autonomously by single UAV. Figure 6.5 demonstrates the simulation framework.

Teleoperation Results

We performed experiments using the aforementioned hardware equipped onto the UAVs through teleoperation to validate our platform. Approaching the object for grasping is relatively difficult due to rapid visual changes. A more crucial problem when approaching the object is the ground effect which results in unexpected behaviour and the UAV becomes unstable very quickly, which makes it very hard to control when the UAV is near ground. Consider this, we suspended the magnet gripper with ropes and move the UAV to a certain height that can hover stably, then once the target was locked, the UAV descend immediately to make picking attempt. On average we can pick 5 static objects within 8 minutes by single UAV as Figure 6.6 shows.

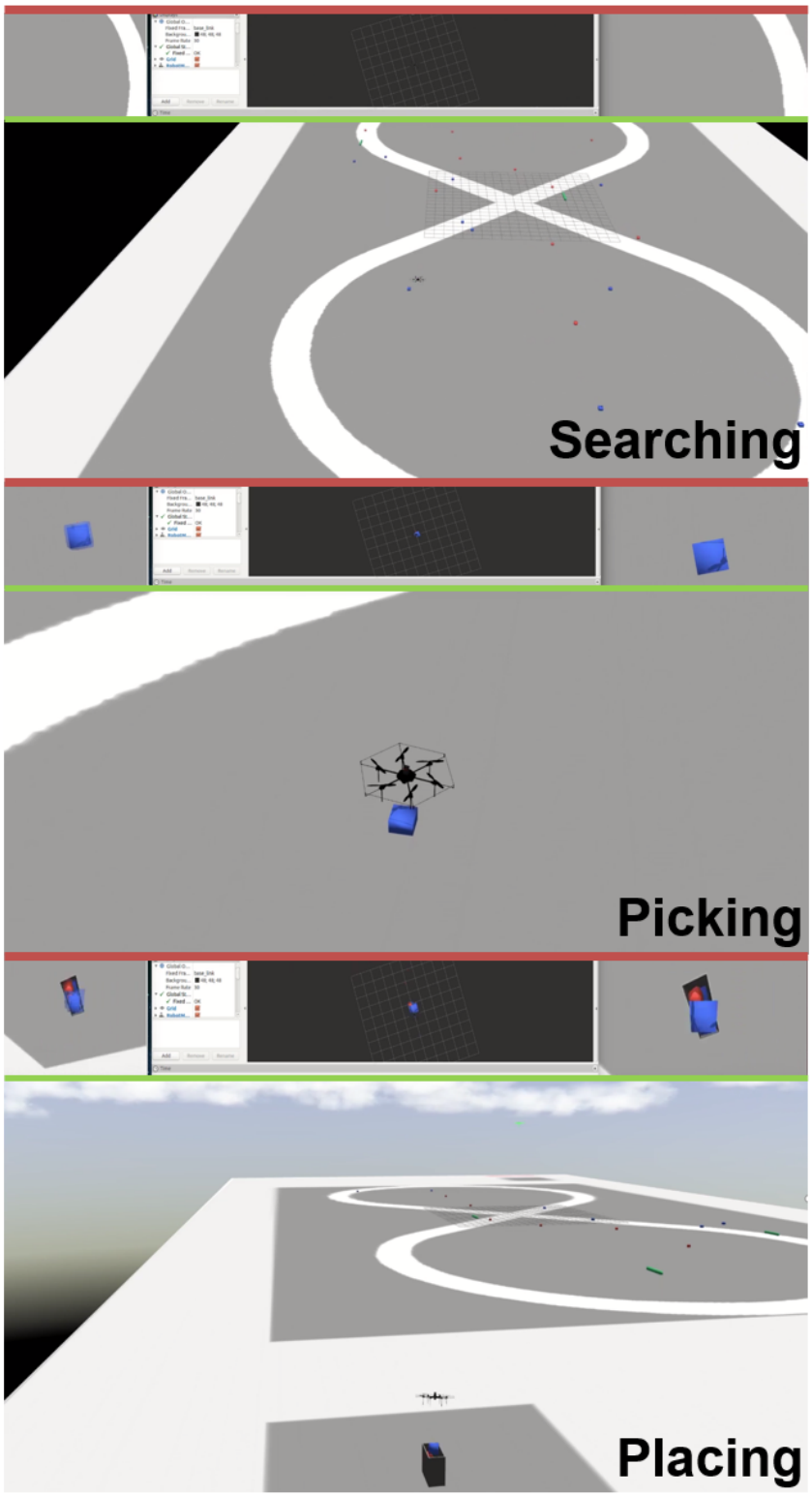


Figure 6.5: Challenge 3 GAZEBO Simulation. (The top part of each figure shows the data of camera, projected 3D points and camera narrow view)

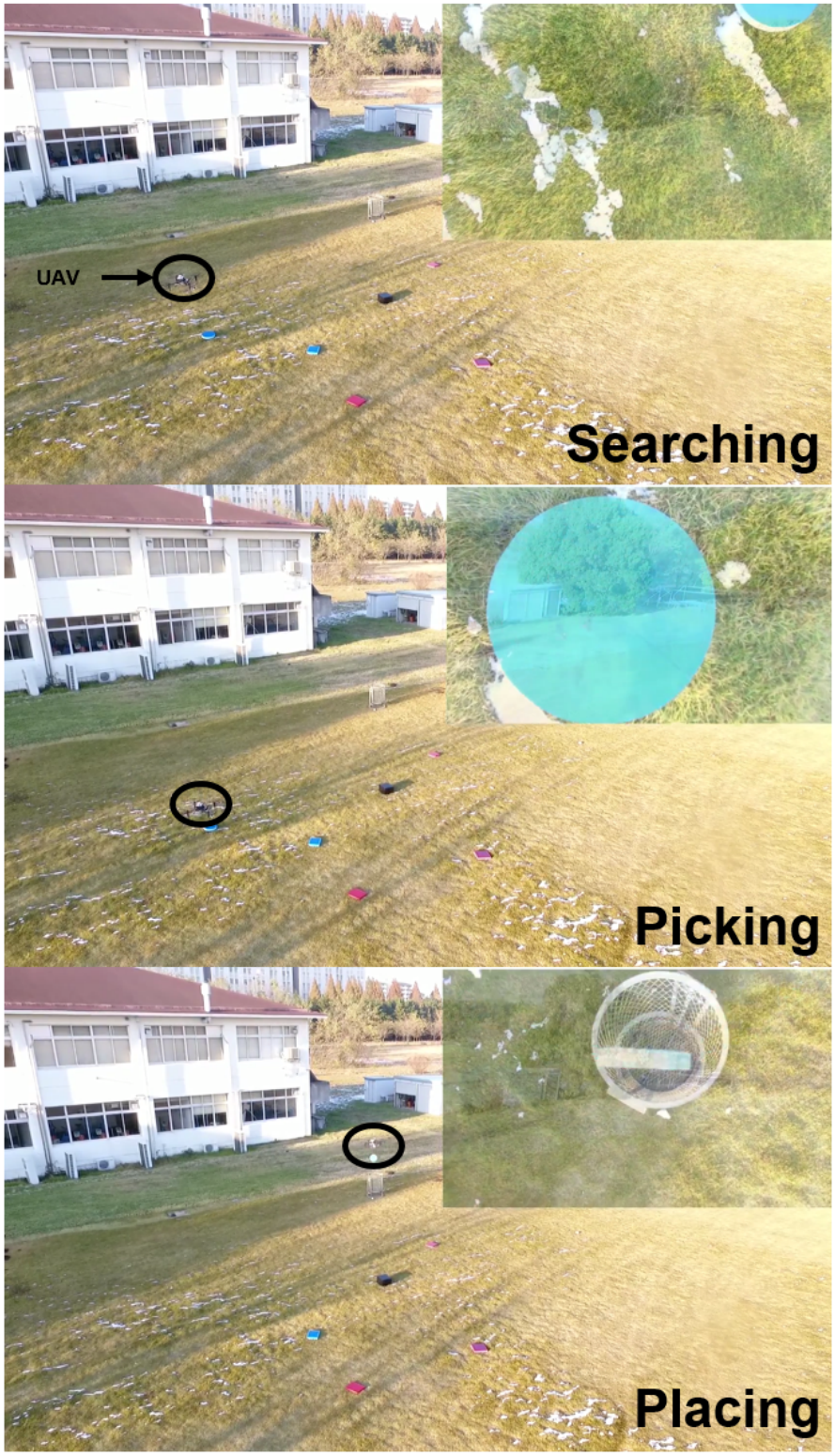


Figure 6.6: Pick, Place and Search in Teleoperation

6.1.4 Discussion

The simulation of task 3 was finished about half year before the challenge. The hardware platform and UAV controller are big problems for our teams. We were planning to apply multi-link UAV **Hydrus** in challenge 3 at the beginning but the challenge rule finally changed and the objects turned out to be a very thin iron plate which makes it impossible to use such methodology. For DJI M100 UAV, although both the vision algorithm and the state machine works at the final challenge, the successful picking rate can not be guaranteed due to several reasons. Firstly the DJI API velocity control relies heavily on the sensor data accuracy from DJI guidance and GPS, which are not precise enough for such accurate requirement. Secondly the FOV(field of view) of our camera is not so wide as we realized in the final stage. Lastly the unexpected wind and sensor data error brought bad luck to us in the final challenge.

6.2 Multi-link Aerial Robot Whole-body Object Manipulation Using Lightweight Tiny Laser Line Sensor

As described in Chapter 5, challenge 3 requires a UAV to detect and pick several small objects and then place them into a given box. We applied the tradition quadcopter drone in the challenge based on vision approach. According to our experiments we found out the biggest difficulty of this challenge is the control accuracy of the UAV when the UAV is approaching to the target object due to the ground effect. Also we consider that for some common aerial manipulation tasks, manipulating from the top demands very accurate hover control ability or a very exquisite end effector to compensate to the ground effect. Our recent work [136] shows another novel perspective toward aerial manipulation. The use of multi-link transformable UAV can manipulate the objects from side and can transform by changing links to adjust to the shape of the objects, which avoids some of the mentioned problems that manipulating from top.

For object detecting, we already proposed a vision based approach in Chapter 5 from top view. However, manipulation from side requires the side information of the target objects so that the UAV can adjust itself to approach and hold the objects. It is very important for the UAV to obtain the side shape information of the objects and

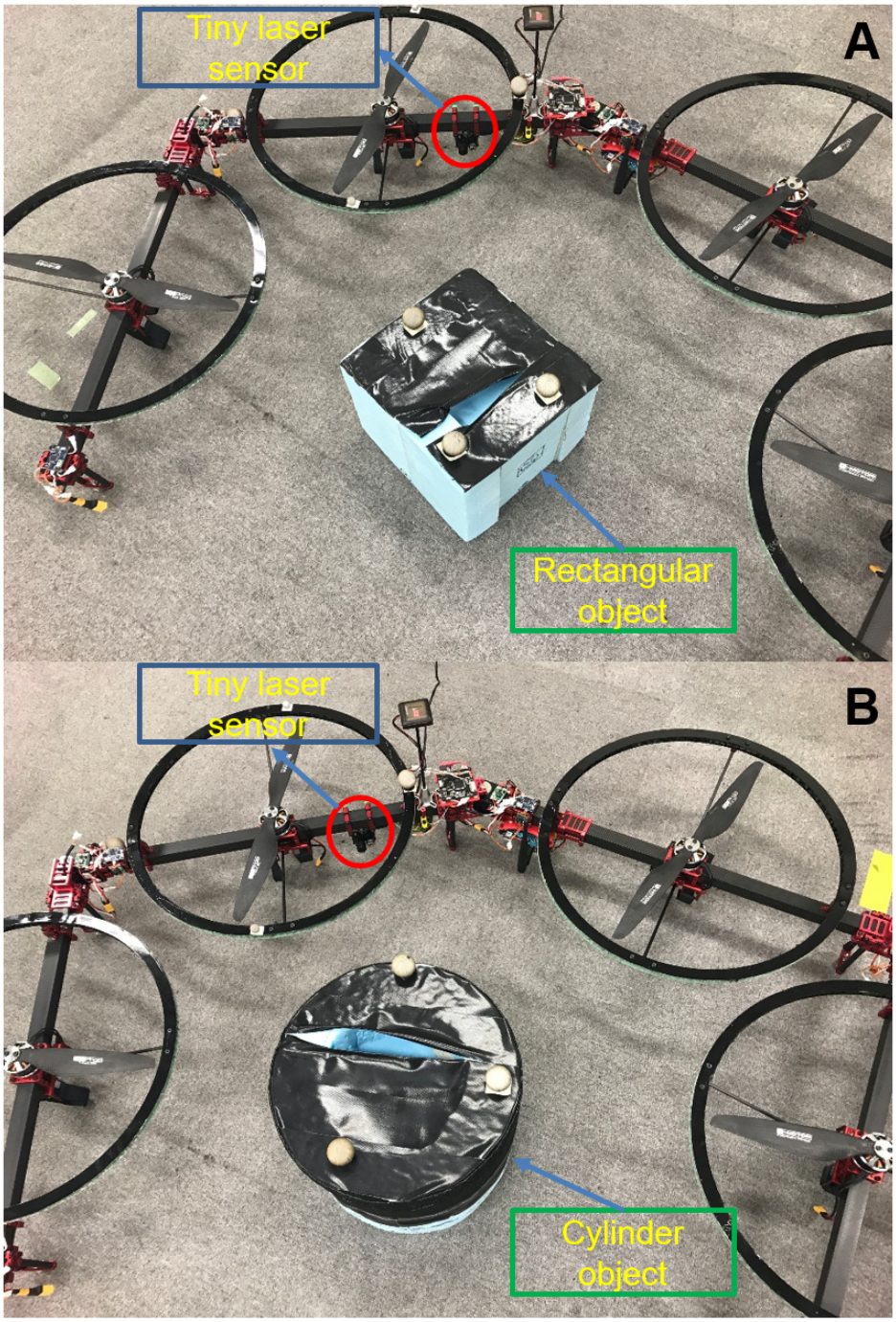


Figure 6.7: Multi-link transformable aerial robot Hydus manipulation demonstration using ultra-tiny line laser sensor. (A shows a manipulation of a rectangular object and B refers to cylinder object, the data output is shown in Figure 6.8)

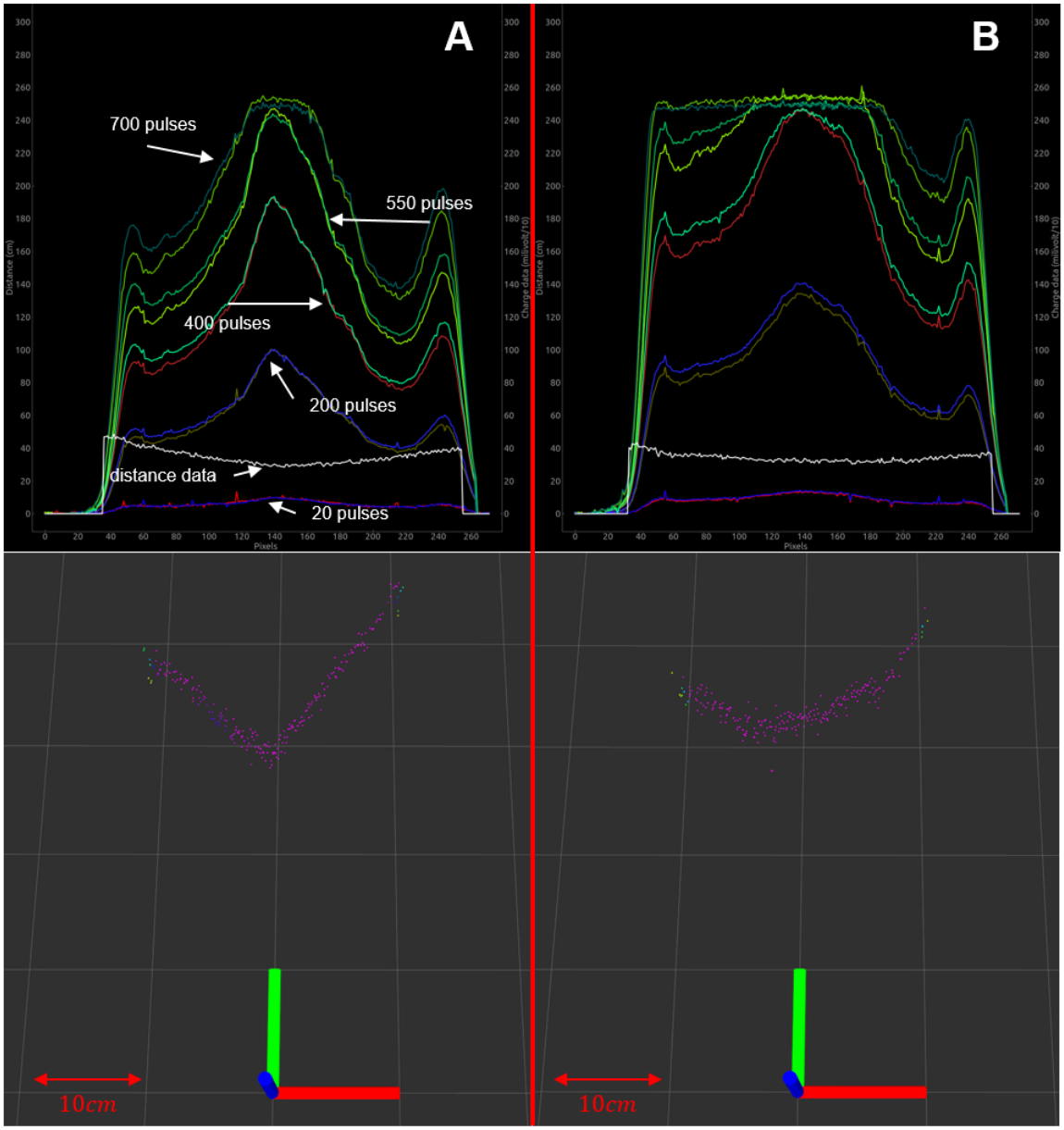


Figure 6.8: Laser data plot of aerial manipulation task for different objects. (The A and B figure correspond to the A and B pictures in Figure 6.7; The grid size is 10[cm]; The colourful plot lines are the capacitor raw charge data with different emitter pulses, the white line refers to the distance output;)

also the corresponding distance of the objects. Thus local sensing is very necessary for this task. The use of our ultra-tiny line laser sensor can provide dense line range data accurate enough to recognize the shape of object and its small size, light weight and low power consumption makes it very suitable for this approach.

Figure 6.7 illustrates the idea of installing tiny laser in multi-link UAV to realize whole body manipulation. Figure 6.8 demonstrates the ultra-tiny line laser data output of detecting different shape objects. In figure 6.8, the top plot is the raw data output, including the discharge amount of two capacitors of different emit pulses. As the pulses increase, the discharge amount rises and if the reflectance is very high or the objects are very close, the discharge reaches saturation. The distance is calculated by the ratio of the discharge amounts of the corresponding two capacitors at same pulses.

6.2.1 Multi-Sensors Framework with Sequence Measurement Mode

For the multi-link aerial robot, it is possible to mount several tiny laser sensor on different links to measurement the side information of the target manipulation objects. However, since the basic theory of out tiny laser sensor is based on the accumulation of the incident light with corresponding wavelength, interference happens when the FOV intersects. To address this problem, sequence measurement and synchronization of the sensors are necessary. Thus a new algorithm framework of our tiny laser is developed. As Algorithm 4 shows, the multi-sensors sequence measurement mode is very similar to the continuous measurement mode 3 except it receives indications through serial port from the synchronizer in host computer, during the timer interrupt 3, the operation of laser emitter switches between on and off according to the received indications. However, this mode will lead to the measurement frequency to reduce by n times when n laser sensors are used.

6.2.2 Sampling based Object Detection using Whole-body Laser Sensing Data

Attribute to the multi-sensors sequence measurement mode, it becomes possible for the multi-link aerial robot **Hydrus** to install multiple tiny lasers at the same altitude so that the surface information of the manipulated objects can be obtained. Then, according to the pre-defined or pre-detected model, the robot can fit the laser points

Algorithm 4 Framework of multi-sensors sequence measurement mode

Hardware Initialization:

CPU Initialization: System Clock, NVIC, DMA and System Watchdog;

Peripheral Initialization: ADC, DAC, High Resolution Timer, Common Timer, Serial Communication and GPIO.

Parameters:

High Resolution Timer channel delay buffer D_n: This buffer determines the accurate time delay among laser pulse, VTX1, VTX2 and VTX3.

Timer counter T and Timer frequency f: emission and receive once during the timer period;

Accumulation Pulse Buffer P_n: where at each $P_i \in P_n$, pulse accumulated to send the raw capacitor data from ADC;

Interrupts Vector V_n: each $V_i \in V_n$ represents an interrupt and each interrupt is bound to a callback function.

Laser On/Off flag O: enable and disable the laser emitter to remove interference.

1: In the main loop, The CPU processes the raw data from ADC DMA and waits for interrupts in this mode, when the timer callback is triggered, jump to 3, when the ADC callback is triggered, jump to 4, when the serial receive callback is triggered, jump to 2;
2: Read laser On/Off indication O from the host computer, if $O = 0$, disable timer counter; if $O = 1$, enable timer counter and set $T = 0$;
3: $T = T + 1$, open one shot high resolution timer under delay buffer D_n; if $T = 1$, jump to 5, if $T = P_i$, jump to 6, if $\forall P_i, T > P_i$, jump to 7;
4: ADC convert and ADC-DMA transmit completed, raw data are stored in memory and being processed in the main loop 1;
5: Initialize the receive sensor, trigger ADC to obtain the reset capacitor voltage;
6: Since $T = P_i$, the P_i pulses are accumulated, call Serial-DMA service to transmit the processed data;
7: All transmission finished, start a new measurement, $T = 0$.

to the defined model and align itself to manipulate the object. This local sensing based framework overcomes the limitation of only use global vision sensor for all range detection which inevitably fail when the robot need to manipulate the object at a very close range. In addition, the real time data flow provided by our tiny laser can provide indispensable information for the robot to manipulate the dynamic objects and even some of the deformable objects in the future.

To fit the points to the given primitive models, the sampling based approach is used. The algorithm depends on the primitive mathematical expressions like the circle, rectangular, triangular and etc in 2D space, sphere, box, cylinder and etc in 3D space. In our application, we mount the laser at the same plane thus the laser distance data can be regressed to 2D expressions.

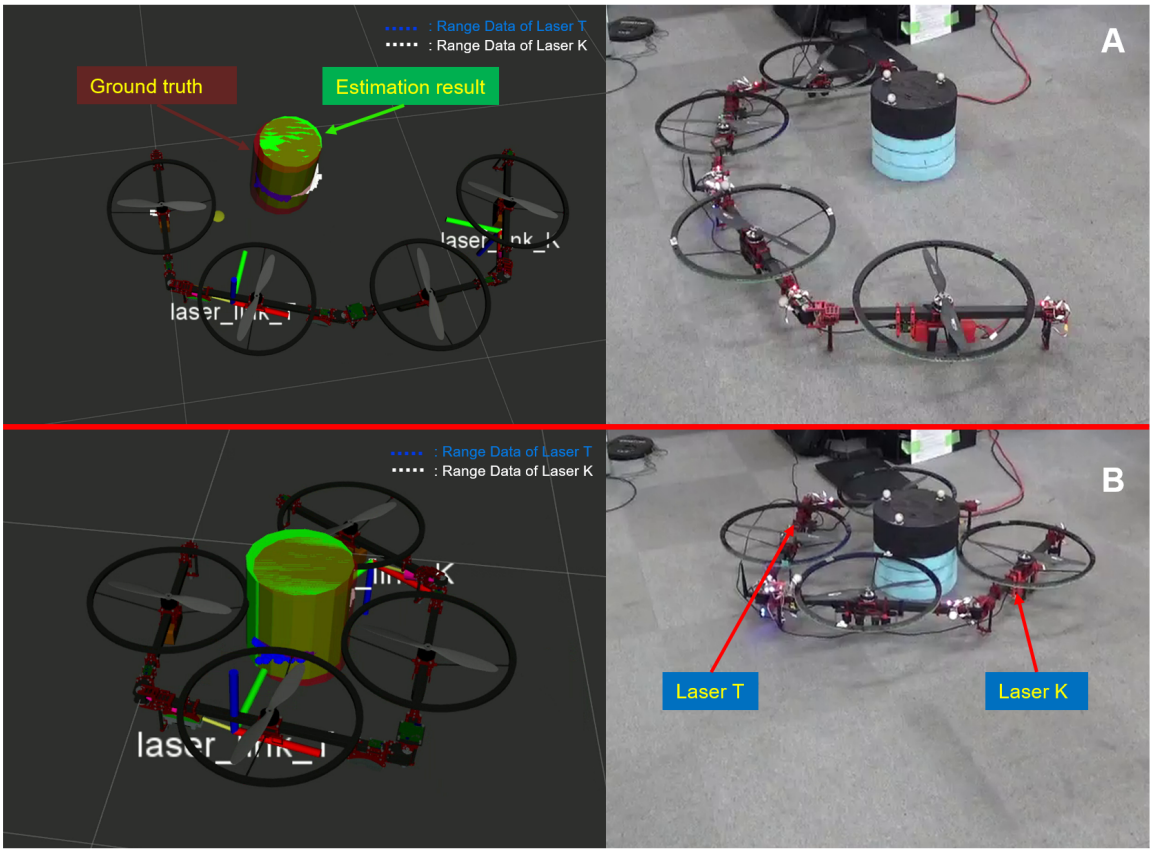


Figure 6.9: Demonstration of object local sensing using multiple laser sensor for multi-link aerial robot manipulation. (The red and green cylinders are the ground truth and estimated result by motion capture and local sensing respectively, note here height is given; The white and blue laser range points on the side of the cylinder are acquired by different laser sensors.)

We take cylinder object as an example in the manipulation task. The laser measurement of the cylinder should be a number of points in $\mathcal{S}^2$ space with each points satisfies the following circle equation:

$$||(p-c)^T(p-c)|| = r^2 \tag{6.7}$$

where c is the center of the circle and r is the corresponding radius. Thus for the points of the circle primitive:

$$\forall p_{i,j} \in (\mathcal{S}^2 \cap \mathbb{R}^2)$$

we have:

$$||(p_i-c)^T(p_i-c)|| = ||(p_j-c)^T(p_j-c)|| = r^2 \tag{6.8}$$

As mentioned in last section that every two distinct points provide one linear equation of 3 unknowns in our case, it requires 3 points to obtain the center coordinate and the radius [118].

6.2.3 Experiments

In the experiments, we demonstrate the laser configuration framework and the cylinder estimation result in Figure 6.9. Multiple tiny line laser sensors are installed at different link of the robot and the range data of different laser sensors are shown in different color. The ground truth of the object is the red cylinder which detected by the indoor motion capture system. Green cylinder is the circle estimation result with a given height. We also set a constrain filter to reduce the wrong estimation since when the robot transforms, some of the range points may reach to the robot itself and produce a primitive candidate. These points can be removed by considering the model of the robot.

Figure 6.10 shows the aerial robot manipulation demonstration. Based on the range data from multiple sensors, our algorithm provide the center and the radius parameters of the target cylinder object. The robot then adjusts its joints to manipulate the objects according to the updating parameters. Figure 6.10 (A) shows the robot open its links and approaches the cylinder object and Figure 6.10 (B) demonstrates the robot closes its link to hold the objects.

In Figure 6.11, we demonstrate shows the local sensing based multi-link aerial robot manipulation experiment. This experiment explain the whole framework of multi-link aerial robot object manipulation. Firstly the initial coordinate is given to the robot using vision sensor(here we give the approximate position by motion capture) and then the robot take off and fly to the object 6.11 (A). Secondly the robot transform to pretend the holding based on the multiple sensors installed on the link 6.11 (B). Figure 6.11 (C) shows the estimation of the object and the robot carrying the object to the dropping box. Lastly after dropping the robot check if the object is successfully dropped by checking the tiny sensor data and return to the take off position 6.11 (D).

6.2.4 Discussion

Applying the multi-link aerial robot to manipulate the object using whole-body control is a very challenging task. In our previous work [136] we only addressed the control configuration of the robot while the information of the object is generated

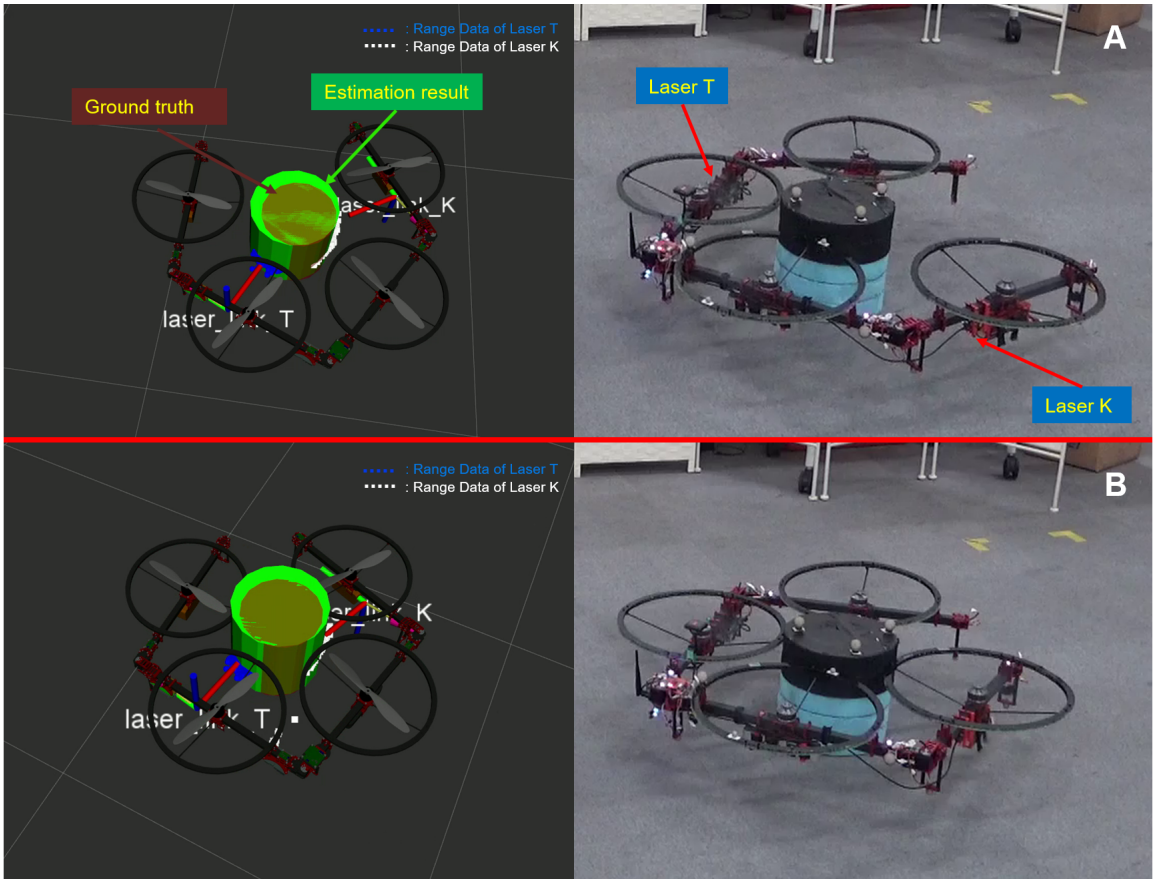


Figure 6.10: Manipulation demonstration. (From (A) to (B) the robot approaches and holds the object; The white and blue laser range points on the side of the cylinder are acquired by different laser sensors.)

by the accurate motion capture system which requires markers on the surface of the object. However, in the real application, the complete sensing system is very necessary for the multi-link aerial robot and local sensing are extremely important in our case. The use of our tiny line laser perfectly satisfies the request and the multiple sensor measurement configuration makes it possible for the **Hydrus** to realize the whole-body manipulation tasks without any external vision systems.

6.3 Summary

In this chapter, we describe the vision recognition and local sensing based object manipulation approaches for aerial robot systems. Firstly the global vision object detection part is illustrated. The feasibility and effectiveness are demonstrated through the Gazebo simulation and MBZIRC challenge 3 task using common UAV platform

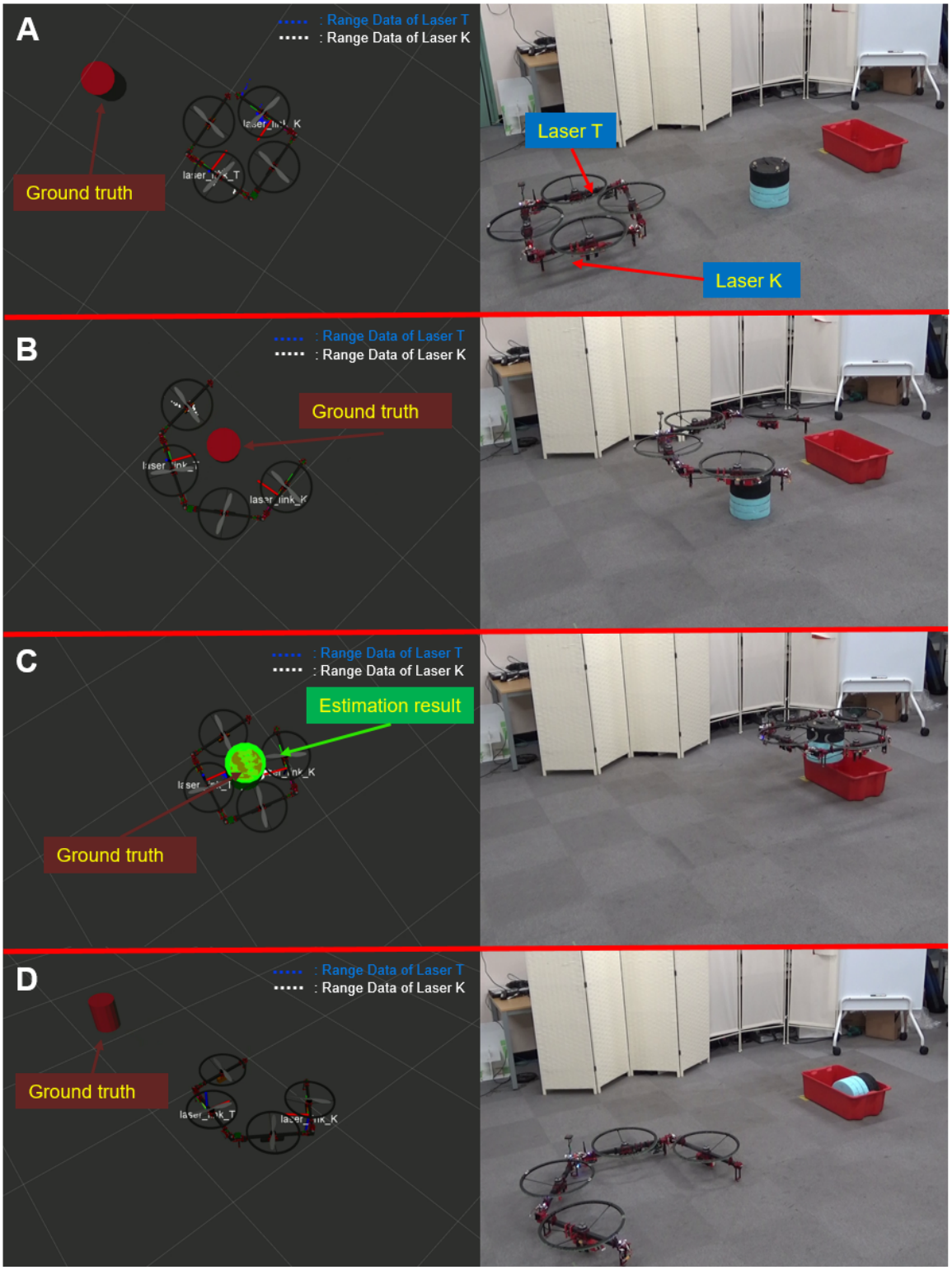


Figure 6.11: Whole-body aerial robot manipulation experiment. (From (A) to (D) the robot take off, hold, carry and release the manipulated object; Note that in D the ground truth of the cylinder object is the last frame when the object can be observed by the motion capture system, namely the frame before being dropped in the box.)

DJI M100. However, considering the control difficulties when the UAV approaches the object near ground, we also design a multi-link aerial robot to manipulate the object from side instead pick from top. As the side surface information are very important for the multi-link aerial robot when performing manipulation, a local sensing based framework using out tiny laser line sensor is proposed. The result justifies the unique effectiveness of the tiny laser for UAV tasks where the lightweight, flexibility and customizable performance are primarily considered.

Chapter 7

Conclusion and Perspectives

This dissertation addresses perceptual incompleteness issue which is considered as a pervasive property of most autonomous robot systems. We develop our novel tiny laser line sensor system with both hardware and software design which can be used as a multi-purpose range sensor to improve the sensing flexibility of robot system with respect to its small size and light weight. Next a reasoning based vision recognition approach is proposed since we discover that with very common prior knowledge of the world like the physics, geometry and functionality, it is possible for the robot to make an approximate and reasonable guess of "what is where" from the incomplete sensory data. We demonstrate this idea by handling the very difficult tomato harvesting task and the effectiveness of combination of both methodologies to handle perceptual incompleteness are also claimed when the laser perform as a proximity sensor to justify the initial guess. By building the task-oriented robot systems for the international robotics challenges, the effectiveness of local sensing applications using tiny laser line sensor are evaluated.

Chapter 3 describes the development of the a lightweight tiny laser line sensor with the details of both hardware and software. Several versions of circuits design, hardware PCB and lens attachment are upgraded to achieve the temporary version. We create a thorough model for the wholes system and proposed our hardware and software calibration methods and addresses the unsolved problem of the laser sensor. In Chapter 4 we document our reasoning and active local verification based vision recognition approach in solving the very difficult tomato harvesting task. The idea that using the common world prior knowledge to handle perceptual incompleteness is illustrated and justified. Chapter 5 and 6 document how we build the task-oriented robot systems based on flexible local sensing in robotic challenges and the advantages of flexible local sensing in handling these real world tasks is explained. Thus the

feasibility and effectiveness of our laser line sensor used as active local sensing purpose in above task applications are illustrated.

7.1 A Low-cost Lightweight Tiny Laser Line Sensor for Robotic Applications

The advantage of our design is that we keep the sensor tiny in size and light in weight without sacrificing measurement accuracy. This is attributed to the high performance circuit modules we use for calibration. We model errors in the sensor system and propose an efficient and intuitive algorithm to calibrate the system according to our new calibration model that consider the light source effect. According to the experiments, our sensor achieves measurement biases and repeatable accuracy of less than $2[cm]$ under certain condition, which is acceptable for a lot of range sensing applications. The small size, low cost, and low weight make this unconventional sensor a valuable tools for robotic applications. The sensor is very suitable for robotic local sensing applications which require the size, weight and flexibility of sensor.

7.2 Tomato Harvesting Robot System with Reasoning and Active Local Verification based Vision Approach

For the reasoning based vision part, the basic idea comes from the observation of the natural laws of physics and is unique and novel since the robot compute reasonable guesses, which resembles more of human behaviour. After obtain the guesses using the reasoning method, we show how our tiny laser line sensor can be used in the final approaching step of harvesting, which verifies and increases the certainty of the results. In the experiment we showed the feasibility and effectiveness of our approach. Our trials of applying humanoid robot for agriculture harvesting which targets at this very difficult problem demonstrate the feasibility of our method by experiments and shows another interesting way of robot harvesting. In addition, based on the laser range data, a verification framework is applied to reduce the vision recognition and reasoning error.

7.3 Task-oriented Robotic System Applications Based on Active Local Sensing Using Lightweight Flexible Tiny Laser Line Sensor System

We demonstrate several task-oriented robotic applications that based on active local sensing using our light weight tiny laser line sensor. we developed both the hardware architectures and the software algorithms of the robot systems. While dealing with the tasks like metallic tools grasping and multi-link aerial robot object manipulation, we explain the difficulties and challenges of these tasks and how the robot system could be altered handle them. In the experiment, attribute to the flexible and customizable design of the sensor, it become possible to have an unconventional but effective sensor solution to overcome such difficult challenges.

7.4 Contributions

This dissertation has presented several contributions: First, a low-cost lightweight flexible tiny laser line sensor is developed. The hardware design and software calibration development make this sensor flexible and customizable for robotic applications. Second, a reasoning and local sensing based vision recognition approach is proposed in tomato harvesting robot system, Combine with the tiny sensor we develop as a active local verification sensor, we show the feasibility of selective picking for tomato, which has been considered as a very difficult robotic task. Third, the advantages of our tiny laser line sensor as well as the development of these task-oriented robotic systems toward robotic challenges are illustrated in several task-oriented robotic applications including the metallic tools grasping for humanoid robot and multi-link aerial robot object manipulation.

7.5 Future Work

There are several areas for improvement. For the tiny sensor development, the frequency can be increased by installing more laser emitter since the main limitations for this sensor now is the duty rate of the laser emitter. As all the PLDs(Pulse Laser Diode) require enough time to cool down after incident, mounting multiple PLDs and divide the emission by time so that the receiver accumulation time can be increased. It is possible to increase the frequency to several hundred Hz while uses the similar

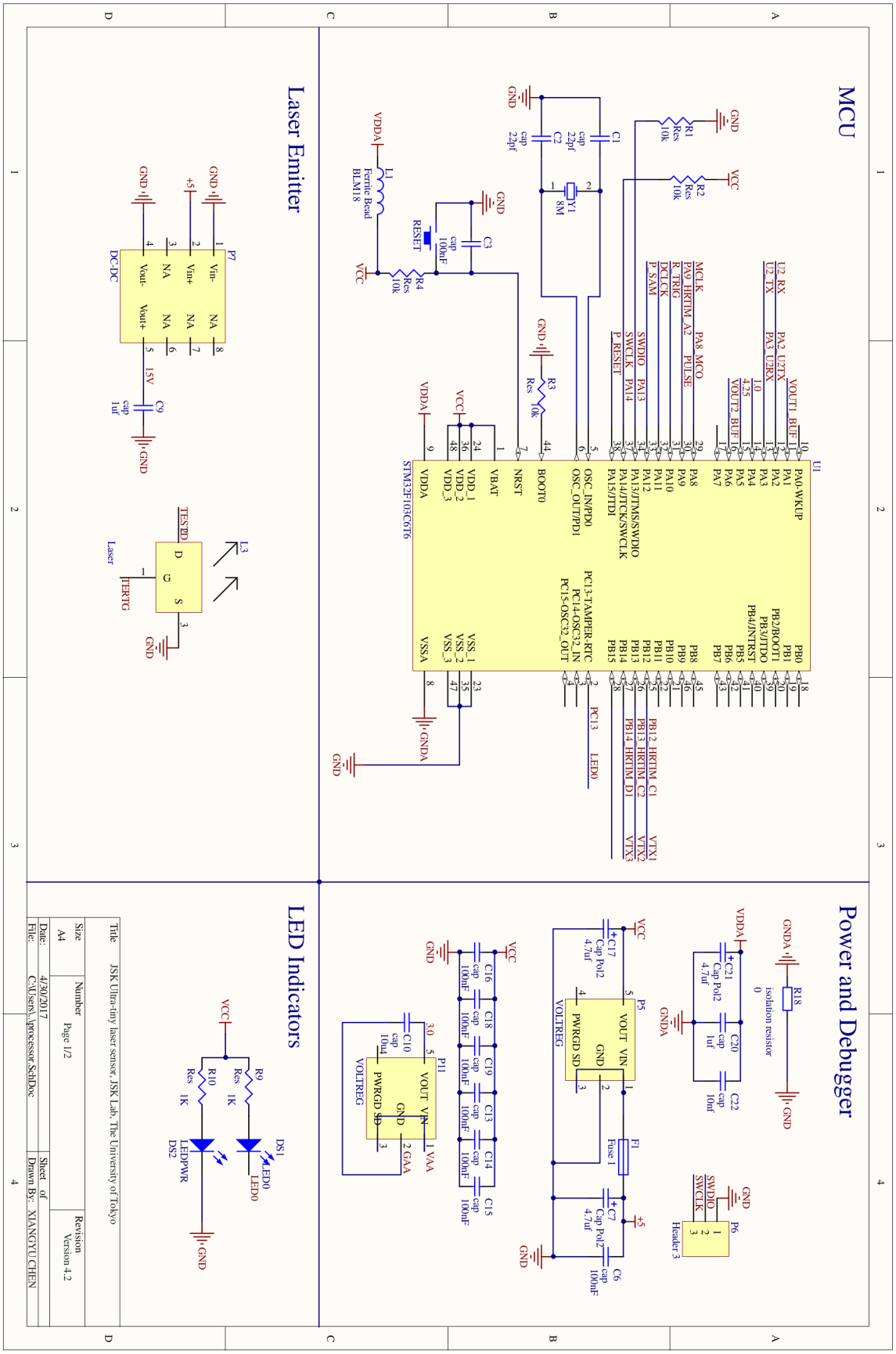
hardware design and software configuration of our sensor, which will enable the robot to handle tasks that requires high speed sensing rate. For the active local sensing based task-oriented applications, we are trying to apply this tiny laser line sensor to a lot of traditional tasks such as biped humanoid terrain real-time scanning, mechanical prostheses slope detection and tiny mobile robot(Darwin) range sensing, etc.

Appendix A

Laser Sensor Design Details

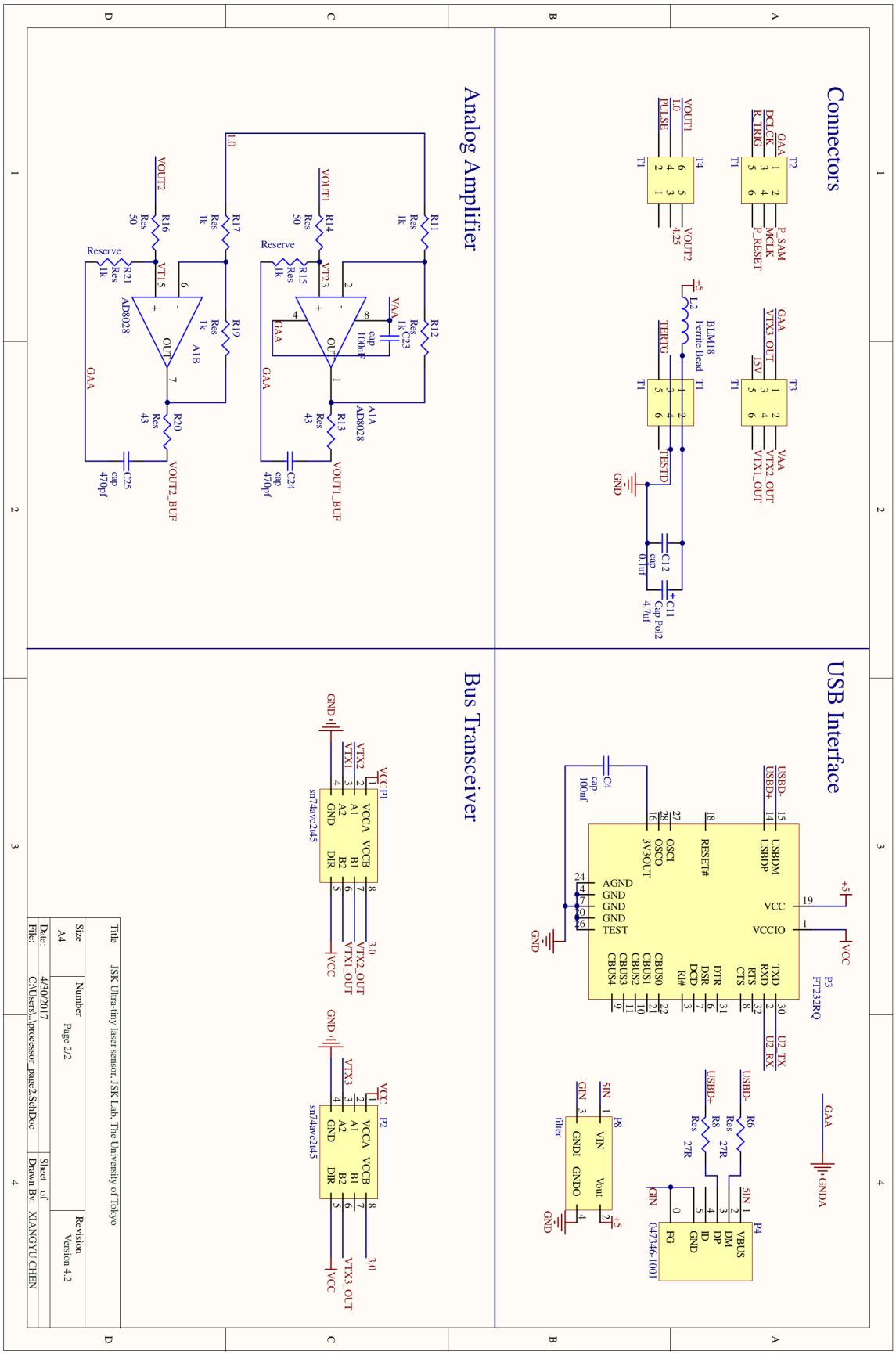
A.1 Hardware Schematic

A.2 Calibration Objects



119

Figure A.1: Hardware Bottom PCB Schematic Page 1



120

Figure A.2: Hardware Bottom PCB Schematic Page 2

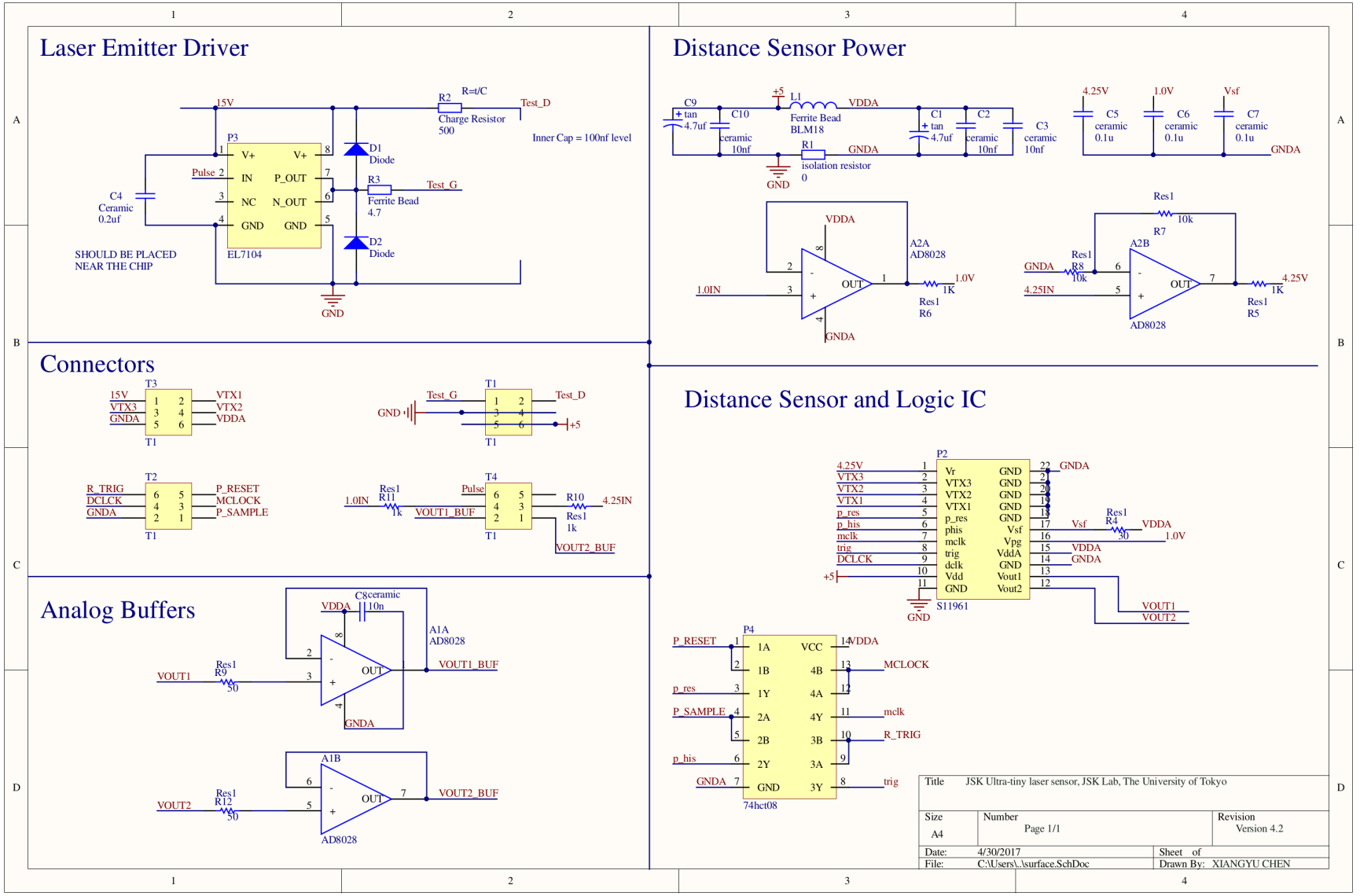


Figure A.3: Hardware Top PCB Schematic

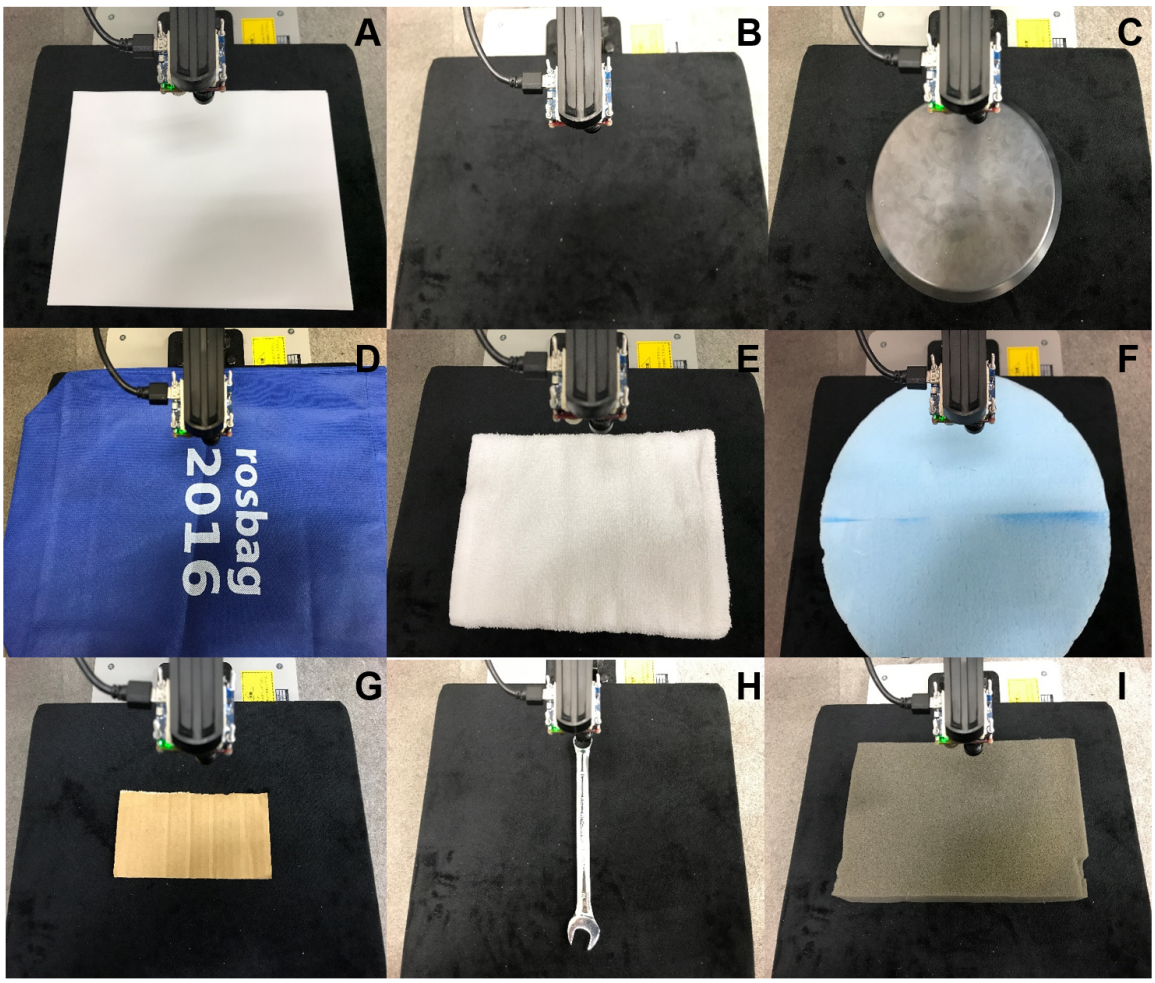


Figure A.4: Tested Objects

Appendix B

Publications and Awards

B.1 First Author Publications

• **Xiangyu Chen**, Krishneel Chaudhary, Yoshimaru Tanaka , Kotaro Nagahama, Hiroaki Yaguchi, Kei Okada, and Masayuki Inaba. Reasoning-based vision recognition for agricultural humanoid robot toward tomato harvesting. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on. IEEE, 2015.* p. 6487-6494.

• **Xiangyu Chen**, Moju Zhao, Lingzhu Xiang, Fumihito Sugai, Hiroaki Yaguchi, Kei Okada, and Masayuki Inaba. Development of a low-cost ultra-tiny line laser range sensor. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on. IEEE, 2016.* p. 111-116.

• **Xiangyu Chen**, Kohei Kimura, Hiroto Mizohana, Moju Zhao, Fan Shi, Krishneel Chaudhary, Wesley P. Chan, Shunichi Nozawa, Yohei Kakiuchi, Kei Okada and Masayuki Inaba. Development of task-oriented high power field robot platform with humanoid upper body and mobile wheeled base. In *System Integration (SII), 2016 IEEE/SICE International Symposium on. IEEE, 2016.* p. 349-354.

B.2 Other Publications

• Moju Zhao, Koji Kawasaki, **Xiangyu Chen**, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Transformable Multirotor with Two-Dimensional Multilinks: Modeling, Control, and Whole-Body Aerial Manipulation. In *International Symposium on Experimental Robotics. Springer, Cham, 2016.* p. 515-524.

• Moju Zhao, Koji Kawasaki, **Xiangyu Chen**, Shintaro Noda, Kei Okada, and Masayuki Inaba. Whole-Body Aerial Manipulation by Transformable Multirotor with Two-Dimensional Multilinks. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on. IEEE, 2017.* p. 5175-5182.

• Krishneel Chaudhary, **Xiangyu Chen**, Wesley P. Chan, Kei Okada, and Masayuki Inaba. STAIR3D: Simultaneous Tracking And Incremental Registration For Modeling 3D Handheld Objects. In *Advanced Intelligent Mechatronics (AIM), 2017 IEEE International Conference on. IEEE, 2017.* p. 185-192. **Best Student Paper Award**

• Tomoki Anzai, Moju Zhao, **Xiangyu Chen**, Fan Shi, Koji Kawasaki, Kei Okada and Masayuki Inaba. Multilinked Multirotor with Internal Communication System for Multiple Objects Transportation based on Form Optimization Method. **Accepted** In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on. IEEE, 2017.*

• Krishneel Chaudhary, Moju Zhao, Fan Shi, **Xiangyu Chen**, Kei Okada and Masayuki Inaba. Robust Real-Time Visual Tracking Using Dual-Frame Deep Comparison Network Integrated with Correlation Filters. **Accepted** In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on. IEEE, 2017.*

• 溝花弘登, Chan Wesley, 木村航平, **陳相羽**, 岡田慧, 稲葉雅幸: 移動台車型ヒューマノイドによる屋外環境での工具取得操作行動における環境と自己の陰影を考慮する認識行動, in 第17回SICEシステムインテグレーション部門講演会講演概要集, pp.2611–2616, 2016.

B.3 Awards

• Winner of the First Tomato Robot Harvesting Challenge Competition in 2014.

• Best Student Paper Award of *IEEE International Conference on Advanced Intelligent Mechatronics (AIM), 2017 IEEE*

Bibliography

[1] Xiangyu Chen, Moju Zhao, Lingzhu Xiang, Fumihito Sugai, Hiroaki Yaguchi, Kei Okada, and Masayuki Inaba. Development of a low-cost ultra-tiny line laser range sensor. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 111–116. IEEE, 2016.

[2] Xiangyu Chen, Krishneel Chaudhary, Yoshimaru Tanaka, Kotaro Nagahama, Hiroaki Yaguchi, Kei Okada, and Masayuki Inaba. Reasoning-based vision recognition for agricultural humanoid robot toward tomato harvesting. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 6487–6494. IEEE, 2015.

[3] Xiangyu Chen, Kohei Kimura, Hiroto Mizohana, Moju Zhao, Fan Shi, Krishneel Chaudhary, Wesley P Chan, Shunichi Nozawa, Yohei Kakiuchi, Kei Okada, et al. Development of task-oriented high power field robot platform with humanoid upper body and mobile wheeled base. In *System Integration (SII), 2016 IEEE/SICE International Symposium on*, pages 349–354. IEEE, 2016.

[4] Shimon Y Nof. *Handbook of industrial robotics*, volume 1. John Wiley & Sons, 1999.

[5] Y Sarig. Robotics of fruit harvesting: A state-of-the-art review. *Journal of agricultural engineering research*, 54(4):265–280, 1993.

[6] Mario M Foglia and Giulio Reina. Agricultural robot for radicchio harvesting. *Journal of Field Robotics*, 23(6-7):363–377, 2006.

[7] Arno Ruckelshausen, Peter Biber, Michael Dorna, Holger Gremmes, Ralph Klose, Andreas Linz, Florian Rahe, Rainer Resch, Marius Thiel, Dieter Trautz, et al. Bonirob–an autonomous field robot platform for individual plant phenotyping. *Precision agriculture*, 9(841):1, 2009.

[8] Anthony R Lanfranco, Andres E Castellanos, Jaydev P Desai, and William C Meyers. Robotic surgery: a current perspective. *Annals of surgery*, 239(1):14–21, 2004.

[9] Gabriel I Barbash and Sherry A Glied. New technology and health care costs—the case of robot-assisted surgery. *New England Journal of Medicine*, 363(8):701–704, 2010.

[10] Gyung Tak Sung and Inderbir S Gill. Robotic laparoscopic surgery: a comparison of the da vinci and zeus systems. *Urology*, 58(6):893–898, 2001.

[11] Hobart R Everett and Douglas W Gage. From laboratory to warehouse: Security robots meet the real world. *The International Journal of Robotics Research*, 18(7):760–768, 1999.

[12] Colin Rennie, Rahul Shome, Kostas E Bekris, and Alberto F De Souza. A dataset for improved rgbd-based object detection and pose estimation for warehouse pick-and-place. *IEEE Robotics and Automation Letters*, 1(2):1179–1185, 2016.

[13] Yoshiro Hada, Harunori Gakuhari, Kunikatsu Takase, and Edward Indyanto Hemeldan. Delivery service robot using distributed acquisition, actuators and intelligence. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2997–3002. IEEE, 2004.

[14] Tolga Bektas. The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, 34(3):209–219, 2006.

[15] Chase C Murray and Amanda G Chu. The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Research Part C: Emerging Technologies*, 54:86–109, 2015.

[16] Wesley Patrick Chan. *Research on Affordance-Focused Learning and Generalization through Observation of Proper Handovers and Object Usages in Robot-Human Interactions*. PhD thesis, The University of Tokyo, 2016.

[17] Keiichi Yamazaki, Michie Kawashima, Yoshinori Kuno, Naonori Akiya, Matthew Burdelski, Akiko Yamazaki, and Hideaki Kuzuoka. Prior-to-request and request behaviors within elderly day care: Implications for developing service robots for use in multiparty settings. In *ECSCW 2007*, pages 61–78. Springer, 2007.

[18] David Fischinger, Peter Einramhof, Konstantinos Papoutsakis, Walter Wohlkinger, Peter Mayer, Paul Panek, Stefan Hofmann, Tobias Koertner, Astrid Weiss, Antonis Argyros, et al. Hobbit, a care robot supporting independent living at home: First prototype and lessons learned. *Robotics and Autonomous Systems*, 75:60–78, 2016.

[19] J Edward Colgate, J Edward, Michael A Peshkin, and Witaya Wannasuphoprasit. Cobots: Robots for collaboration with human operators. 1996.

[20] Sebastian Thrun, Mike Montemerlo, Hendrik Dahlkamp, David Stavens, Andrei Aron, James Diebel, Philip Fong, John Gale, Morgan Halpenny, Gabriel Hoffmann, et al. Stanley: The robot that won the darpa grand challenge. *Journal of field Robotics*, 23(9):661–692, 2006.

[21] Chris Urmson, Joshua Anhalt, Drew Bagnell, Christopher Baker, Robert Bittner, MN Clark, John Dolan, Dave Duggins, Tugrul Galatali, Chris Geyer, et al. Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics*, 25(8):425–466, 2008.

[22] Junichi Urata, Koichi Nshiwaki, Yuto Nakanishi, Kei Okada, Satoshi Kagami, and Masayuki Inaba. Online decision of foot placement using singular lq preview regulation. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 13–18. IEEE, 2011.

[23] Junichi Urata, Yuto Nakanishi, Kei Okada, and Masayuki Inaba. Design of high torque and high speed leg module for high power humanoid. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 4497–4502. IEEE, 2010.

[24] Ill-Woo Park, Jung-Yup Kim, Jungho Lee, and Jun-Ho Oh. Mechanical design of humanoid robot platform khr-3 (kaist humanoid robot 3: Hubo). In *Humanoid Robots, 2005 5th IEEE-RAS International Conference on*, pages 321–326. IEEE, 2005.

[25] Frank Tobe. 62 market research reports study robotics industry. `https://www.therobotreport.com/news/62-market-research-reports`, Octorber 2016.

[26] Lawrence D. Rosenblum. *See What I'm Saying: The Extraordinary Powers of Our Five Senses*. W. W. Norton Company, 2010.

[27] Giorgio Cannata and Marco Maggiali. Implementation of listing's law for a tendon driven robot eye. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 3940–3945. IEEE, 2006.

[28] Alexander Lenz, Sean R Anderson, Anthony G Pipe, Chris Melhuish, Paul Dean, and John Porrill. Cerebellar-inspired adaptive control of a robot eye actuated by pneumatic artificial muscles. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(6):1420–1433, 2009.

[29] Roger Y Tsai and Reimar K Lenz. A new technique for fully autonomous and efficient 3d robotics hand/eye calibration. *IEEE Transactions on robotics and automation*, 5(3):345–358, 1989.

[30] Adi Bonen, Ricardo E Saad, Kenneth C Smith, and Beno Benhabib. A novel electrooptical proximity sensor for robotics: Calibration and active sensing. *IEEE Transactions on Robotics and Automation*, 13(3):377–386, 1997.

[31] Ray A Jarvis. A perspective on range finding techniques for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2):122–139, 1983.

[32] Christopher M Bishop. Pattern recognition. *Machine Learning*, 128:1–58, 2006.

[33] Alex Flint, David Murray, and Ian Reid. Manhattan scene understanding using monocular, stereo, and 3d features. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2228–2235. IEEE, 2011.

[34] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 3212–3217. IEEE, 2009.

[35] Zhaoyin Jia, Andrew Gallagher, Ashutosh Saxena, and Tsuhan Chen. 3d-based reasoning with blocks, support, and stability. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2013.

[36] Abhinav Gupta, Martial Hebert, Takeo Kanade, and David M Blei. Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In *Advances in neural information processing systems*, pages 1288–1296, 2010.

[37] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics, 2002.

[38] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[39] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[40] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008.

[41] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[42] Hans Moravec and Alberto Elfes. High resolution maps from wide angle sonar. In *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, volume 2, pages 116–121. IEEE, 1985.

[43] John J Leonard and Hugh F Durrant-Whyte. *Directed sonar sensing for mobile robot navigation*, volume 175. Springer Science & Business Media, 2012.

[44] François Blais. Review of 20 years of range sensor development. *Journal of Electronic Imaging*, 13(1):231–243, 2004.

[45] Kurt Konolige, Joseph Augenbraun, Nick Donaldson, Charles Fiebig, and Pankaj Shah. A low-cost laser distance sensor. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 3002–3008. IEEE, 2008.

[46] Marc Rioux. Laser range finder based on synchronized scanners. *Applied optics*, 23(21):3837–3844, 1984.

[47] Olaf Hall-Holt and Szymon Rusinkiewicz. Stripe boundary codes for real-time structured-light range scanning of moving objects. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 359–366. IEEE, 2001.

[48] Li Zhang, Brian Curless, and Steven M Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *3D Data Processing Visualization and Transmission, 2002. Proceedings. First International Symposium on*, pages 24–36. IEEE, 2002.

[49] Majd Alwan, Matthew B Wagner, Glenn Wasson, and Pradip Sheth. Characterization of infrared range-finder pbs-03jn for 2-d mapping. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 3936–3941. IEEE, 2005.

[50] Hirohiko Kawata, Akihisa Ohya, Shinichi Yuta, Wagle Santosh, and Toshihiro Mori. Development of ultra-small lightweight optical range sensor system. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 1078–1083. IEEE, 2005.

[51] Andreas Kolb, Erhardt Barth, Reinhard Koch, and Rasmus Larsen. Time-of-flight cameras in computer graphics. In *Computer Graphics Forum*, volume 29, pages 141–159. Wiley Online Library, 2010.

[52] Yuke Zhu, Alireza Fathi, and Li Fei-Fei. Reasoning about object affordances in a knowledge base representation. In *European conference on computer vision*, pages 408–424. Springer, 2014.

[53] Yixin Zhu, Yibiao Zhao, and Song Chun Zhu. Understanding tools: Task-oriented object modeling, learning and recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2855–2864, 2015.

[54] Yibiao Zhao and Song-Chun Zhu. Scene parsing by integrating function, geometry and appearance models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3119–3126, 2013.

[55] Yixin Zhu, Chenfanfu Jiang, Yibiao Zhao, Demetri Terzopoulos, and Song-Chun Zhu. Inferring forces and learning human utilities from videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3823–3833, 2016.

[56] B. Espiau. An overview of local environment sensing in robotics. *Sensors and Sensory Systems for Advanced Robots*, F43:125–151, 1988.

[57] Dieter Koller, Joseph Weber, and Jitendra Malik. Robust multiple car tracking with occlusion reasoning. In *European Conference on Computer Vision*, pages 189–196. Springer, 1994.

[58] Bo Zheng, Yibiao Zhao, C Yu Joey, Katsushi Ikeuchi, and Song-Chun Zhu. Detecting potential falling objects by inferring human action and natural disturbance. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 3417–3424. IEEE, 2014.

[59] Rodney A Brooks. Elephants don't play chess. *Robotics and autonomous systems*, 6(1-2):3–15, 1990.

[60] Rodney A Brooks. Intelligence without representation. *Artificial intelligence*, 47(1-3):139–159, 1991.

[61] Nils J Nilsson. Shakey the robot. Technical report, DTIC Document, 1984.

[62] Lonnie Chrisman, Rich Caruana, and Wayne Carriker. Intelligent agent design issues: Internal agent state and incomplete perception. In *Proceedings of the AAAI Fall Symposium on Sensory Aspects of Robotic Intelligence. AAAI Press/MIT Press*. Citeseer, 1991.

[63] Paul J Besl. Active, optical range imaging sensors. *Machine vision and applications*, 1(2):127–152, 1988.

[64] SICK AG. *LMS5xx Laser Measurement Sensors*, 2015.

[65] Michael Montemerlo, Jan Becker, Suhrid Bhat, Hendrik Dahlkamp, Dmitri Dolgov, Scott Ettinger, Dirk Haehnel, Tim Hilden, Gabe Hoffmann, Burkhard Huhnke, et al. Junior: The stanford entry in the urban challenge. *Journal of field Robotics*, 25(9):569–597, 2008.

[66] Hansjoerg Gaertner, Peter Lehle, and Hans J Tiziani. New highly efficient binary codes for structured light methods. In *Photonics East'95*, pages 4–13. International Society for Optics and Photonics, 1996.

[67] Giovanna Sansoni, S Corini, S Lazzari, R Rodella, and F Docchio. Three-dimensional imaging based on gray-code light projection: characterization of the measuring algorithm and development of a measuring system for industrial applications. *Applied Optics*, 36(19):4463–4472, 1997.

[68] Patrick Mihelich Kurt Konolige. Technical description of Kinect calibration. `http://wiki.ros.org/kinect_calibration/technical`, December 2012.

[69] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 573–580. IEEE, 2012.

[70] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568. ACM, 2011.

[71] Michael Van den Bergh, Daniel Carton, Roderick De Nijs, Nikos Mitsou, Christian Landsiedel, Kolja Kuehnlenz, Dirk Wollherr, Luc Van Gool, and Martin Buss. Real-time 3d hand gesture interaction with a robot for understanding directions from humans. In *RO-MAN, 2011 IEEE*, pages 357–362. IEEE, 2011.

[72] Andrew Payne, Andy Daniel, Anik Mehta, Barry Thompson, Cyrus S Bamji, Dane Snow, Hideaki Oshima, Larry Prather, Mike Fenton, Lou Kordus, et al. 7.6 a 512× 424 cmos 3d time-of-flight image sensor with multi-frequency photodemodulation up to 130mhz and 2gs/s adc. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, pages 134–135. IEEE, 2014.

[73] Hamed Sarbolandi, Damien Lefloch, and Andreas Kolb. Kinect range sensing: Structured-light versus time-of-flight kinect. *Computer vision and image understanding*, 139:1–20, 2015.

[74] Rolf Kaufmann, Michael Lehmann, Matthias Schweizer, Michael Richter, Peter Metzler, Graham Lang, Thierry Oggier, Nicolas Blanc, Peter Seitz, Gabriel Gruener, et al. A time-of-flight line sensor: development and application. In *Photonics Europe*, pages 192–199. International Society for Optics and Photonics, 2004.

[75] Robert Lange and Peter Seitz. Solid-state time-of-flight range camera. *IEEE Journal of quantum electronics*, 37(3):390–397, 2001.

[76] Gavriel J Iddan and Giora Yahav. Three-dimensional imaging in the studio and elsewhere. In *Photonics West 2001-Electronic Imaging*, pages 48–55. International Society for Optics and Photonics, 2001.

[77] Giora Yahav, Gabi J Iddan, and David Mandelboum. 3d imaging camera for gaming application. In *Consumer Electronics, 2007. ICCE 2007. Digest of Technical Papers. International Conference on*, pages 1–2. IEEE, 2007.

[78] Shoji Kawahito, Izhal Abdul Halin, Takeo Ushinaga, Tomonari Sawada, Mitsuru Homma, and Yasunari Maeda. A cmos time-of-flight range image sensor with gates-on-field-oxide structure. *IEEE Sensors Journal*, 7(12):1578–1586, 2007.

[79] Seong-Jin Kim, James DK Kim, Byongmin Kang, and Keechang Lee. A cmos image sensor based on unified pixel architecture with time-division multiplexing scheme for color and depth image acquisition. *IEEE Journal of Solid-State Circuits*, 47(11):2834–2845, 2012.

[80] Shoji Kawahito, Guseul Baek, Zhuo Li, Sang-Man Han, Min-Woong Seo, Keita Yasutomi, and Keiichiro Kagawa. Cmos lock-in pixel image sensors with lateral electric field control for time-resolved imaging. In *Proc. Int. Image Sensor Workshop (IISW)*, pages 361–364, 2013.

[81] Hongbin Zha, Ken'ichi Morooka, Tsutomu Hasegawa, and Tadashi Nagata. Active modeling of 3-d objects: planning on the next best pose (nbp) for acquiring range images. In *3-D Digital Imaging and Modeling, 1997. Proceedings., International Conference on Recent Advances in*, pages 68–75. IEEE, 1997.

[82] Sumantra Dutta Roy, Santanu Chaudhury, and Subhashis Banerjee. Active recognition through next view planning: a survey. *Pattern Recognition*, 37(3):429–446, 2004.

[83] Christian Potthast and Gaurav S Sukhatme. A probabilistic framework for next best view estimation in a cluttered environment. *Journal of Visual Communication and Image Representation*, 25(1):148–164, 2014.

[84] Cl Connolly. The determination of next best views. In *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, volume 2, pages 432–435. IEEE, 1985.

[85] Helmut Grabner, Juergen Gall, and Luc Van Gool. What makes a chair a chair? In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1529–1536. IEEE, 2011.

[86] Bo Zheng, Yibiao Zhao, Joey C Yu, Katsushi Ikeuchi, and Song-Chun Zhu. Beyond point clouds: Scene understanding by reasoning geometry and physics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3127–3134, 2013.

[87] Mingtao Pei, Yunde Jia, and Song-Chun Zhu. Parsing video events with goal inference and intent prediction. In *Computer vision (iccv), 2011 ieee international conference on*, pages 487–494. IEEE, 2011.

[88] Jessica Hamrick, Peter Battaglia, and Joshua B Tenenbaum. Internal physics models guide probabilistic judgments about object dynamics. In *Proceedings of the 33rd annual conference of the cognitive science society*, pages 1545–1550. Cognitive Science Society Austin, TX, 2011.

[89] Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41, 2013.

[90] Rodney Brooks. A robust layered control system for a mobile robot. *IEEE journal on robotics and automation*, 2(1):14–23, 1986.

[91] Lorenzo Jamone, Lorenzo Natale, Giorgio Metta, and Giulio Sandini. Highly sensitive soft tactile sensors for an anthropomorphic robotic hand. *IEEE sensors Journal*, 15(8):4226–4233, 2015.

[92] Yosuke Suzuki, Keisuke Koyama, Aiguo Ming, and Makoto Shimojo. Grasping strategy for moving object using net-structure proximity sensor and vision sensor. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 1403–1409. IEEE, 2015.

[93] Stefan Escaida Navarro, Maximiliano Marufo, Yitao Ding, Stephan Puls, Dirk Göger, Björn Hein, and Heinz Wörn. Methods for safe human-robot-interaction using capacitive tactile proximity sensors. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 1149–1154. IEEE, 2013.

[94] Hiroaki Hasegawa, Yosuke Suzuki, Aiguo Ming, Keisuke Koyama, Masatoshi Ishikawa, and Makoto Shimojo. Net-structure proximity sensor: High-speed and free-form sensor with analog computing circuit. *IEEE/ASME Transactions on Mechatronics*, 20(6):3232–3241, 2015.

[95] Keisuke Koyama, Yosuke Suzuki, Aiguo Ming, and Makoto Shimojo. Grasping control based on time-to-contact method for a robot hand equipped with proximity sensors on fingertips. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 504–510. IEEE, 2015.

[96] Huijing Zhao, Jie Sha, Yipu Zhao, Junqiang Xi, Jinshi Cui, Hongbin Zha, and Ryosuke Shibasaki. Detection and tracking of moving objects at intersections using a network of laser scanners. *IEEE transactions on intelligent transportation systems*, 13(2):655–670, 2012.

[97] Frank Moosmann and Christoph Stiller. Velodyne slam. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 393–398. IEEE, 2011.

[98] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005.

[99] Ismael Colomina and Pere Molina. Unmanned aerial systems for photogrammetry and remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 92:79–97, 2014.

[100] Hamamatsu Photonics. *Distance Linear Image Sensor, S11961-01CR, User Manual*, 2015.

[101] OSRAM Opto Semiconductors. *Hybrid Pulsed Laser Diode with Integrated Driver Stage 70W Peak, User Manual*, 2009.

[102] Jasmin Blanchette and Mark Summerfield. *C++ GUI programming with Qt 4*. Prentice Hall Professional, 2006.

[103] Intersil Americas Inc. *High Speed, Single Channel, Power MOSFET Driver*, 2006.

[104] Oliver Wasenmüller and Didier Stricker. Comparison of kinect v1 and v2 depth images in terms of accuracy and precision. In *Asian Conference on Computer Vision*, pages 34–45. Springer, 2016.

[105] Yael Edan, Dima Rogozin, Tamar Flash, and Gaines E Miles. Robotic melon harvesting. *IEEE Transactions on Robotics and Automation*, 16(6):831–835, 2000.

[106] Mahmoud Omid, Majid Lashgari, Hossein Mobli, Reza Alimardani, Saeid Mohtasebi, and Reza Hesamifard. Design of fuzzy logic control system incorporating human expert knowledge for combine harvester. *Expert Systems with Applications*, 37(10):7080–7085, 2010.

[107] Johan Baeten, Kevin Donné, Sven Boedrij, Wim Beckers, and Eric Claesen. Autonomous fruit picking machine: A robotic apple harvester. In *Field and service robotics*, pages 531–539. Springer, 2008.

[108] Shigehiko Hayashi, Kenta Shigematsu, Satoshi Yamamoto, Ken Kobayashi, Yasushi Kohno, Junzo Kamata, and Mitsutaka Kurita. Evaluation of a strawberry-harvesting robot in a field test. *Biosystems engineering*, 105(2):160–171, 2010.

[109] Rong Xiang, Huanyu Jiang, and Yibin Ying. Recognition of clustered tomatoes based on binocular stereo vision. *Computers and Electronics in Agriculture*, 106:75–90, 2014.

[110] Achmad Irwan Setiawan, Tomonari Furukawa, and Adam Preston. A low-cost gripper for an apple picking robot. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 5, pages 4448–4453. IEEE, 2004.

[111] Baozeng Jia, Anmin Zhu, Simon X Yang, and Guari S Mittal. Integrated gripper and cutter in a mobile robotic system for harvesting greenhouse products. In *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on*, pages 1778–1783. IEEE, 2009.

[112] Fabien Gravot, Atsushi Haneda, Kei Okada, and Masayuki Inaba. Cooking for humanoid robot, a task that needs symbolic and geometric reasonings. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 462–467. IEEE, 2006.

[113] Kimitoshi Yamazaki, Ryohei Ueda, Shunichi Nozawa, Yuto Mori, Toshiaki Maki, Naotaka Hatao, Kei Okada, and Masayuki Inaba. System integration

of a daily assistive robot and its application to tidying and cleaning rooms. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 1365–1371. IEEE, 2010.

[114] Shunichi Nozawa, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Controlling the planar motion of a heavy object by pushing with a humanoid robot using dual-arm force control. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1428–1435. IEEE, 2012.

[115] Wesley P Chan, Chris AC Parker, Hf Machiel Van Der Loos, and Elizabeth A Croft. A human-inspired object handover controller. *The International Journal of Robotics Research*, 32(8):971–983, 2013.

[116] Masaki Murooka, Shintaro Noda, Shunichi Nozawa, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Manipulation strategy decision and execution based on strategy proving operation for carrying large and heavy objects. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 3425–3432. IEEE, 2014.

[117] Daniel Herrera, Juho Kannala, and Janne Heikkilä. Joint depth and color camera calibration with distortion correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(10):2058–2064, 2012.

[118] Ziv Yaniv. Random sample consensus (ransac) algorithm, a generic implementation. *Imaging*, 2010.

[119] Ruwen Schnabel, Roland Wahl, and Reinhard Klein. Efficient ransac for point-cloud shape detection. In *Computer graphics forum*, volume 26, pages 214–226. Wiley Online Library, 2007.

[120] David Levin. Mesh-independent surface interpolation. In *Geometric modeling for scientific visualization*, pages 37–49. Springer, 2004.

[121] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[122] JSK MBZIRC Team. Statement of Work in Mohamed Bin Zayed International Robotics Challenge of Team JSK. `http://www.jsk.t.u-tokyo.ac.jp/~xychen/mbzirc/mbzirc_proposal.pdf`, December 2015.

[123] JSK MBZIRC Team. Team-JSK: MBZIRC Progress Report. `http://www.jsk.t.u-tokyo.ac.jp/~xychen/mbzirc/first_progress_report.pdf`, August 2016.

[124] JSK MBZIRC Team. Team-JSK: MBZIRC Progress Report 2. `http://www.jsk.t.u-tokyo.ac.jp/~xychen/mbzirc/second_progress_report.pdf`, November 2016.

[125] De Xu, Carlos A Acosta Calderon, John Q Gan, Huosheng Hu, and Min Tan. An analysis of the inverse kinematics for a 5-dof manipulator. *International Journal of Automation and Computing*, 2(2):114–124, 2005.

[126] Ying Wang, Haoxiang Lang, and Clarence W de Silva. A hybrid visual servo controller for robust grasping by wheeled mobile robots. *IEEE/ASME transactions on Mechatronics*, 15(5):757–769, 2010.

[127] Chris J Ostafew, Angela P Schoellig, and Timothy D Barfoot. Visual teach and repeat, repeat, repeat: Iterative learning control to improve mobile robot path tracking in challenging outdoor environments. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 176–181. IEEE, 2013.

[128] Marc J Gallant and Joshua A Marshall. Estimating the heading of a husky mobile robot with a lidar compass based on direction maps. 2016.

[129] Kei Okada, Takashi Ogura, Atsushi Haneda, Junya Fujimoto, Fabien Gravot, and Masayuki Inaba. Humanoid motion generation system on hrp2-jsk for daily life environment. In *IEEE International Conference Mechatronics and Automation, 2005*, volume 4, pages 1772–1777. IEEE, 2005.

[130] Kimitoshi Yamazaki, Ryohei Ueda, Shunichi Nozawa, Mitsuharu Kojima, Kei Okada, Kiyoshi Matsumoto, Masaru Ishikawa, Isao Shimoyama, and Masayuki Inaba. Home-assistant robot for an aging society. *Proceedings of the IEEE*, 100(8):2429–2441, 2012.

[131] Sourabh Bhattacharya, Rafael Murrieta-Cid, and Seth Hutchinson. Optimal paths for landmark-based navigation by differential-drive vehicles with field-of-view constraints. *IEEE Transactions on Robotics*, 23(1):47–59, 2007.

[132] Anthony Mandow, Jorge L Martinez, Jesús Morales, José L Blanco, Alfonso Garcia-Cerezo, and Javier Gonzalez. Experimental kinematics for wheeled skid-steer mobile robots. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1222–1227. IEEE, 2007.

[133] RJE Merry, MJG Van de Molengraft, and Maarten Steinbuch. Velocity and acceleration estimation for optical incremental encoders. *Mechatronics*, 20(1):20–26, 2010.

[134] Nathan Koenig and Andrew Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2149–2154. IEEE, 2004.

[135] Moju Zhao, Koji Kawasaki, Kei Okada, and Masayuki Inaba. Transformable multirotor with two-dimensional multilinks: modeling, control, and motion planning for aerial transformation. *Advanced Robotics*, 30(13):825–845, 2016.

[136] Moju Zhao, Koji Kawasaki, Xiangyu Chen, Shintaro Noda, Kei Okada, and Masayuki Inaba. Whole-body aerial manipulation by transformable multirotor with two-dimensional multilinks. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 5175–5182. IEEE, 2017.

[137] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[138] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11):1330–1334, 2000.

[139] Johannes Meyer, Alexander Sendobry, Stefan Kohlbrecher, Uwe Klingauf, and Oskar Von Stryk. Comprehensive simulation of quadrotor uavs using ros and gazebo. In *International Conference on Simulation, Modeling, and Programming for Autonomous Robots*, pages 400–411. Springer, 2012.