Title	New sensing system for autonomous space rendezvous using time of flight camera
Sub Title	
Author	Nguyen, Thi Mai Thu(Haruyama, Shinichiro) 春山, 真一郎
Publisher	慶應義塾大学大学院システムデザイン・マネジメント研究科
Publication year	2016
Jtitle	
JaLC DOI	
Abstract	
Notes	修士学位論文. 2016年度システムエンジニアリング学 第222号
Genre	Thesis or Dissertation
URL	https://koara.lib.keio.ac.jp/xoonips/modules/xoonips/detail.php?koara_id=KO40002001-00002016-0004

Master's Dissertation 2016

New Sensing System for Autonomous Space Rendezvous Using Time of Flight Camera

Nguyen Thi Mai Thu
（Student ID Number：81434542）

Supervisor Professor Shinichiro Haruyama

September 2016

Graduate School of System Design and Management,
Keio University
Major in System Design and Management

SUMMARY OF MASTER'S DISSERTATION

Student Identification Number	81434542	Name	NGUYEN THI MAI THU

Title

New Sensing System for Autonomous Space Rendezvous
Using Time of Flight Camera

Abstract

This research proposes the application of time-of-flight cameras (TOF cameras) in a new sensing system, which is responsible for relative navigations, in space rendezvous.

Space rendezvous is an event in which one spacecraft chases and gets closer to another spacecraft or space body. This is the important activity, which takes place in space events such as supplying resources to the International Space Station or landing equipment to comets and asteroids. To be able to achieve this mission, the chasing spacecraft (chaser), through its sensing system, knows the relative position and attitude of the target at a near-real time.

Following the small satellite trend, this research is the work of designing a new sensing system using commercial TOF cameras. These cameras are cheap, small, light, and fast equipped. Therefore, this system can be used in not only huge spacecraft but also the micro class of satellites.

A TOF camera transmits modulated infrared lights, then receives the reflected ones, and compares them to generate information. Three corner-cube-reflectors (CCRs) are attached to the target, followed a special design, to form the interface. Using this combination, the navigation software is created to calculate the navigating parameters.

The tests were conducted to evaluate the quality of the system. The prototype used a TOF camera from Stanley and three CCRs. The navigation software was written in the C++ programming language.

The results showed that this new sensing system can be well applied in many space rendezvous events such as supplying the ISS (cooperative target) with a support of a robot arm, collecting space debris, or landing to space bodies (non-cooperative targets).

Key Word (5 words)

Space Rendezvous, TOF Camera, Computer Vision, Laser Beam, Corner Cube Reflector.

ACKNOWLEDGEMENT

This master thesis was a long journey in which I gained a lot of technical knowledge as well as a life lesson of how to approach a difficult goal. I would first love to thank my thesis supervisor, Prof. Shinichiro HARUYAMA, who continuously and patiently guided and helped me in every single step and throughout the entire process of my research. He was always there to steer me towards the right direction and to give me insightful advice. He provided me all the necessary devices and components for my work. He, by his hands, created the prototyping model, which was essential for this research. Without his support, my research could not be completed. The effort he gave me was enormous that at this moment, I find words could not fully express my gratitude to him.

I would love to send a big thank to Dr. Masaaki Mokuno, who graduated from Keio University and now is working as an engineer at JAXA, for all his documents, which are about the space rendezvous tests that he has experienced before, and also for his kindness to come and to give me a chance to consult him about my idea. All those crucial help and information were very important for this master degree.

I would also like to thank Mr. Kosuke Mori, who is a doctoral student in Keio University, for his enthusiasm in helping me from the very first step when I got to work with a device that was totally new to me. Without his support, one-third of my work could not be achieved.

I would like to send my thankfulness to Dr. Hideaki Uchiyama, who is an assistant professor at Kyushu University, Japan, for coming to our laboratory and giving us the lecture of his beautiful work in computer vision technologies.

I would love to express here my gratitude to Prof. Tomohiko Taniguchi for being my secondary advisor. He, despite a busy schedule, took times to review and to give me a lot of useful advice and comments to help me improve the quality of my research.

I would also like to thank all the SDM professors, members, students, and staff those have been supporting me for the two years studying here. Especially, I want to thank the professors and faculty members, who belong to our MDG project, for their readiness to help me anytime I needed, and for the important information related to the space industry that they gave me through many discussions as well as along the process of making our MDG satellite.

I want to thank all the researchers whose papers were published, with the huge amount of information, knowledge, and achievements that I have learnt from and have used to contribute my thesis.

Last but not least, I would love to express my profound gratitude to my family members, my friends, and my VNSC colleagues those always remotely support me from either my home country or other countries. The encouragement from them has brought me through all the tough times in the process of doing this research.

Thank you!

Nguyen Thi Mai Thu.

Yokohama, September 2016.

Contents

Table of Figures

List of Tables

Abbreviations List

TOF:	Time-Of-Flight
CCR:	Corner-Cube-Reflector
IR:	Infrared
ISS:	International Space Station
NASA:	National Aeronautics and Space Administration
ESA:	European Space Agency
JAXA:	Japan Aerospace Exploration Agency
ATV:	Automated Transfer Vehicle
OBC:	Onboard Computer
ACS:	Attitude Control System
AOBC:	ACS Onboard Computer
RF:	Radio Frequency
FOV:	Field-Of-View
LOS:	Line-Of-Sight
CMOS:	Complementary Metal-Oxide-Semiconductor
CCD:	Charge Coupled Device
APD:	Avalanche Photodiode
RVR:	Rendezvous Docking
ETS:	Engineering Test Satellite
PMD:	Photonic Mixer Device
2D:	Two-Dimensional
3D:	Three-Dimensional
DLL:	Dynamic-Link Library

1 Introduction

This chapter first introduces the background of space rendezvous **(1.1)**, then, the importance of the relative satellite navigation in space rendezvous will be explained **(1.2)**, followed by the clarification of some space rendezvous scenarios **(1.3)** and the necessary measurement parameters **(1.4)**. Finally, the layout of this report will be briefly described to give an overview of what the research has done **(1.5)**.

1.1 Background

Going to space has always been a dream of human beings. The further that can be reached, the more knowledge there is to be discovered about this mysterious place where everything exists.

Back to the ancient times, people looked at the sky and wondered about the Earth and the universe, guessed and created theories. Many scientists had desired the answers of their questions to find out the truth: Nicolaus Copernicus (February 19, 1473 – May 24, 1543), Johannes Kepler (December 27, 1571 – November 15, 1630), Galileo Galilei (February 15, 1564 – January 8, 1642) and Isaac Newton (December 25, 1642 – March 20, 1726) in the sixteenth and seventeenth centuries to name a few. They, with the contribution of many others, changed the concept of the universe by their scientific model(s) and put the fundamental for space travel by their physical laws, which are still being used now.

However, space travel had been in the imagination of literature, of science fiction and cinema until the nineteenth century, the time of rocketry pioneers and visionaries (Joseph A. Angelo, 2006).

Since the beginning of the modern rocketry, which was marked by the event of the world's first solid-fueled rocket successfully launched by the pioneer Robert Goddard (October 5, 1882 – August 10, 1945) on 16th March 1926 (Marconi, 2004), the age of space ambitions and technologies had been opened. As people suspected, that was the sign of humanity to go higher and further into the universe. Indeed, mankind has been making quite many giant steps in space exploration. Thirty-one years later from this event, on October 4th, 1957, there was the witness of the dawn of the space era, the launch of the first man-made satellite of the world, Sputnik 1, by the Soviet Union, using the Sputnik rocket (Garber, 2007). Only one year after that, on 29th July 1958, the National Aeronautics and Space Administration (NASA) was established, and formally opened for business on the first of October the same year (Dick, 2008). Under this name, the Apollo mission, which is the huge

mission of discovering the universe and bringing human beings into space, was selected in 1962 (Loff, 2015), with the first test was done in 1967 (NASA, 2012). Numerous of other space projects were also launched in the same period of time, to observe the earth, to discover the space, or to go to another planet. Those activities attracted many scientists to be involved. In 1969, after two years since the Apollo project was started, NASA succeeded to make a giant leap of the conquest of space: the Apollo 11 spacecraft landed the first man on the moon (NASA, 2014).

When people thought bigger and wanted to make things bigger, the time of international cooperation was started. As the result, the collaboration of several nations in the program of the International Space Station, which is known as the ISS, was born at the beginning of 1998 (Wild, 2011). Figure 1 is the configuration of the ISS, which can be easily recognised as the combination of a number of separate parts.

The birth of orbital rendezvous had already happened in the 1960s, the era of the space race between the United State and the Soviet Union (Woffinden & Geller, 2008). However, space rendezvous has become crucial since the assembly of the ISS. This is the forming of the huge space-flight from many modules, those can be seen in Table 1 (Aeronautics, 2010). This growth has taken more than a decade, with many rendezvous and docking activities, and is still planned to add more in the future.

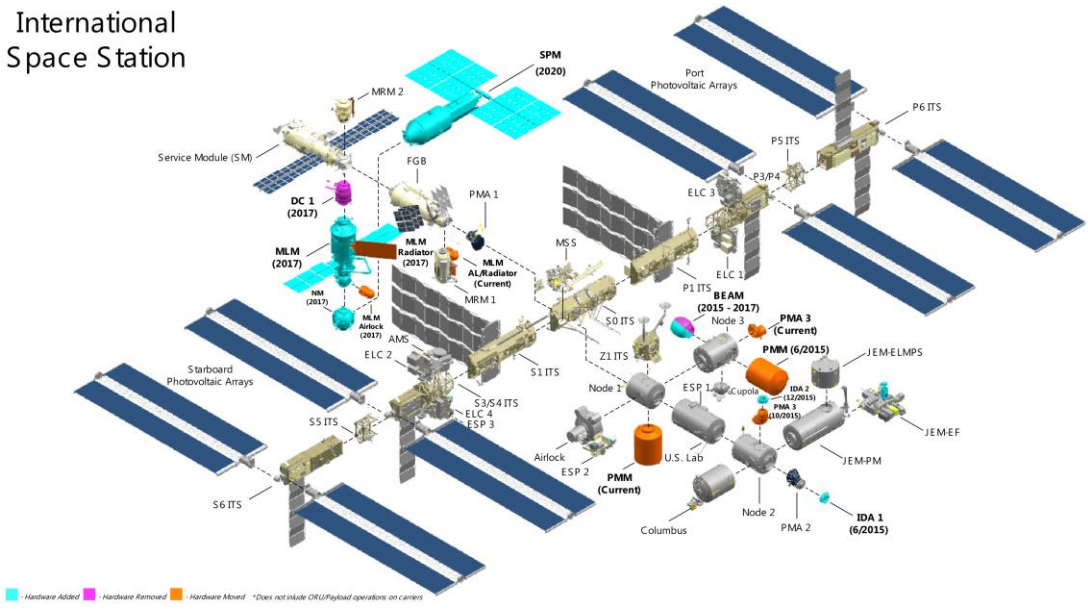


Figure 1 ISS configuration[1]

[1] Image credit from webpage www.altecspace.it (https://www.altecspace.it/wp-content/uploads/2015/05/ISS-NEW-CONFIGURATION1.png)

Table 1 ISS modules list[2]

Components of the ISS (continued)

Module	Length	Launched	Module	Length	Launched
Zarya	12.8 m (42 ft)	1998	Node 2	6.1 m (21 ft)	To be launched
Unity	5.5 m (18 ft)	1998	Columbus	6.9 m (22.6 ft)	To be launched
Zvezda	13.1 m (43 ft)	2000	Experiment Logistics		
Z1 Truss	4.6 m (15 ft)	2000	Module (ELM)		
P6 Truss	18.3 m (60 ft)	2000	Pressurized Section (PS)	3.9 m (12 ft)	To be launched
Solar Array	73.2 m (240 ft)	2000	Dextre	3.5 m (11.4 ft)	To be launched
Destiny	8.5 m (28 ft)	2001	Kibo	11.2 m (36.7 ft)	To be launched
Canadarm 2	16.9 m (56 ft)	2001	S6 Truss	13.7 m (45 ft)	To be launched
Quest Airlock	5.5 m (18 ft)	2001	ELM Exposed Section	4.9 m (16.1 ft)	To be launched
Pirs Airlock	4.9 m (16 ft)	2001	Kibo Exposed Facility	5.6 m (18.4 ft)	To be launched
S0 Truss/Mobile			Russian Multi-Purpose		
Transporter	13.4 m (44 ft)	2001	Laboratory Module	12.8 m (42 ft)	To be launched
Mobile Base	5.8 m (19 ft)	2002	Node 3	6.1 m (21 ft)	To be launched
S1 Truss	13.7 m (45 ft)	2002	Cupola	3 m (9.8 ft)	To be launched
P1 Truss	13.7 m (45 ft)	2002	Russian Research		
P3/P4 Truss	13.7 m (45 ft)	2006	Module	12.8 m (42 ft)	To be launched
P5 Truss	3.3 m (15 ft)	2006	Soyuz	7 m (22.9 ft)	Ongoing
S3/S4 Truss	13.7 m (45 ft)	To be launched	Progress	7.4 m (24 ft)	Ongoing
S5 Truss	3.3 m (15 ft)	To be launched			

Until now, many space rendezvous projects have been launched, including the purpose of supplying resources to the ISS. There are also many experiments of autonomous space rendezvous have been made to create new possible space missions. Figure 2 shows the flight profile of the ATV, the automated transport vehicle from the European Space Agency (ESA), in order to supply the ISS. There is one note that should be mentioned here in Figure 2, the S4 hold point at which the ATV starts its final approach to the target, the ISS (SpaceFlight101, 2016). The difference between this final step and those previous ones among the entire process is that it requires more parameters with the better accuracies. This separates the far range and the near range scenarios of space rendezvous that will be discussed in section **1.3**. In Figure 3, there is the picture of the ATV-5 when it was going to dock with the ISS.

[2] Table credit from NASA

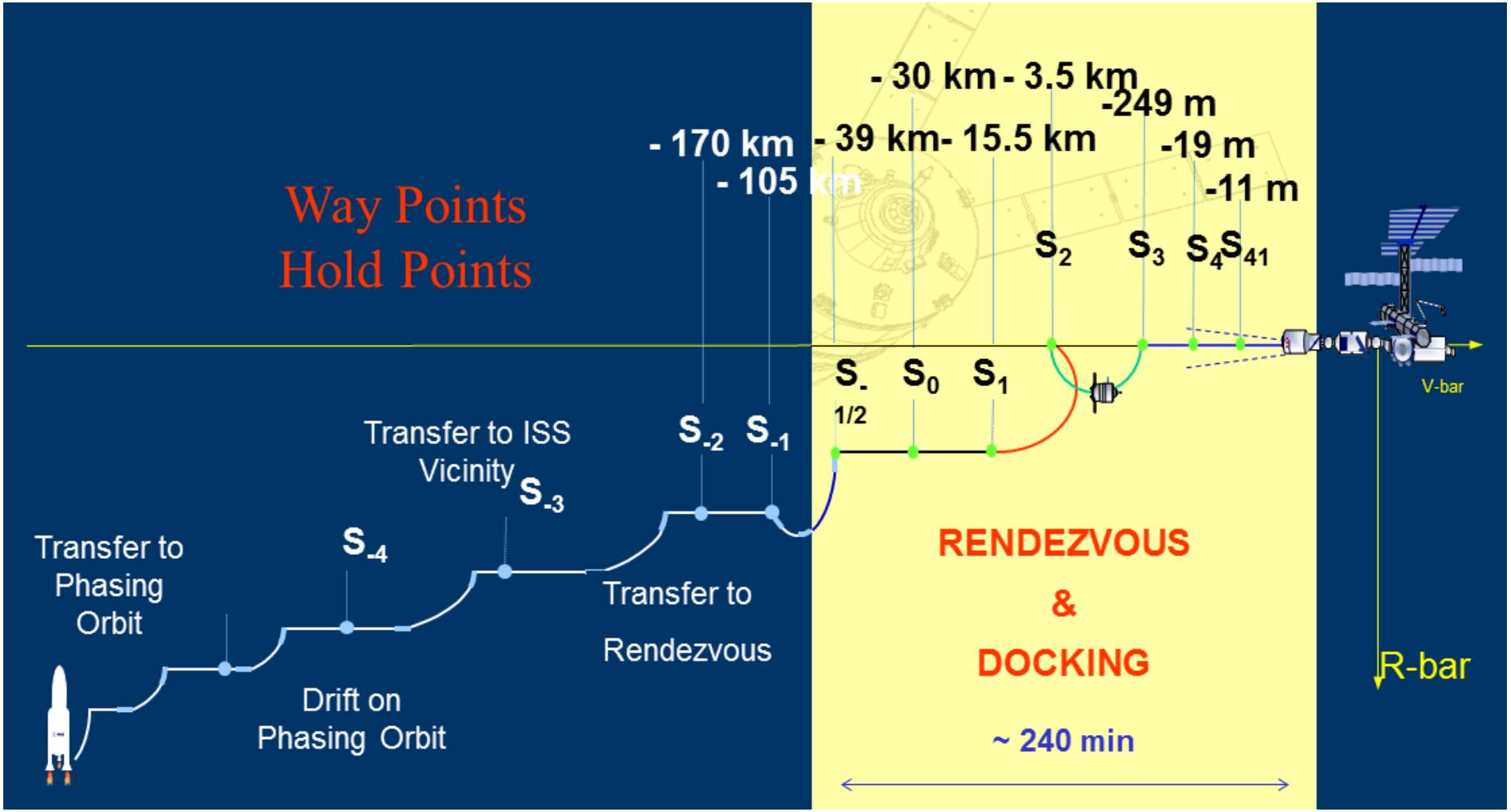


Figure 2 ATV flight profiles in order to supply the ISS[3]

[3] Image credit of ESA

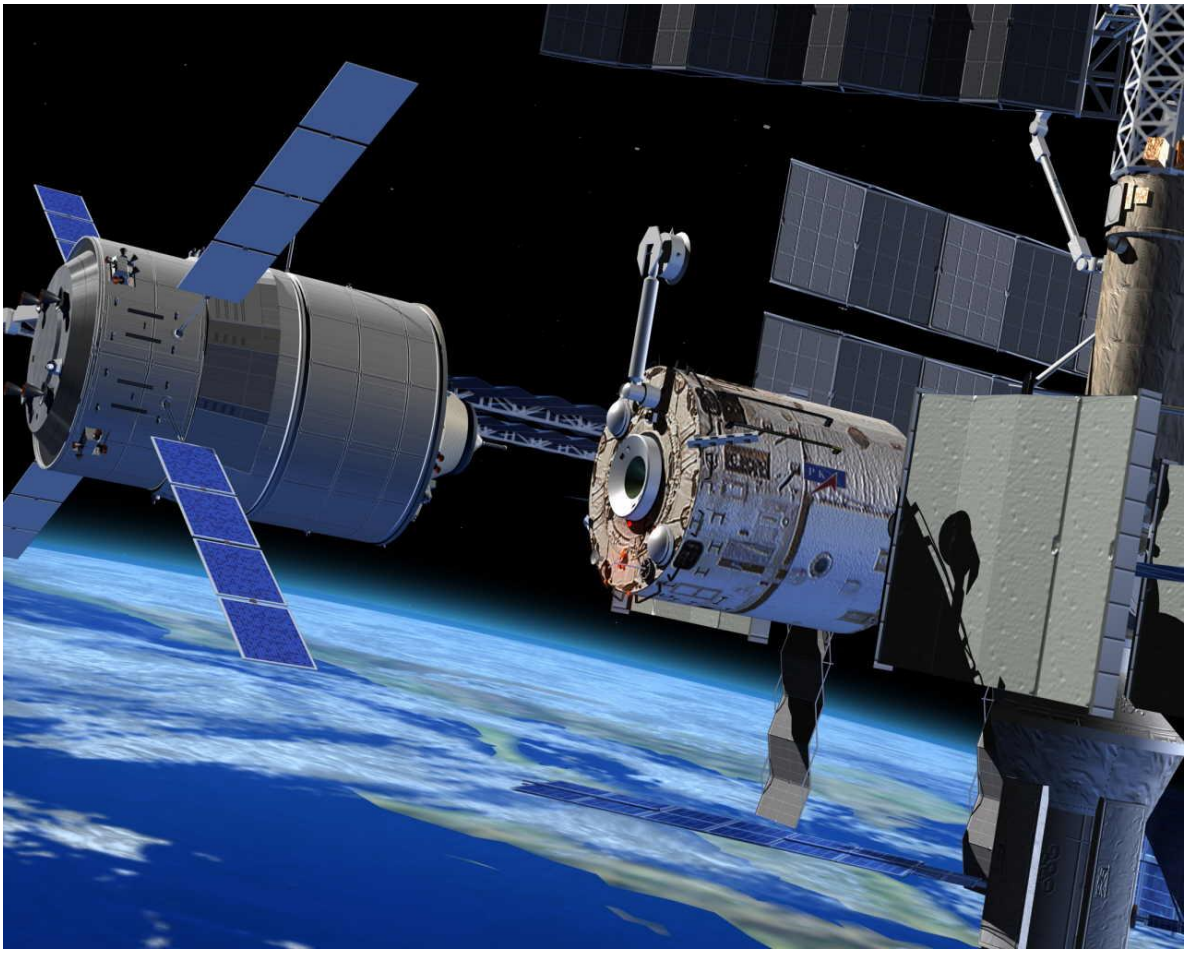


Figure 3 ATV-5 in docking with the ISS[4]

In recent years, these rendezvous activities are growing so fast, especially, under the international cooperation. At the first place, there was the need of astronauts in tracking and controlling a spacecraft to approach the space rendezvous missions, then there came a requirement of autonomous space rendezvous, which does not need any people to be involved. This attempt would reduce a lot of human resource's risks, which ideally should be zero. There are many types of sensors and sensing systems have been used in autonomous space rendezvous, they will be shown and discussed in **Chapter 2** of this report.

[4] Image credit from www.airbusdefenceandspace.com (https://airbusdefenceandspace.com/wp-content/uploads/2015/03/atv-5.jpg)

1.2 Relative Satellite Navigation

Satellites and spacecraft have very complicated movements in space, including orbiting around a space body and rotating itself. The most important factor in space rendezvous is the relative navigation, which is different from the absolute navigation. By knowing this information, one satellite knows the other's position and attitude with reference to its coordinate system.

According to Wigbert Fehse, 2003, in his book, "Automated Rendezvous and Docking of Spacecraft", chapter 7 – Sensors for Rendezvous Navigation pp. 218-282, the absolute state of a satellite, such as the attitude with reference to its local orbital frame and the position with reference to the Earth-centred equatorial frame, should be required and measured for every satellite. But in the scenario of space rendezvous, when the contact with another satellite is needed, relative position and attitude of one need to be calculated with reference to the other's local orbital frame. Therefore, one (normally the chaser) knows about the moving state of the other (in this case, the target) with reference to its coordinate system, and then, can be able to chase and, if required, merge with its target.

The relative navigation can be detected by differentiating the absolute measurements from the information of both the chaser and the target. However, this way would lead to large errors. There is a probability that each absolute measurement already has its own error(s). Therefore, in the combination of noises from transmitting environment between the two spacecraft, the errors can only be worsened. As the two spacecraft are getting closer, a continuously increased accuracy of the relative position and attitude of one with reference to the other is necessary. That is the reason why a direct measurement for relative navigations (a sensing system) between the two needs to be made separately with the absolute navigation system

1.3 Space Rendezvous Scenario

Talking about space rendezvous, there are some scenarios need to be distinguished based on the differences in their requirements. These requirements are various from mission to mission and also are different from the scenario in which the space rendezvous activity takes place.

1.3.1 Target Type

As introducing in section **1.1**, in a space rendezvous, there are usually two objects: the chaser and the target. Two following types of targets: the cooperative and the non-cooperative will make different requirements to the rendezvous sensing system.

1.3.1.1 Cooperative Target

The cooperative type of targets is the option in which the two satellites are designed to merge with the other at the particular surface of each satellite. They have a specific mechanism that provides the chaser with the ability to recognise and estimate the position of the target. By using their absolute navigating sensors (such as GPS sensors, star tracker sensors, sun sensors, earth sensors), and by exchanging their absolute navigating information, they always can point themselves to the same reference (for example, the Earth or the Sun) to make sure that the merging surface of the target is visible to the chaser. From this achievement, the chaser only needs a sensing system from a certain distance to determine the position of the target with more accurate parameters.

An example of a cooperative target is the ISS. Every time a spacecraft needs to get to the ISS, it knows at which position the docking port of the ISS is pointing to, and by which direction it has to approach to get to the port.

1.3.1.2 Non-cooperative Target

The non-cooperative type of targets is the option in which the chaser is not designed to know the navigating information of the target. This is difficult for the chaser to estimate even the low accurate relative position of the target.

One example of this type of targets is the space debris, which can be a retired satellite or some spontaneous flying object. For science study purpose, there are small space bodies, which do not have strong enough gravity to pull a satellite down on their surfaces, such as comets or asteroids, also considered as non-cooperative targets. Autonomous space rendezvous take place in the process of landing a satellite with specific equipment on the

surface of these space bodies. In those non-cooperative target examples, the position of the target cannot be estimated, but the requirements to the sensing system are quite more relaxed.

There is an uncertain scenario in which the chaser has to catch the non-cooperative target with the same requirements as they were designed to be cooperative. That makes the difficulty higher, but this scenario only happens in the emergency cases, for example, when the target lost its control, or when there are some errors happened with a satellite and it needs another satellite to come, dock and fix its problems.

1.3.2 Merging Type

After the space rendezvous, the way by which the two satellites make a physical contact with each other can be considered having two options: autonomous docking and berthing using robot arms (ESA, n.d.).

The requirement of accuracies in the autonomous docking scenario is much higher than the other, especially in the term of the relative attitude determination. The berthing is the action of one satellite using a robot arm to literally catch the other satellite. It can happen from a distance of about 1 metre. The autonomous docking action requires precise relative navigation parameters to be provided. The chaser will use those parameters to correctly approach and merge with the docking port of the target.

1.3.3 Range scenario

A rendezvous sensing system starts its mission from approximate hundreds of metres to the beginning of a physical contacting action. But this process can be divided into two scenarios: far range rendezvous and near range rendezvous.

As the distance changes, the required navigation parameters will also be different, thus, the algorithm used for the calculations also needs to be changed. The switching point chosen here is the distance of 20 metres. This can be considered the starting point of the final approach of the space rendezvous, according to the ATV flight profile (SpaceFlight101, 2016) once mentioned in section **1.1**.

Far range rendezvous is the case in which the distance between the two satellites is longer than 20 metres, up to the range 200 metres. This range varies, depending on the frequency, the focal length, the resolution, the field of view and some other characteristics of the camera itself. However, it is expected to be around 200 metres or more. In this scenario, the chaser satellite only needs to know the distance and the direction (LOS angles) of the target.

Near range rendezvous scenario is the case in which the distance between the two satellites is smaller than 20 metres. As mentioned before, this distance is considered as the division point, from that the final approach starts. Therefore, from here, simultaneously with the distance and the direction, the relative attitude of the target should be also detected. The chaser uses these six-degree-movement parameters to get to the target correctly. The high accurate relative attitude is most important in the case of autonomous docking.

1.4 Measurement Parameters

In principle, the sensing system can be implemented on either the chaser or the target, and sometimes it is distributed on both of them. However, a mechanical part of the chasing process must be put on the chaser, because the chaser needs to possess the final navigating results to react towards the target. Therefore, it is better to put the calculating part of the sensing system on the chaser, to avoid the error(s) and noises from the transmitting environment. This calculating mission will gather information from the sensor part of the sensing system and compute the values of the navigating parameters. These parameters are the outputs of the sensing system. The final destination of these outputs is the attitude control subsystem (ACS). They can be sent through the onboard computer subsystem (OBC), the ACS onboard computer (AOBC), or directly to the ACS. The choice depends on the configuration of the bus system of the satellite itself. The ACS then controls the satellite based on this information.

The necessary relative parameters that need to be calculated and provided by the sensing system are (Fehse, 2003):

The distance or range: the physical long distance between the two vehicles (chaser and target).

Range-rate: can be considered as the relative linear velocity of the target with reference to the chaser (can be referred from comparing the distances at some different time spots to find the changing amount). It is not necessary if the range can be provided in a near-real time.

The line-of-sight (LOS) angles or direction: the direction of the target's centre with reference to the chaser's local frame. This is the 2D position of the centre of the target in the focal plane of a camera, which is its two-axis coordinate system. The direction normally is represented by the two-axis angles (horizontal and vertical angles).

The relative attitude: the 3D attitude of the target spacecraft with reference to the chaser's three-axis coordinate system. The attitude normally is represented by the three-axis angles (the three orientations around the three axes).

Angular rate: relative angular velocity of the target, that makes the rotation, with reference to the chaser's coordinate system (can be referred from comparing the attitudes at some different time spots to find the changing amount). It is not necessary if the attitude can be provided in a near-real time

These measurement parameters are required differently for each scenario in which a space rendezvous takes place. Based on the description of space rendezvous scenarios in section **1.3**, the technical requirements to a rendezvous sensing system can be clarified:

- For far range scenario, the three parameters: the distance and the two angles of LOS direction are necessarily provided by the rendezvous sensing system:
    - High accuracies are required for cooperative targets, emergency docking.
    - Medium accuracies are required for debris collecting, space body landing.
- For near range scenario, the six parameters: the distance, the two angles of LOS direction and the three angles of orientation attitude are necessarily provided by the rendezvous sensing system:
    - High accuracies are required for autonomous docking.
    - Medium accuracies are required for berthing, debris collecting, space body landing.

1.5 Thesis layout

This report will continue with the problem definition by **Chapter 2**, including the satellite trend, the introduction to current systems along with their existing problems especially in the case of small satellites, and the literature review. After that, **Chapter 3** will describe the scope of this research, introduce TOF camera, and report the progress of designing the concept of the new sensing system for autonomous space rendezvous. **Chapter 4** shows the prototyping, the configuration, and the results of the tests in order to verify the research. Finally, **Chapter 5**, based on the test results and analyses, concludes these works and proposes the future works to continue in a deeper study and a higher level of research.

2 Problem Definition

In this chapter, firstly, the trend of small satellites in current time will be considered **(2.1)**, then, some current methods and examples are shown as the introduction to existing sensing systems for autonomous space rendezvous **(2.2)**. Afterwards, in the next section **(2.3)**, the literature review of small devices, with time-of-flight principle, related to space rendezvous, will be discussed. Finally, in the last section of this chapter **(2.4)**, some problems that still exist, as well as the limitation of the current methods, will be shown. That is also the reason to do this research.

2.1 Satellite Trend

Space technology and industry have been growing faster every day since the last century. As time goes by, there are more satellites have been launched everyday with many different purposes and missions. There are no signs showing that people are going to stop thinking about creating their new space tasks. However, there is one factor that should not be forgotten. It has to be concerned throughout the whole progress of a plan to make a satellite. That is the launching system, including the cost and the vehicle. It is not difficult to imagine that the bigger the satellite to be made, the stronger the rocket to be chosen, and the more expensive the launching cost to be spent. Technically speaking, the launching cost is roughly estimated at about fifteen thousand euros (15,000€) for one kilogram of the satellite (Koelle & Janovsky, 2007; Wilfried Ley, Klaus Wittmann, 2009). More details about the calculation for the launching cost is given in Table 2. This table is about the historical launch vehicle costs (Wertz, Everett, & Puschell, 2011). Other than that, a big launch also incidentally affects to the natural resources in the term of the amount of propellant or other materials using in the launching event.

Many commercial space transportation companies, such as SpaceX (www.spacex.com), are working so hard to find the new techniques and solutions that can be able to reduce the cost of launching a spacecraft. There was a big event showing their attempts at doing this purpose that is quite popular these days: the experiments of the reusable rockets (SpaceX, 2015). This is one of the examples of ways to decrease the launching cost, as well as to prevent the waste of rocket's material using in the launch.

Table 2 Historical launch vehicle costs for predicting SME-SMAD WBS 2.0 Cost[5]

Mass Class (a)	Vehicle Name	Country of Origin	Capacity Payload in kg to Orbit		Average Launch Cost (FY2010 $K) (d)	Launch Cost in FY2010 $K per kg Placed into Orbit (e)	
			LEO (b)	GTO (c)		LEO (b)	GTO (c)
			Small				
Small	***Athena 2***	USA	2,065	590	$32,688	$15.8	$55.4
Small	***Cosmos***	Russia	1,500	N/A	$18,387	$12.1	N/A
Small	***Pegasus XL***	USA	443	N/A	$18,454	$41.5	N/A
Small	***Falcon 1e***	USA	1,010	N/A	$10,900	$10.8	N/A
Small	***Minotaur IV***	USA	1,650	N/A	$22,000	$13.3	N/A
Small	***Rockot***	Russia	1,850	N/A	$18,454	$9.9	N/A
Small	***Shtil***	Russia	430	N/A	$272	$0.7	N/A
Small	***START***	Russia	632	N/A	$10,215	$15.9	N/A
Small	***Taurus***	USA	1,380	448	$25,878	$18.8	$57.7
			Medium and Intermediate				
Inter.	***Ariane 44L***	Europe	10,200	4,790	$153,225	$15.0	$32.0
Inter.	***Atlas 2AS***	USA	8,618	3,719	$132,795	$15.4	$35.7
Medium	***Delta 2***	USA	5,144	1,800	$74,910	$14.6	$41.7
Medium	***Dnepr***	Russia	4,400	N/A	$20,430	$4.6	N/A
Inter.	***Falcon 9***	USA	10,450	4,540	$56,750	$5.4	$12.5
Medium	***Long March 2C***	China	3,200	1,000	$30,645	$9.5	$30.6
Inter.	***Long March 2E***	China	9,200	3,370	$68,100	$7.4	$20.2
Inter.	***Soyuz***	Russia	7,000	1,350	$51,075	$7.4	$37.9
			Heavy				
Heavy	***Ariane 4G***	Europe	18,000	6,800	$224,730	$12.5	$33.1
Heavy	***Atlas 5***	USA	20,050	8,200	$172,000	$8.6	$25.3
Heavy	***Delta 4 Heavy***	USA	22,560	13,130	$215,000	$9.5	$16.4
Heavy	***Long March 3B***	China	13,600	5,200	$81,720	$6.0	$15.7
Heavy	***Proton***	Russia	19,760	4,630	$115,770	$5.9	$25.1
Heavy	***Space Shuttle***	USA	28,803	5,900	$408,600	$14.2	$69.2
Heavy	***Zenit 2***	Ukraine	13,740	N/A	$57,885	$4.2	N/A
Heavy	***Zenit 3 SL***	Multinational	15,876	5,250	$115,770	$7.2	$22.1

NOTES: (a) The FAA Office of Commercial Space Transportation has divided all vehicles into four mass classes: small, medium, intermediate, and heavy, based on their payload capacities in pounds to LEO (<5,000, <12,000, <25,000, >25,000); in the table, all weights have been converted to kg. (b) Low Earth Orbit (c) Geosynchronous Transfer Orbit; these costs do not include the additional cost of apogee kick motors or other payload injection means. (d) These costs (in 2010 thousands of dollars) include the launch vehicle and related launch services. Where necessary, the costs were converted to dollars from other currencies. (e) Assuming maximum use of the payload weight capacity in the calculations, these are minimum values.

At the same time, however, on the satellite side itself, the satellites' owners are also thinking about the new concepts that would save their money from the launching activities. One of the solutions is making things smaller and lighter, in order to reduce the size and weight of a satellite. The reason is that these factors directly affect the launching cost. The weight, as it can be seen obviously in Table 2, is the basic unit to calculate the launching cost. That makes the trend of two small satellites. One class of these satellites is the micro-satellites. They are small satellites, within one metre cube size and within 10 to 100 kilograms weight (SpaceWorks Enterprises, 2014). These small size and low weight satellites are easy to launch, because they don't require the particular launching vehicle, therefore, they don't give as many

[5] Table credit of The New SMAD book: Space Mission Engineering, chapter 11 – Cost Estimating

constraints to the launching system as those big spacecraft, and they don't take the huge launching costs. The result is: it would probably take a less amount of time for these satellites to wait until there is an available session that is suitable for launching them. That is the reason why instead of launching the huge spacecraft, people are thinking about launching multiple small satellites. Figure 4 shows the general view of this trend (Rast, Schwehm, & Attema, 1999).

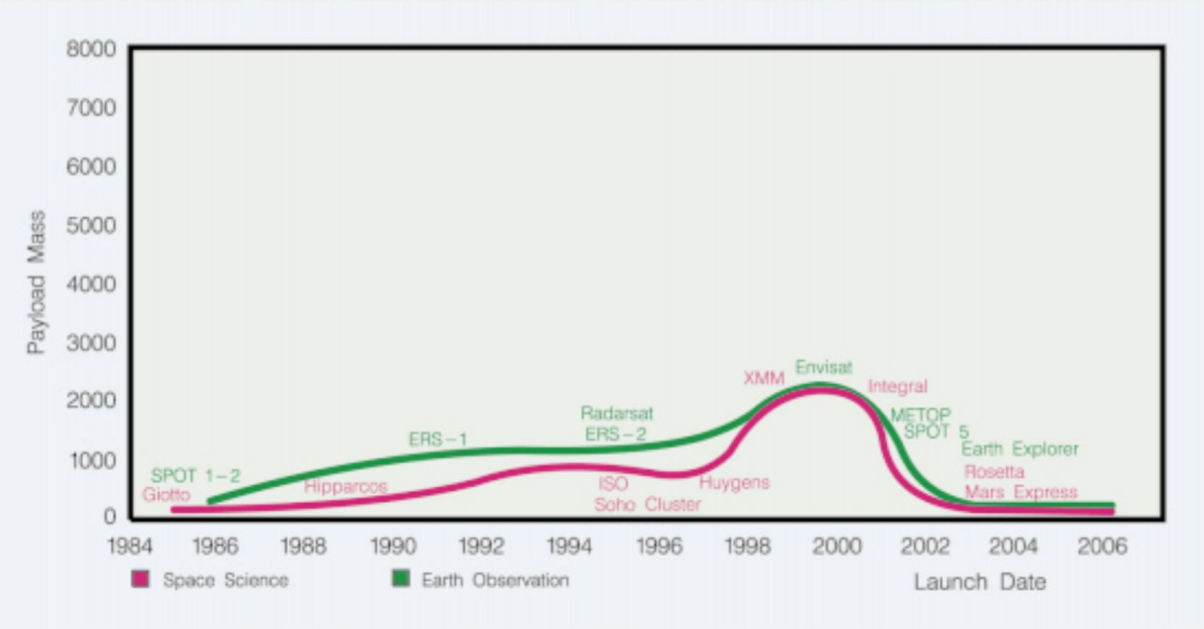


Figure 4 Payload-mass histories for science and earth observation satellites[6]

To make a clear image about the trend of small satellites, there is a conclusion from the journal article "Payload-Mass Trends for Earth-Observation and Space Exploration Satellites" written in the year 1999 by M. Rast, G. Schwehm and E. Attema from ESA Directorate for Scientific Programmes, ESTEC, Noordwijk, The Netherlands (Rast et al., 1999):

"Looking to the future of space exploration, with mankind pushing further and further into deep space and possibly visiting other planets, the demand for knowledge and the resulting requirements will become even more exacting. The size of the individual missions could be reduced by splitting up the payload complements to allow smaller, dedicated and more focused spacecraft to be flown."[7]

[6] http://www.esa.int/esapub/bulletin/bullet97/rast.pdf

[7] http://www.esa.int/esapub/bulletin/bullet97/rast.pdf

For the complementation to this vision, the miniaturization of satellites (e.g., the micro class of satellites and the nano class of satellites) is considered, by many scientists and engineers working in the space industry, one of the two key trends that have the potential to revolutionize the way human conduct spacecraft (Amico et al., 2015), the other key trend is the distribution of payload tasks among multiple coordinated units. It means that the miniaturising of satellites has more advantages rather than just reducing the launching cost.

2.2 Introduction to Existing Sensing Systems

Following the demand of autonomous rendezvous in space activities, there are several sensing methods have been studied and designed, as well as some sensing systems have been created, to achieve the purpose. This part is going to introduce some of the most efficient methods and sensing systems were developed by some famous space organisations.

The content of this section is mainly referred from "Chapter 7 – Sensors for Rendezvous Navigation" in the book "Automated Rendezvous and Docking of Spacecraft" written by Wigbert Fehse, 2003, published by Cambridge University Press (Fehse, 2003).

2.2.1 RF Sensors

A sensing system uses radio frequency (RF) type of sensors is the system formed by several antennas and transponders, based on the characteristics and the behaviours of radio frequency to exchange information between the two satellites.

2.2.1.1 General Principle

In principle, the distance from the target to the chaser can be measured by two ways: case number one – by measuring the time of flight of the signal, Figure 5; case number two – by measuring the phase shift between the incoming (reflected) signal and its corresponding outgoing one, Figure 6. Both of them require the act of recording at both transmission and reception sides in order to get the information of:

In case 1: the time of flight $(t_2 - t_1)$ of an electro-magnetic wave (the signal).

Therefore the relationship between the distance r of the two satellites, which is equalled to the travelled distance of the wave, and the time of flight will be given by:

$$t_2 - t_1 = \frac{r}{c} \tag{3}$$

Where c is the speed of light (300000000 m/s).

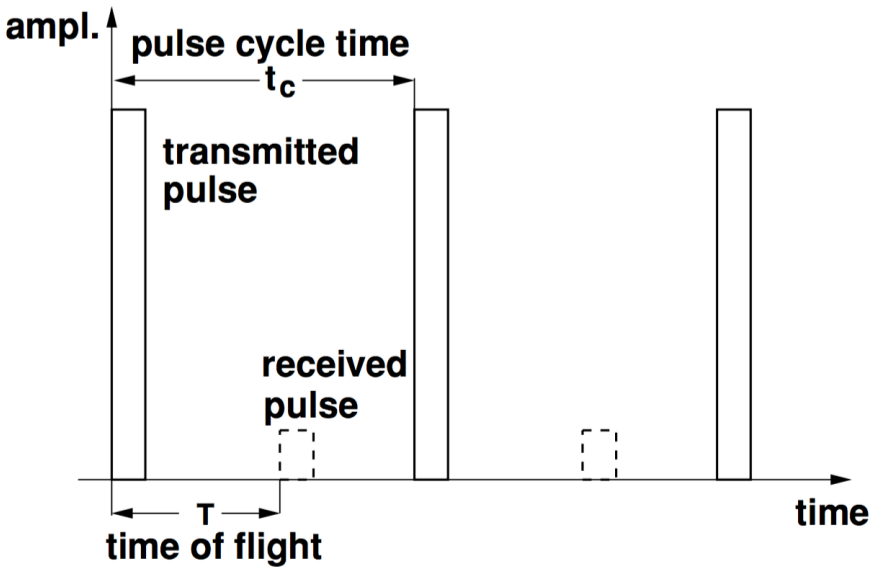


Figure 5 Functional principle of range measurement via time delay[8]

<u>In case 2:</u> the shift or the difference of the phase $(\phi_2 - \phi_1)$ between the incoming signal and the outgoing signal.

Therefore the relationship between the distance r of the two satellites which is equalled to the travelled distance of the wave and the shift phase will be given by:

$$\phi_2 - \phi_1 = \frac{2\pi r}{\lambda} = \frac{2\pi r f}{c} \tag{4}$$

While λ is the wavelength of the signal and f is its frequency.

For the direction and the relative attitude measurements, there are many methods used for the RF type of sensors in space rendezvous. The measurement of the LOS angles or the direction information can be done by pointing of the narrow beam antennas, via time delay or phase shift or via amplitude and antenna rotation angles. The measurement of the relative attitude can be done through tone modulation and the rotating pattern on the target or by using antenna beams with different tone modulations.

[8] http://ebooks.cambridge.org/chapter.jsf?bid=CBO9780511543388&cid=CBO9780511543388A043&tabName=Chapter

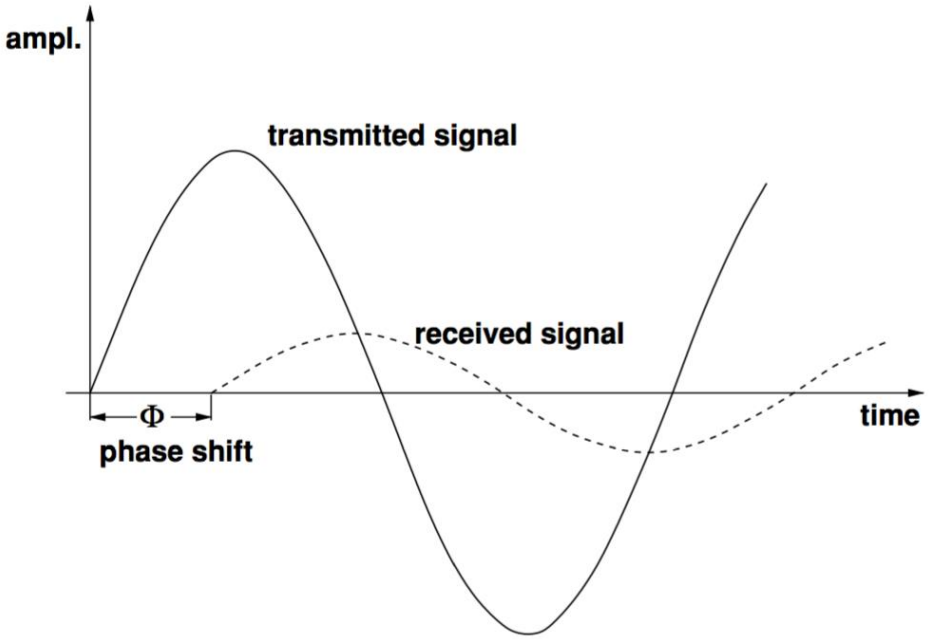


Figure 6 Functional principle of range measurement via phase shift[9]

2.2.1.2 Example: the Russian Kurs system

The Kurs (Course) system has been used for the Russian Soyuz and Progress vehicles (which carries people and supplies to the ISS, also brings people back to earth in case of Soyuz (May, 2013) and brings supplies and fuel to the ISS and raises the aptitude or control the orientation of the space station in case of Progress (Wright, 2015)) with the purpose of space rendezvous navigation for a long time, in order to approach to first the Russian space station Mir, and then the ISS. It is the combination of various radio wave principles and components to form one navigation system (Fehse, 2003).

Radio frequency type of sensors - Antennas

The Kurs radio engineering system uses an S-band radio frequency, i.e. short wave varies from 2 to 4 GHz (IEEE, 2009), comprises active functions on both the chaser and the target. The placements of antennas used for space rendezvous navigation mission on each vehicle are shown in Figure 7 below.

There are five antennas on each spacecraft.

On the chaser (Soyuz or Progress):

[9] http://ebooks.cambridge.org/chapter.jsf?bid=CBO9780511543388&cid=CBO9780511543388A043&tabName=Chapter

A1, A2 These are Omni-directional antennas used for range and range-rate determination. Both can transmit (frequency f_1) and receive (frequency f_2) radio signals.

A3 This is a wide-angle mechanical scanning antenna with a special beam pattern. It only receives signals (frequency f_2) from the target and is used in LOS angles determination.

A4 This is a fixed antenna, can both transmit f_1 and receive f_2 signals and is used for range, range-rate, LOS angles tracking. In addition, at the distance of the proximity phase ($\leq$200m), it receives the frequencies f_3 and f_4 from the target and these signals are used for roll determination by LOS angles.

A5 This is a fixed antenna with a narrow beam only receives signal f_4 from the target. It is used for relative attitude angles determination.

On the target (Mir space station or ISS):

B1, B2 These are Omni-directional antennas, transmit the radio signal at frequency f_2 and receive the one at frequency f_1. They are corresponding to the antennas A1 and A2 on the chaser.

B3 This is a fixed antenna used in the proximity phase ($\leq$200m) to transmit the signal f_2 and also to receive the f_1 signal. The behaviour of this antenna (transmit or receive) is switched under the control of one transponder.

B4 Transmits the signal of frequency f_4, which will go to the antennas **A4** and **A5** on the chaser side. It is mainly used in the last 30m of the approach due to the requirement of high accuracy of the LOS angles and the need for the relative attitude angles determination.

B5 Transmits the signal of frequency f_3, which latterly will be received by the antenna **A4** in the chaser side.

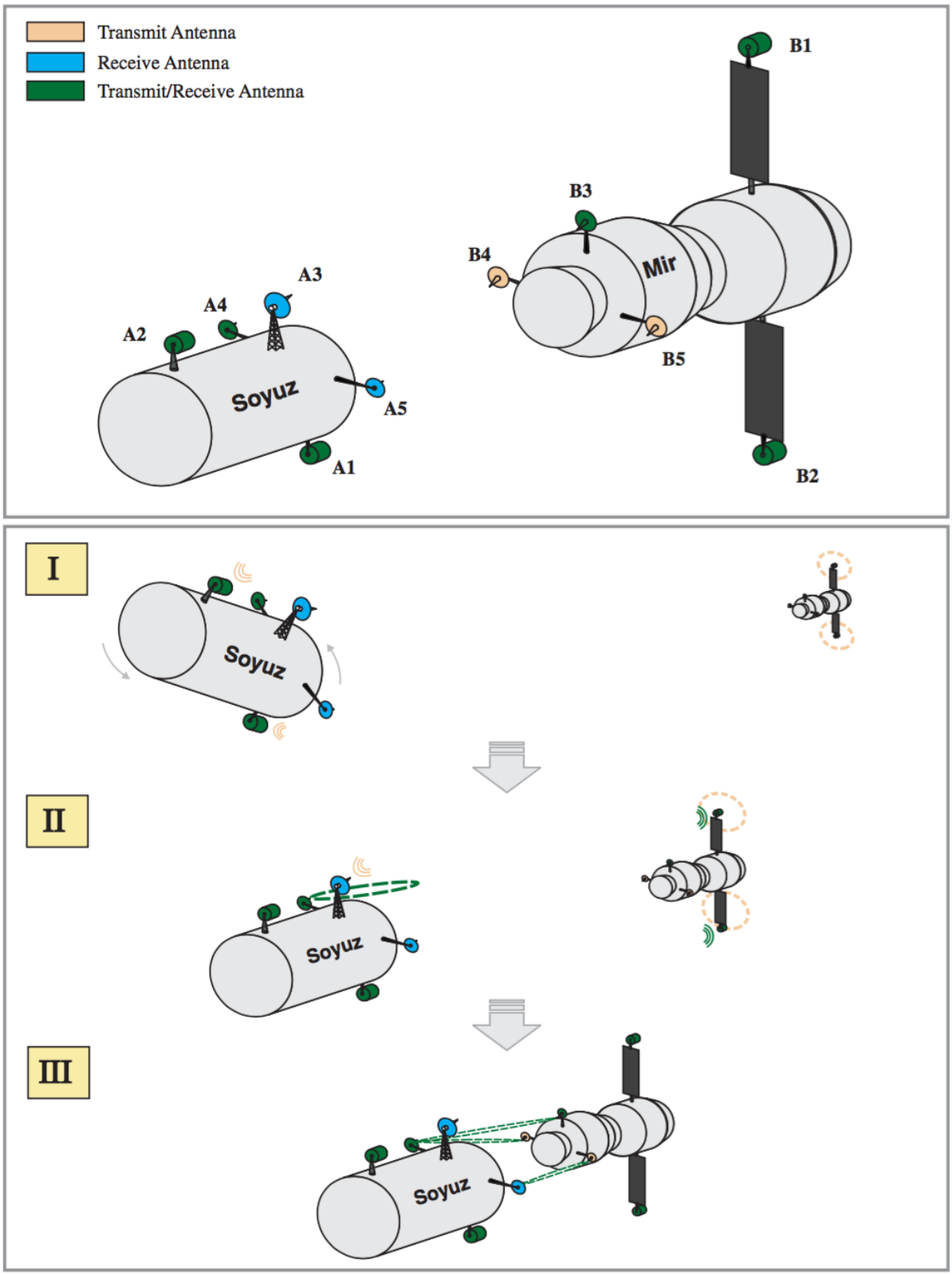


Figure 7 The Soyuz Kurs (Course) rendezvous radar system[10]

[10] http://arc.aiaa.org/doi/pdf/10.2514/1.30734

Hardware of the Kurs system on the chaser vehicle (Soyuz or Progress)

The equipment of the Kurs system on the chaser side is called the "interrogator" because it contains most of the active functions of the measurement process. In Figure 8, it is shown as the functional block diagram of the "interrogator" equipment, which is attached to the chaser vehicle, in this example, the Soyuz or the Progress spacecraft.

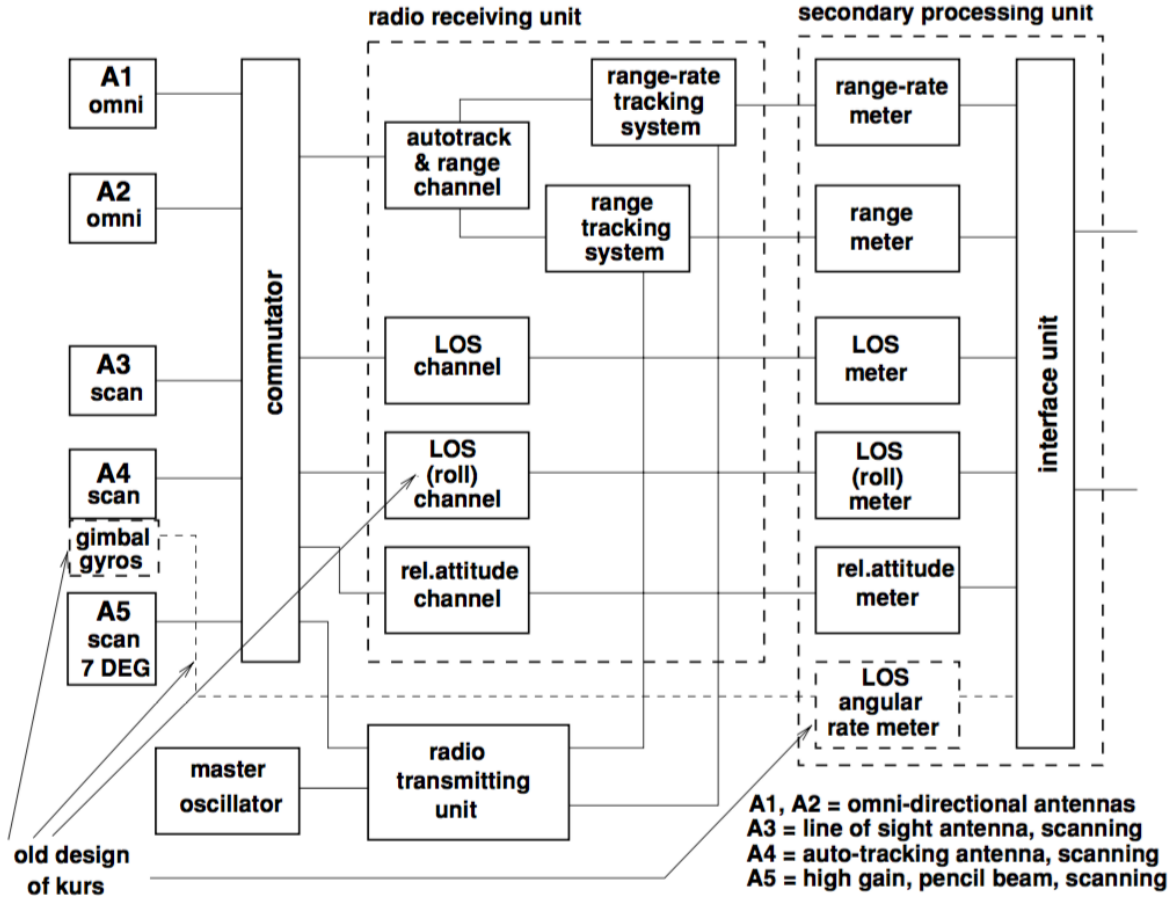


Figure 8 Functional block diagram of Kurs 'interrogator' equipment[11]

Operation of antennas during rendezvous (happens on the target side – Mir or ISS)

Depends on the appropriate distance, which is related to the measurement requirements of the rendezvous process, the antennas on the target side will be operated to send signals to the chaser through five antennas described above. In Figure 9, there is the functional block diagram of the "transponder" equipment attached on the target side, in this example is the Russian space station Mir, but it can be the ISS in another case.

[11] http://ebooks.cambridge.org/chapter.jsf?bid=CBO9780511543388&cid=CBO9780511543388A043&tabName=Chapter

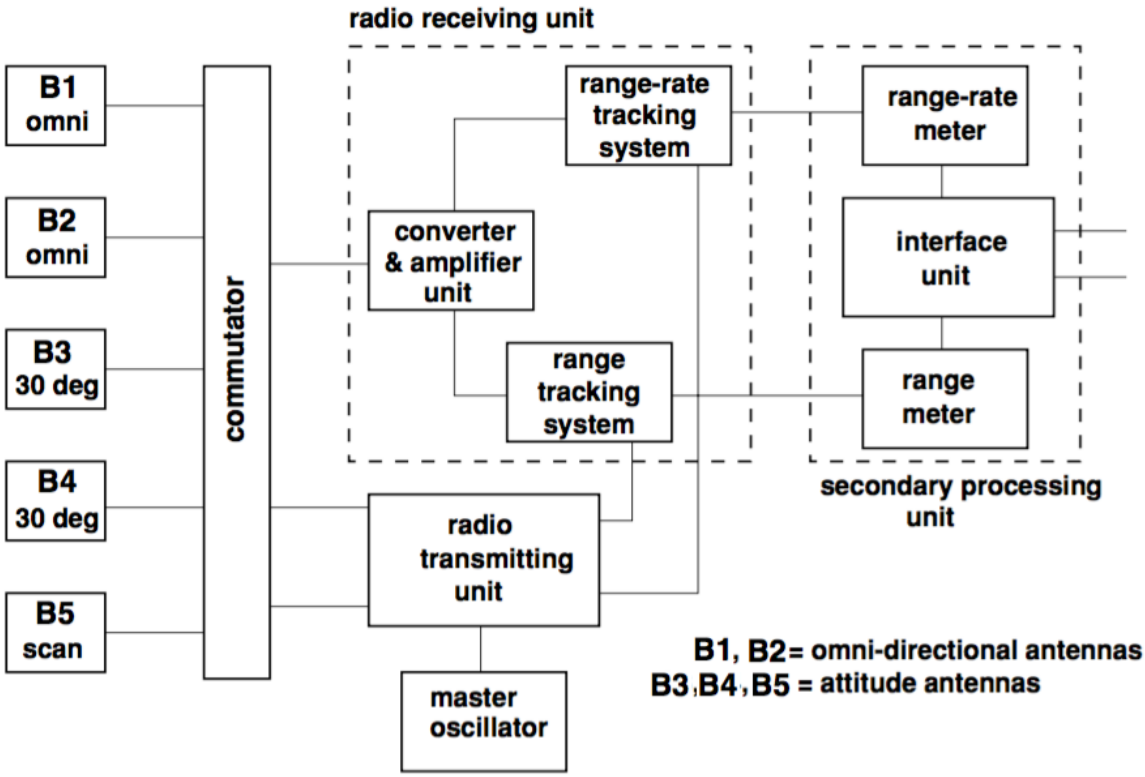


Figure 9 Functional block diagram of Kurs 'transponder' equipment[12]

2.2.1.3 Problem of RF Sensors

Through the example of the Kurs system, put into the trend of satellites, there were some problems. This is the combination of many single components (such as **10 antennas**). It used **four different radio frequencies**. This construction made the system **big**, **complex** and **difficult** in operation. This sensing system can mostly be used in **huge spacecraft**, and probably **not applied to small satellites**.

2.2.2 Optical Rendezvous Sensors

The RF type of sensors using for autonomous space rendezvous as has been shown above contains quite a lot of components (for example, ten antennas) with the complicated process (four different frequencies were used), optical type of sensors seems to be a more optimal idea.

There are two types of optical sensors discussed in this section: scanning-laser-range-finder type of sensors and camera type of sensors. Both require optical reflectors to be attached to the target as the interface.

[12] http://ebooks.cambridge.org/chapter.jsf?bid=CBO9780511543388&cid=CBO9780511543388A043&tabName=Chapter

With the advantage of image processing and other new technologies, optical sensors are expected for a large variety of sensor principles, which will be available and suitable for automated rendezvous, in the future (Fehse, 2003). Having to mention that this research is also included inside one of these varieties.

2.2.2.1 Scanning Laser Range Finder Sensor

Scanning laser range finder sensor uses the same basic physical principles with the RF radar type of sensors described above. The difference is the wavelength of the electro-magnetic signal used in each method does not equal to the others'. Radar type of sensors uses the radio wave signal while the scanning laser range finder type of sensors uses the near-infrared laser light signal.

On the chaser side, the range or distance can be detected by measuring the time of flight or the phase shift of the signal as it is done in the radar type of sensors; the direction is determined by measuring the LOS angles of the returned signal. The angles, then, can be read and used by the chaser to obtain the LOS angles ϑ and ψ to the target.

On the target side, there are numerous of reflectors will be attached as the markers (which also form the communication interface between the chaser and the target) with a task of reflecting back the transmitted signal beam from the target to the chaser.

Figure 10 shows the functional principles of the scanning laser range finder type of sensors and Figure 11 will show the target pattern, which will be used inside the chaser to calculate the needed parameters.

As it is seen, the distance can be calculated by comparing the outgoing pulse and the incoming pulse to get the phase shift or the time of flight. The two mirrors are used to tilt the light and therefore, the information of the angles of the mirrors can be known. They are then, sent to the signal processor to be processed to get the LOS angles and also the relative attitude Figure 10.

The target pattern is used mostly to calculate the relative attitude, which can be represented by the 3D angles. This relative attitude determination can be done by using some geometry calculations based on the three different distances: R1, R2 and R3 of the three different points on the pattern Figure 12.

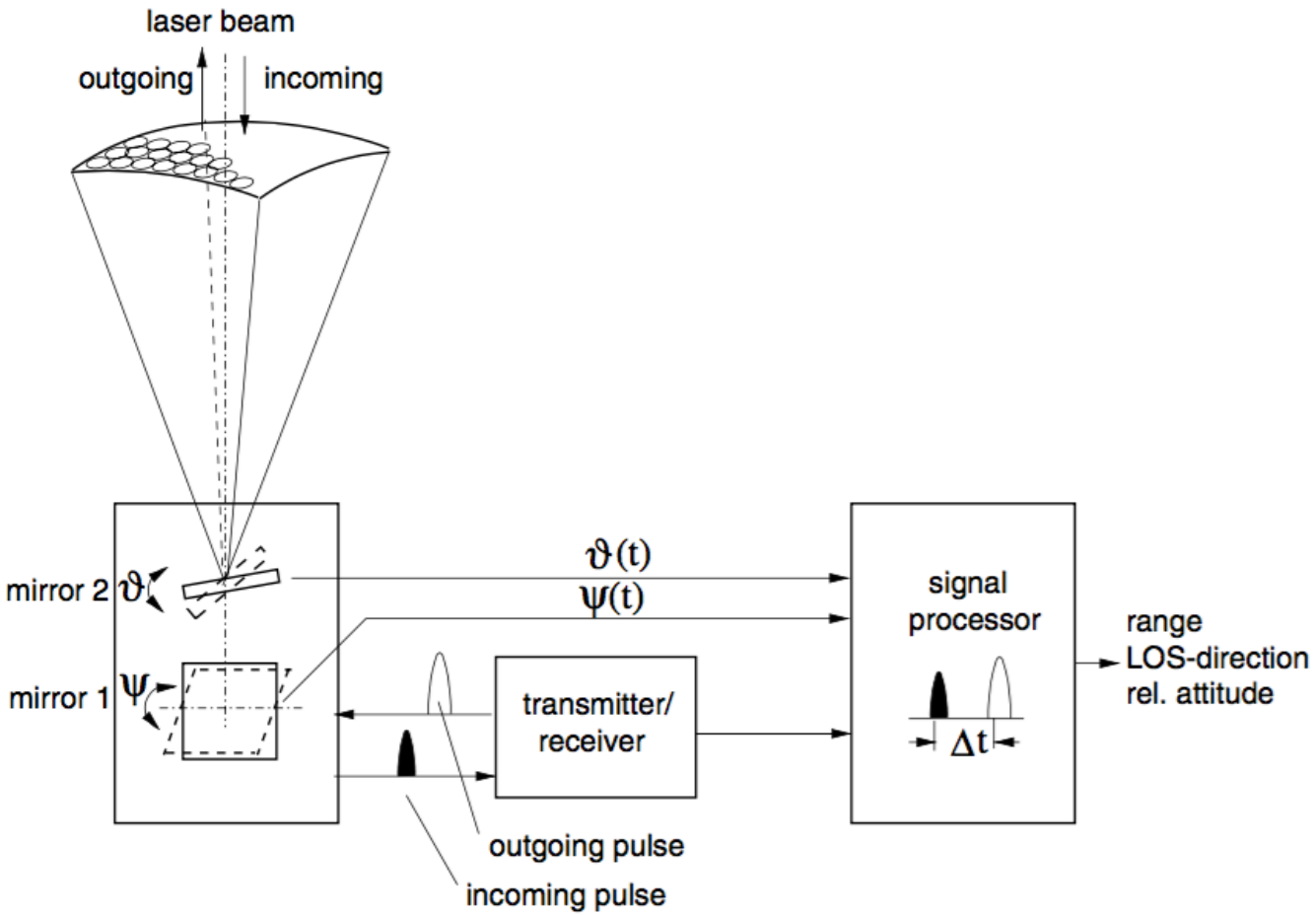


Figure 10 Functional principle of a scanning laser range finder[13]

[13]http://ebooks.cambridge.org/chapter.jsf?bid=CBO9780511543388&cid=CBO9780511543388A043&tabName=Chapter

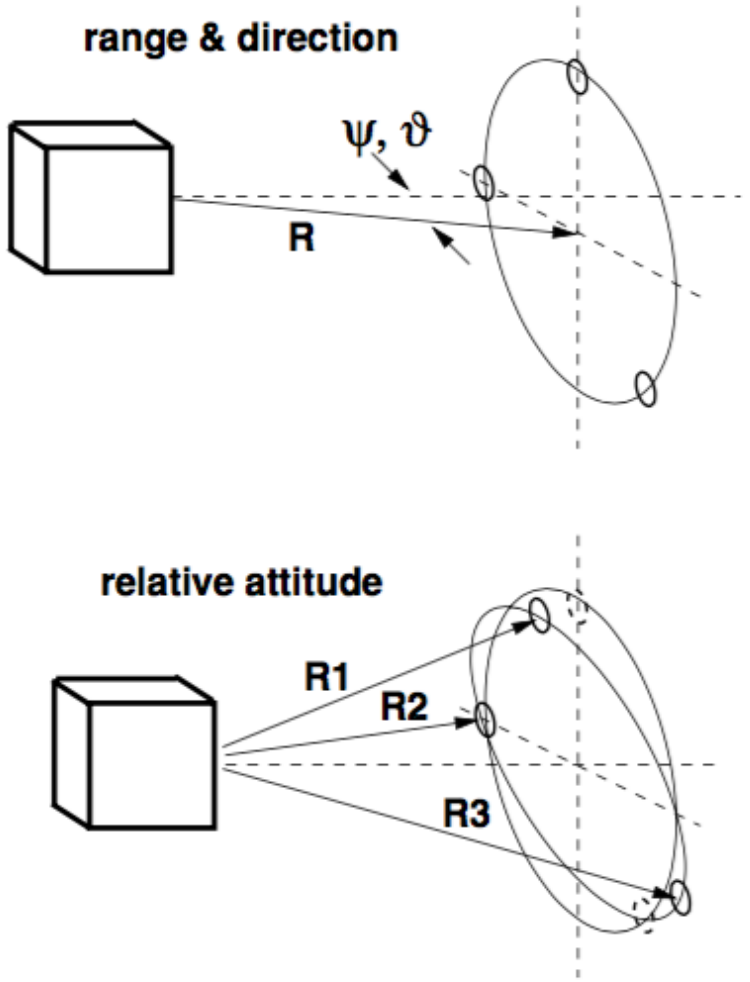


Figure 11 Target pattern for a laser range finder[14]

2.2.2.2 Camera sensor

The principle of the measurement of the camera type of sensors is based on the physical laws of image processing on the focal plane of the lens of the camera. With the advent of the CCDs (Charge Coupled Device) and the CIDs (Charge Injection Devices), which are two types of solid state charge transfer devices, compact cameras with high resolution could be built. It opened the way for an optical camera to be a solution for many sensor applications, including autonomous space rendezvous.

[14] http://ebooks.cambridge.org/chapter.jsf?bid=CBO9780511543388&cid=CBO9780511543388A043&tabName=Chapter

The basic functional principle of the camera type of rendezvous sensors is shown in Figure 12, and the target pattern is shown in Figure 13.

The distance or range, r, then, can be calculated from the known distance D between two reflectors on the target and the distance d of the images of those reflectors fell on the focal plane of the camera on the chaser, obeyed the formula below:

$$r = D.\frac{f}{d} \tag{5}$$

With f is the value of the focal length of the camera, which is assumingly known by the specification of the camera.

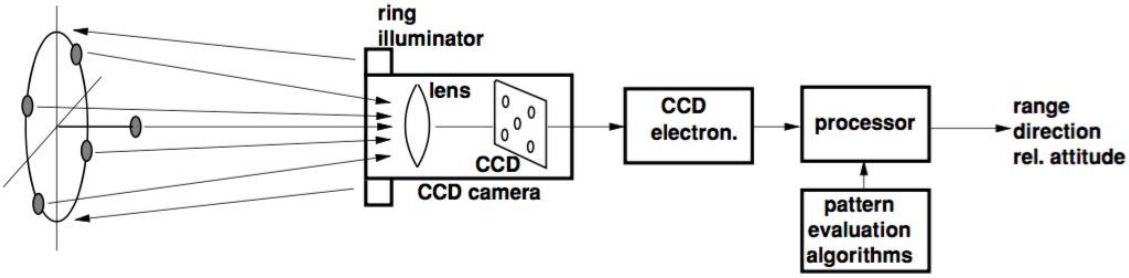


Figure 12 Functional principle of a camera rendezvous sensor[15]

The LOS angles ϑ and ψ of each reflector on the target can be determined from the two-dimensional (2D) coordinates x_{fp} and y_{fp} of the image of that reflector on the focal plane of the camera on the chaser, follow the pair of equations below:

$$\psi = \psi_{max}\frac{x_{fp}}{x_{max}} \tag{6}$$

$$\vartheta = \vartheta_{max}\frac{y_{fp}}{y_{max}} \tag{7}$$

While:

- $\pm\psi_{max}, \pm\vartheta_{max}$ are the maximum field of view (FOV) angles of the camera extended from the focal plane's centre.
- x_{max}, y_{max} are the two edge sizes of the 2D image, or the maximum of the 2D coordinates from the focal plane's centre.

From the value of those angles, the direction and also the attitude of the target can be detected as shown in the analysis of the target pattern's image below. So the relative attitude

[15] http://ebooks.cambridge.org/chapter.jsf?bid=CBO9780511543388&cid=CBO9780511543388A043&tabName=Chapter

angles here are also can be calculated based on some geometry algorithm, but probably are a bit different than the ones used in the laser range finder type.

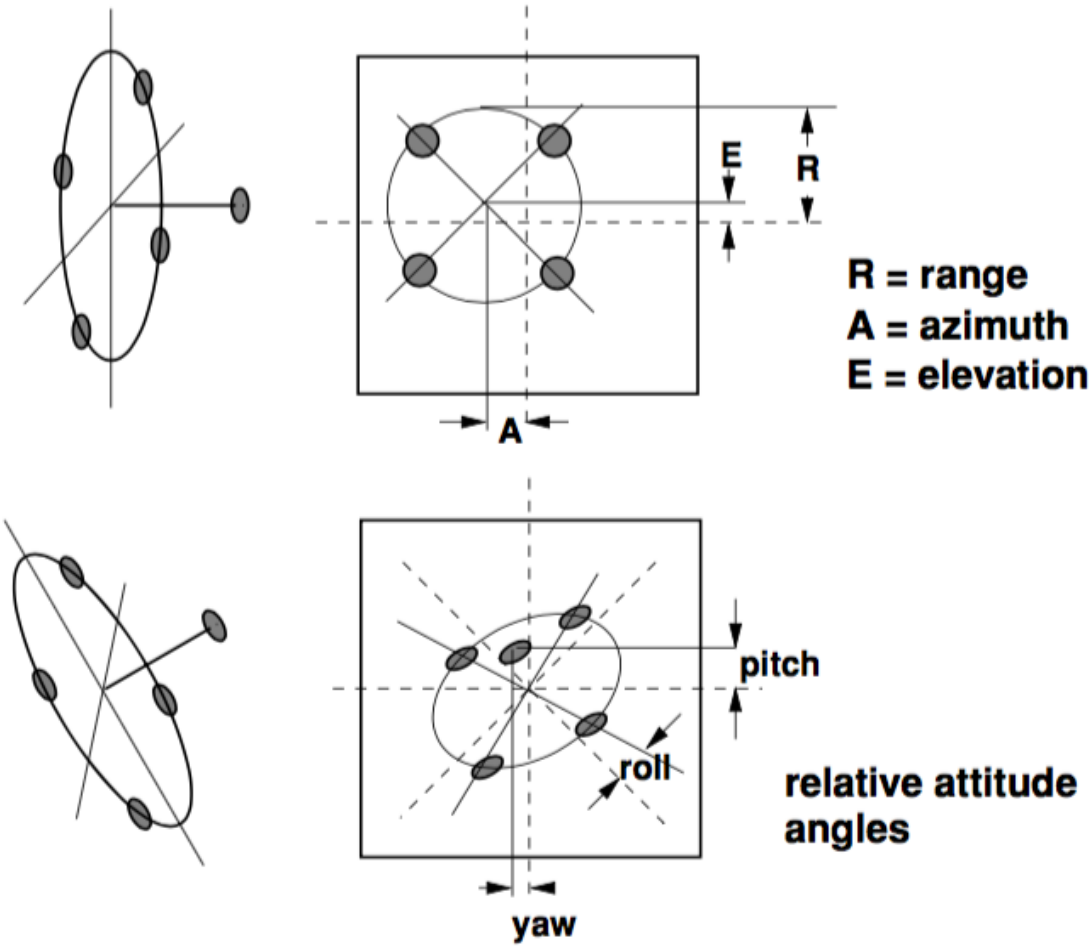


Figure 13 Target pattern for a camera rendezvous sensor[16]

2.2.2.3 Example

The experiment belongs to NASDA (the National Space Development Agency of Japan) and was performed by Masaaki Mokuno, Isao Kawano and Takashi Suzuki, Japan Aerospace Exploration Agency (JAXA), 2004 (Mokuno, Kawano, & Suzuki, 2004).

This is the experiment of unmanned autonomous space rendezvous docking (RVD) using the Engineering Test Satellite VII (ETS-VII), which weighs 2900 kilograms, and the rendezvous laser radar (RVR). The final approach phase profile of the experiment can be seen in Figure 14, the physical arrangement of the components of the sensing system used in this

[16] http://ebooks.cambridge.org/chapter.jsf?bid=CBO9780511543388&cid=CBO9780511543388A043&tabName=Chapter

experiment will be shown in Figure 15, and there is the photo and schematic diagram in Figure 16.

The equipment for this experiment includes:

- Two sets of RVR-head (RVR-H) on the chaser. Each RVR-H has laser transmitting and receiving functions. Components are: the laser diode to generate the outgoing laser beam(s), the half mirror to receive the reflected laser beam(s) and to divide the beam into two lines, one goes to the two dimensions CCD and is used here to detect the two-dimensional position; and the other goes to the Avalanche Photodiode APD and is used here to measure the optical power of the returned light.

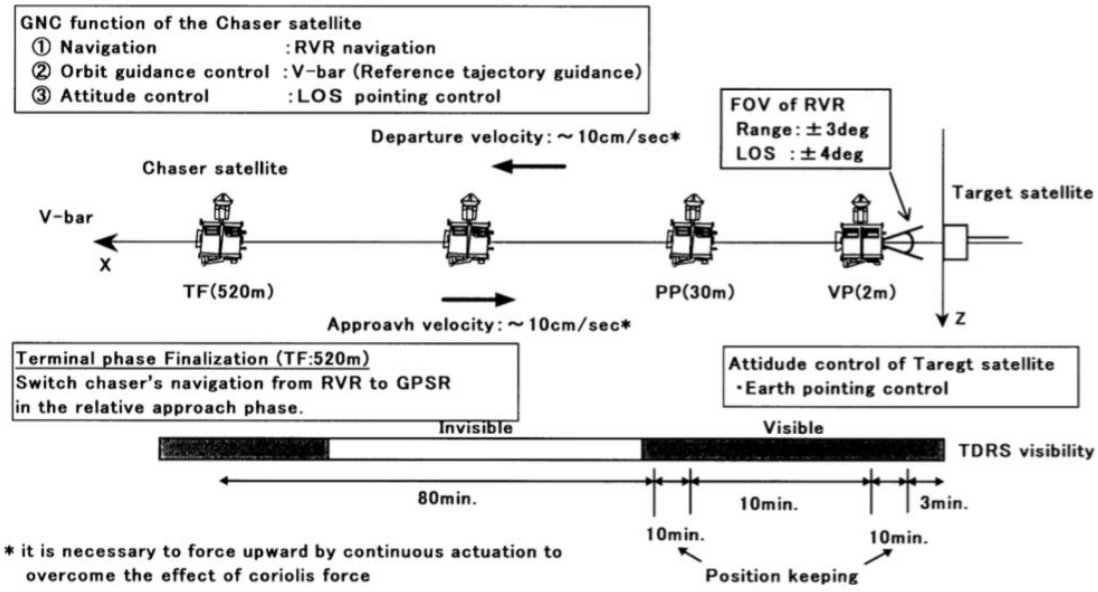


Figure 14 Overview of final approach phase[17]

- RVR-electronics (RVR-E) on the chaser. It has the functions of calculating, controlling and operating telemetry/command. This gathers all the information from the CCD and the APD to determine the relative range and the direction LOS angles as well as the relative attitude of the target.
- RVR-reflectors (RVR-R) on the target. There are two sorts of reflectors were used. RVR-R-1 is the one used for far-range measurement, consists twenty-four corner-cube-reflectors (CCRs) with the edge size of 5 cm, one set. RVR-R-2 is the one used for near-range measurement, each RVR-R-2 consists one CCR with the edge

[17] http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1310009

size of 2 cm, two sets. The reason of this design is that when the chaser gets really close to the target, and because the RVR-R-1 with twenty-four CCRs has probably a big size, its pattern might move out of the FOV of the camera on the chaser, therefore this method requires another target pattern which is smaller so that it can be captured inside the border of the image of the chaser's camera. Also with its smaller size, it can increase the accuracy of the calculation.

In Figure 15, the real pictures of those components can be seen, but probably excluding the RVR-E because this one is a kind of the software or the program, which is compiled inside the processor or the onboard computer once mentioned in section **1.3**.

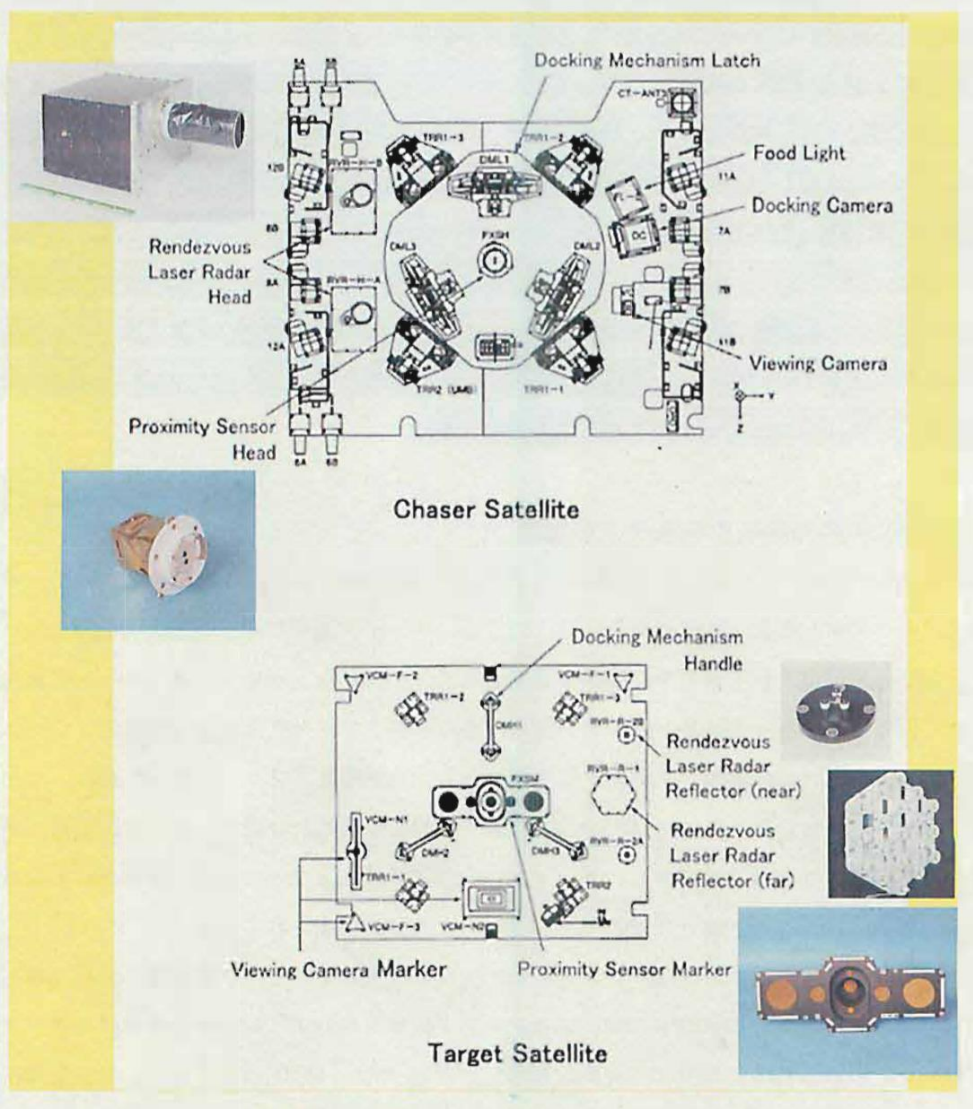


Figure 15 Docking planes of chaser and target satellites[18]

[18] Image credit from Masaaki Mokuno, doctoral thesis (Mokuno, 2011).

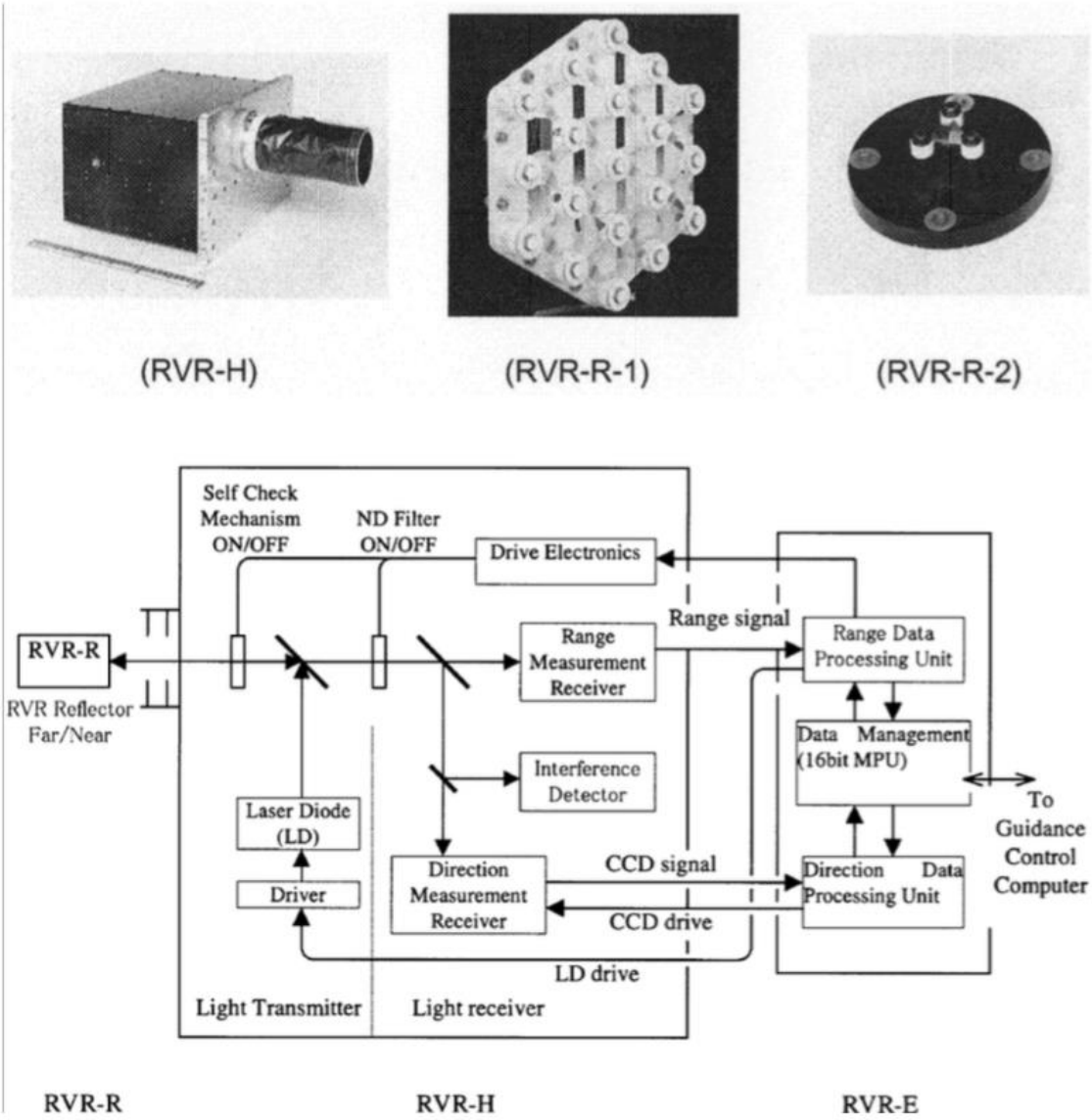


Figure 16 Photo and schematic diagram of the RVR[19]

2.2.2.4 Problem of the RVR System

Generally, this system was **complicated** in operation. It was made from many single components: **24 CCRs, the RVR with the CCD and APD sensors** to name a few. The cost of this sensing system was roughly around **tens to hundred of million Yen** (Masaaiki Mokuno, 2016). These were the reasons it was used to the **ETS-VII, which is the huge satellite** for testing. In the comparison with the cost for a microsatellite, roughly **six hundred million Yen** (Axelspace Corporation, 2015), the cost of this sensing system would be a big deal.

[19] http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1310009

2.3 Literature Review

In this section, some information, experiments and tests regarding the applications of small components with the time-of-flight principle will be introduced. These studies are not yet the individual sensing system for autonomous space rendezvous, but they shows the significant potential.

2.3.1 The DragonEye Product

The DragonEye Space Camera™ is the product from Advanced Scientific Concepts, Inc., (http://www.advancedscientificconcepts.com, 2016). In "A Survey of LIDAR Technology and its Use in Spacecraft Relative Navigation" (Holder, 2013), amongst the components were introduced, the DragonEye camera has the most similar working principle to TOF cameras. The difference is that it uses the APD type of sensors, the same type was used in the experiment at JAXA **(2.2.2.3)** (Mokuno et al., 2004), whilst TOF cameras use the low-cost CMOS sensors (Li, 2014) instead.

This product was designed for NASA's Commercial Orbital Transportation Services applications, and has been tested on some NASA's missions. It was *"the right sensor for Automated Rendezvous and Docking (AR&D) with the International Space Station and On-orbit Satellite Servicing"*[20]. This means that DragonEye was used as a part of the entire rendezvous sensing system in those tests. The result also showed that the range of this time-of-flight principle type of cameras, i.e. emits modulated lights, receives reflected ones and measure the phase shift to calculate the distance, including DragonEye cameras and TOF cameras, can be literally lengthened up to hundreds of metres (Holder, 2013).

There is also the information about the range accuracy of this camera on the website. Its error is about 10 centimetres.

The result showed that DragonEye cameras focus on the range measurement and does not provide information about the direction and the attitude measurements.

2.3.2 Experiment with TOF Cameras in Space Industry

There were some experiments conducted in the years 2012 and 2013 to evaluate the time-of-flight principle and TOF camera in space rendezvous applications. These researches were sponsored at the German Aerospace Centre (DLR).

[20] http://www.advancedscientificconcepts.com/products/older-products/dragoneye.html

Firstly, in April 2012, Leonardo Regoli, MSc. Karthik, Dr. Marco Schmidt, and Prof. Klaus Schilling from the Universität Würzburg, Germany, published their paper "Advanced Techniques for Spacecraft Motion Estimation Using PMD Sensors" (Regoli, Ravandoor, Schmidt, & Schilling, 2012). "PMD" is a "Photonic Mixer Device". In this experiment, the time-of-flight principle was used for the range calculation. For the attitude angles' calculations, the edge detector algorithm called "Canny algorithm" (Wikipedia, 2016a) was investigated, Figure 17.

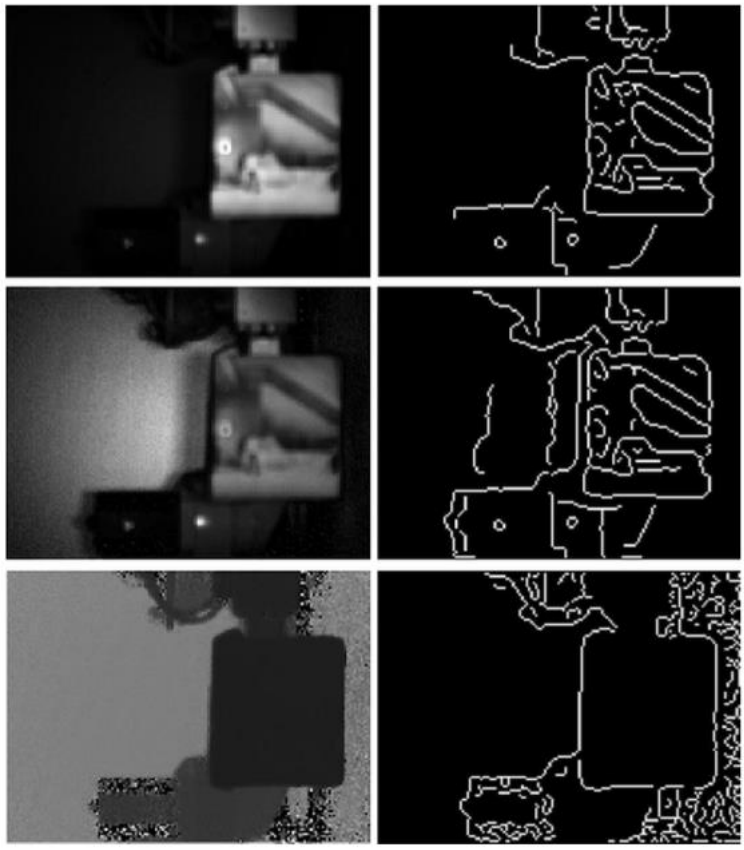


Figure 17 Canny edge detections using amplitude image (top), intensity image (middle) and depth image (bottom)[21]

This figure shows how the edges of the object were captured in the picture by the Canny algorithm. These edges were used to estimate the motion of the object.

The results of the tests, including the range measurement (the distance determination) and the rotational motion measurement (attitude angle determinations), were gathered to plot the graphs of their errors in percentage, as in Figure 18 and Figure 19.

[21] (Regoli et al., 2012)

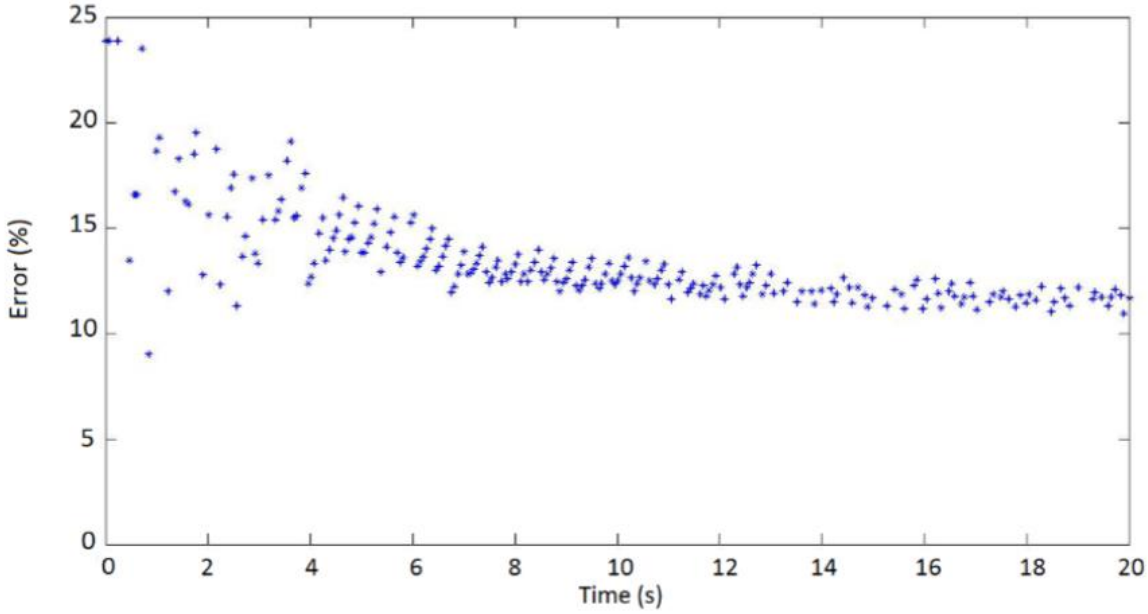


Figure 18 Percentage error for range measurement[22]

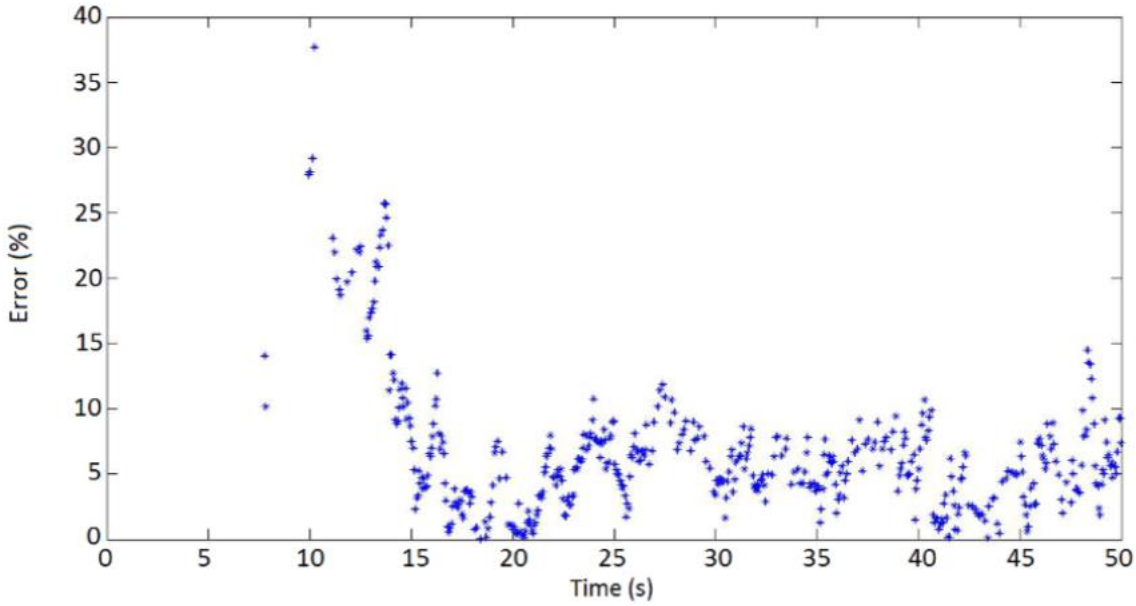


Figure 19 Percentage error for the tracked rotational motion measurement[23]

The point is**, there was no information about the LOS angles to be shown in their paper.** This means that in this experiment, the focus was range and attitude measurements. However, the direction of the target is also important in autonomous space rendezvous.

[22] (Regoli et al., 2012)

[23] (Regoli et al., 2012)

Moreover, the Canny edge detector algorithm was prototyped using MATLAB®. *"Thus the definition for model of computation is still open"*[24]. These mean the sensing system was not completed developed in this research.

Move to September 2013, there was the paper published by Tristan Tzschichholz and Prof. Klaus Schilling also from the University of Würzburg, Germany. This report describes an experiment of a TOF camera with a purpose to evaluate the ability of the extension in the range measurement, regarding to the camera's applications in space rendezvous (Tzschichholz & Schilling, 2013).

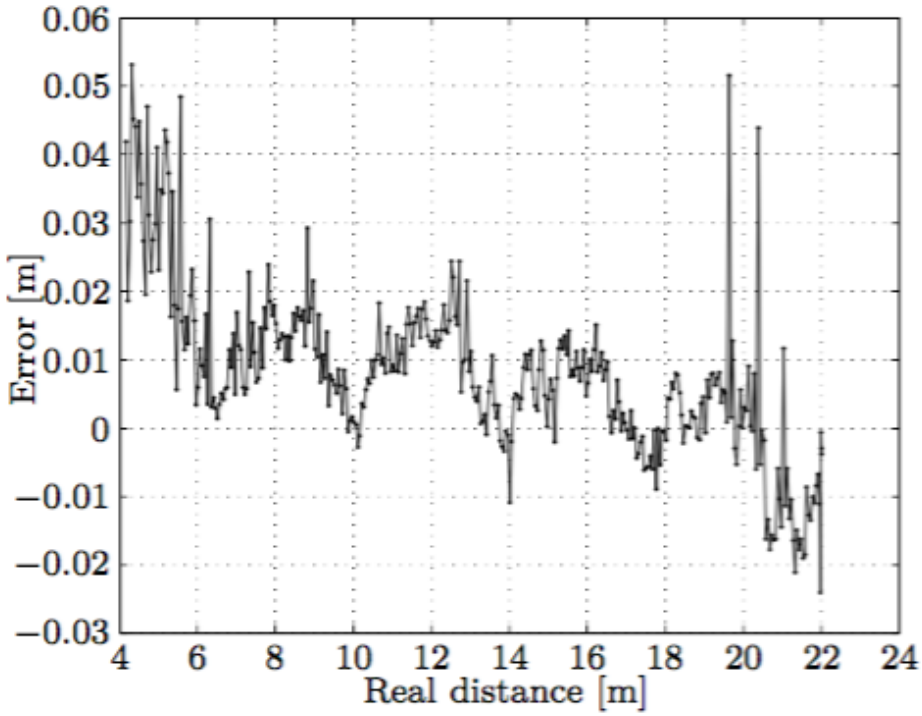


Figure 20 Position estimation error for the distance to target[25]

The result of the test in the term of accuracy is shown in Figure 20. The paper concluded that the measurement range (distance) of TOF cameras is possibly well modified and extended.

This means that this research **focused on the extending the range measurement**. There was **no information about angle measurement results, i.e. direction and attitude measurement** in the paper.

[24] (Regoli et al., 2012)

[25] (Tzschichholz & Schilling, 2013)

There was another paper published in the same year, 2013, about the experiment conducted by Toralf Boge, Heike Benninghoff and Tristan Tzschichholz using TOF camera at the German Aerospace Centre (DLR) (Boge, Benninghoff, & Tzschichholz, 2013). This experiment used the popular existing industrial application of TOF camera, which presents the typical three-dimensional (3D) image of an object. This equipment was used is the tracked model with the real-size 2.3 x 1.8 metres square aluminium part. Figure 21 shows the 3D image of the experiment in which the distances to the target (the satellite mockup in their laboratory) were simulated by different colours.

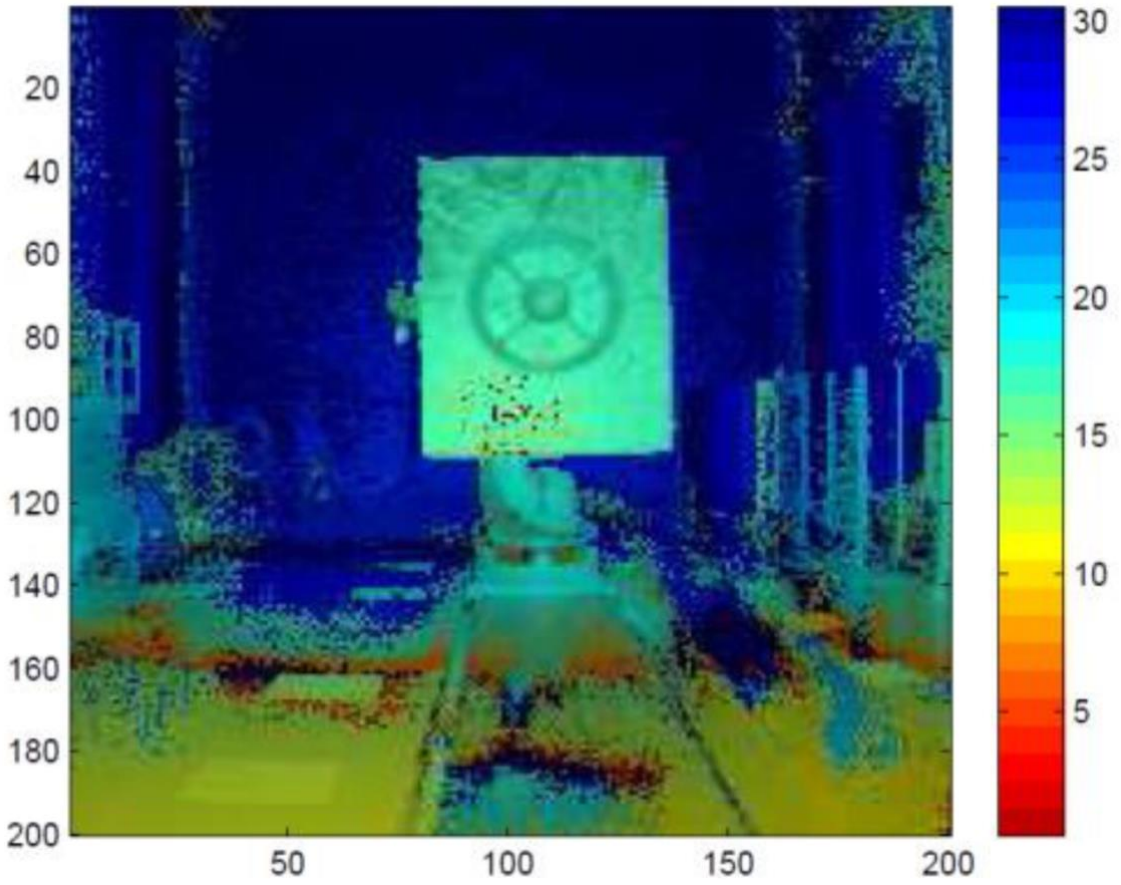


Figure 21 A typical 3D image showing a satellite mockup[26]

In this experiment, a TOF camera was **used together with a monocular camera sensor to detect only the distance**. The result was shown as in Figure 21.

In the same paper, the accuracy evaluations of this monocular carmera are shown in Figure 22 and Figure 23. It shows that the range accuracy in their experiment can be

[26] (Boge et al., 2013)

acceptable up to 20 cm, Figure 22, and the angle accuracy was about the scale of degrees, Figure 23.

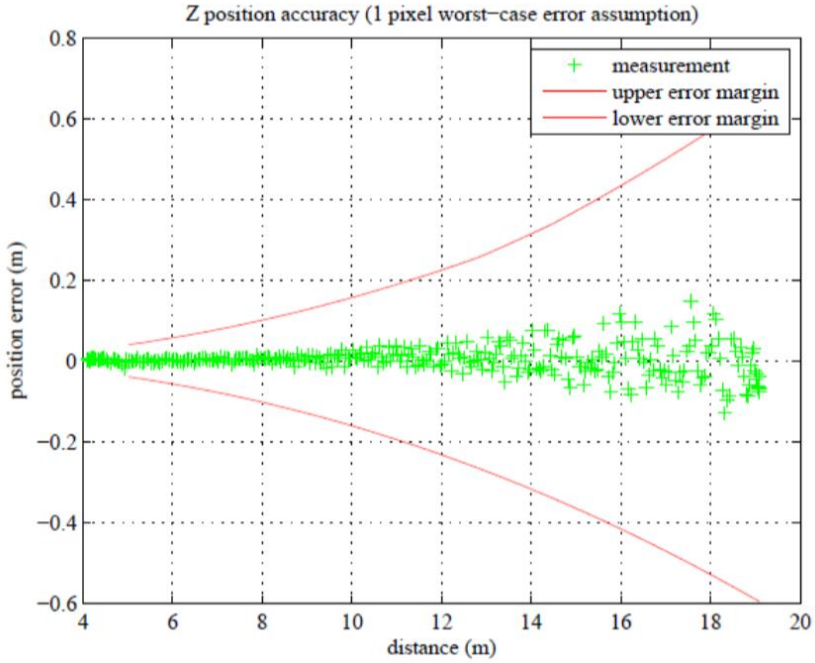


Figure 22 Monocular sensor's range accuracy[27]

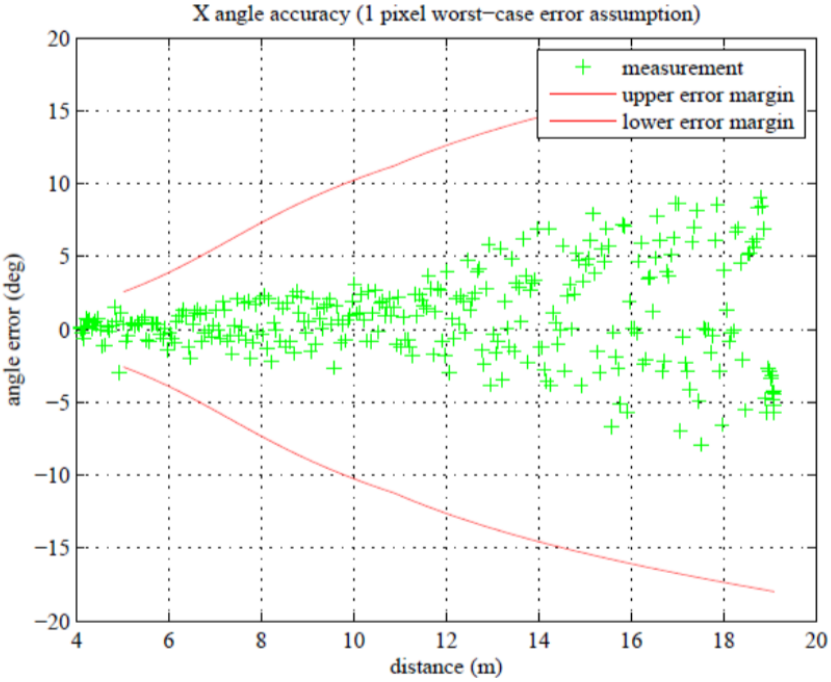


Figure 23 Monocular sensor's angle accuracy[28]

[27] (Boge et al., 2013)

[28] (Boge et al., 2013)

2.4 Problem Definition

There is one common point in the sensing systems that recently have been introduced in section **2.2** (the Kurs and the RVR): they were big, complex, and all designed for the huge satellites with a big number of physical components and high cost.

In the radio frequency radar system, there were ten antennas in total placed on both the chaser and the target, five for each, to do the task (Fehse, 2003). Besides, it also requires in both sides the radio transmitting and receiving units, which have functions of tracking antenna position, modulating (before transmitting) and demodulating (after receiving) the radio signals. It used up to four different frequencies throughout the entire rendezvous.

In the laser range finder type of sensors and the camera type of sensors, as can be seen from the experiment with the ETS-VII, there were also quite a lot of components (28 single components) being arranged on both sides to do the sensing task: on the chaser, there were two sets of RVR-H (30 x 30 x 20 centimetres cube) which is the combination of laser beam transceiver, mirror, CCD, and APD; on the target, there were twenty-four CCRs of the 5 cm edge size and two others of the 2 cm edge size in total (Mokuno et al., 2004). This sensing system approximately costs at around tens of million Yen.

Having to say, the more components to be used, the more difficulties it makes to the design, and the more complicated the process would be, because they need to deal more with each other. Then it comes other consequences: the weight, the size, and the costs of the two spacecraft have a high potential to be increased. That is why these designs were used on such huge spacecraft, such as Soyuz, Mir, and ETS vehicles.

When it comes to small satellites, such as those in the micro class of satellites, it is difficult to operate with these big and complex sensing systems. Moreover, the costs of tens of million Yen are the big deal for a budget of designing small satellites, which is estimated around a few hundreds of millions Yen (Axelspace Corporation, 2015). Therefore, a small and simple sensing system is needed to design to adapt the need of rendezvous for small satellites.

The DragonEye product only focuses on the range measurement and does not from the entire rendezvous sensing system.

In the experiment of the PMD sensor, the entire sensing system was not completed. The navigation software is still opened to develop. The LOS angles were not mentioned in its result.

In the experiments with TOF cameras, firstly, there was only the extension of the range measurement of the TOF camera tested, to confirm the ability of extending the distance

of measurement. Second, the tracked model of the space rendezvous test was also quite big in term of size as mentioned in section **2.3**. Also not only TOF camera was used. There was a monocular camera sensor used in the sensing system. This combination would make the system quite complicated in operation and also difficult to bring to such small satellites. Besides, nothing new was derived from the TOF camera. It was shown in their report that it only provided a colourized picture, which did not provide any parameters.

Then the question is: "**Is there still a need of making a small and light sensing system for autonomous space rendezvous, especially for small satellites?**" The answer is yes. And due to the fact that the budget, both in term of finance and time, for a small satellite is not as much as for a huge spacecraft, there is still a desire of having a simple, small, light, cheap and fast equipped sensing system, which is suitable for the circumstances.

It is important to say that **TOF type of cameras has been studied** a lot in the terms of space technology and industry. People have thought about **extending its range** (including modify the light's frequency and use the retro-reflectors), using its **3D image**, **cooperating with other devices** to form an entire rendezvous sensing system, but none of them have thought about using it, in the combination of image processing technique with a **particular designed target pattern**, to derive all the possible advantages. That will be proposed and described within this report.

3 Research Proposal

This chapter talks about the proposal of making the new sensing system for autonomous space rendezvous using TOF camera as its core component and device. Firstly, there will be the discussion about the goal of this research **(3.1)**, which points out the need of a new sensing system, based on the problem definition in section **2.4**. After that, the basic introduction to the time-of-flight (TOF) type of cameras will be shown, and its potential in the circumstances of autonomous space rendezvous will be discussed **(3.2)**. The last section of this chapter will propose the idea of building the new sensing system with TOF camera, and it will go through the whole progress of the designing phase **(3.3)**.

3.1 Research Goal

First of all, this is an overview of what have been talking about from the beginning of this paper. There are:

1. The need of autonomous rendezvous in space missions, including Earth-observation and Space-exploration. Space rendezvous are important in supplying to the ISS or landing equipment on comets and asteroids, those activities are crucial in the space industry;
2. The problem from existing sensing systems for autonomous space rendezvous is: almost all of them are complicated in equipment and operation, with many components, big size and heavy weight. Therefore, most of them have been used in huge spacecraft;
3. The key trend of miniaturising satellites is giving the new requirement to every element used to form the satellite. They should be made smaller and lighter. And therefore, they are able to reduce the launching cost as an additional consequence.

Based on all these results, the goal of this research is making a new sensing system, which is particularly used in autonomous space rendezvous. This sensing system should achieve some characteristics:

a. Easy to bring to not only huge spacecraft but also small satellites such as those are in the micro class of satellites. This requirement can be broken down as the new sensing system should have these physical requirements:

- Small size
- Low weight

b. Simple in terms of implementation and operation, which means the new sensing system should have:

- Few components to be used
- Simple calculation process

c. Low cost and based on existing techniques and devices.

Due to the goals of the new sensing system, based on the literature review in section **2.**3, TOF cameras were chosen to do this research. It is small and light. The compliment that a TOF camera is the commercial product with low-cost sensors makes it cheap and conveniently available.

This research focuses more on the approach to the cooperative type of targets, but it also can be applied to the scenarios of non-cooperative targets. This design aims to small satellite, however, it can be used in also big satellite.

This application of TOF cameras uses the different algorithm to the research has been reviewed in section **2.3.** The entire sensing system will be built with the combination of the unique designed target pattern.

Precisely, a commercial **TOF camera**, which is placed on the chaser, the three **corner-cube-reflectors (CCRs)**, which are attached to the target to form the communication interface between the two, and the software for calculating the navigation parameters, which is written in **C++ programming language**, will form this whole sensing system. The **connected-component labeling algorithm** (Wikipedia, 2016c) was chosen to use in the code processing the intensity image of TOF camera to achieve the calculating purpose. The low-cost CMOS sensors using in commercial TOF cameras, **the specific design of the target pattern**, and the algorithm used in this new sensing system make the research unique.

3.2 Time-of-Flight Camera

A time-of-flight (TOF) camera, as the one in Figure 24, uses a low-cost complementary metal-oxide-semiconductor (CMOS) pixel array together with an active, modulated light source to produce a depth image (Li, 2014). Each pixel of an image encodes the distance to the corresponding point of the scene (Hansard et al., 2012). Compact construction, easy to use, high accuracy and high frame-rate are the advantages making TOF camera an attractive solution for many practical applications including robot navigation, 3D

reconstruction and human-machine interaction (Hansard et al., 2012; Li, 2014). The cost of this product varies at around tens of thousand Yen.

Figure 24 A time-of-flight camera[29]

This type of camera has the average size of 5 x 5 x 20 centimetres cube, but they can be made differently in size and shape to a certain extent if there is a requirement.

3.2.1 Principle

The principle of TOF camera, visualized in Figure 25, is that it measures the phase delay (or phase shift) between the reflected infrared (IR) light and its corresponding emitted one to calculate the distance from each sensor in the pixel array to the target object (Hansard et al., 2012). The frequency of the modulated light is known.

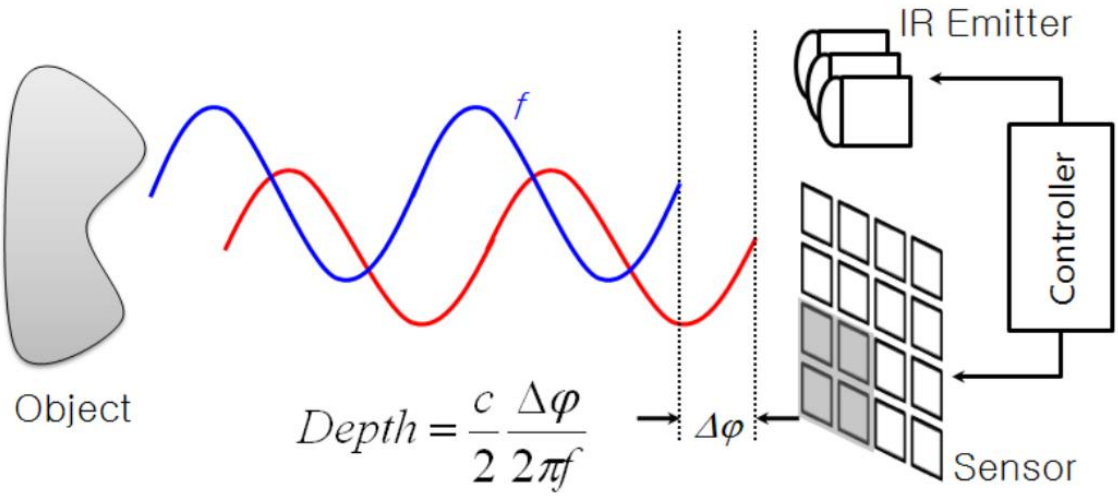


Figure 25 Principle of TOF depth camera[30]

The equation is used for the distance calculation is:

[29] Source from the internet

[30] (Hansard et al., 2012) https://hal.inria.fr/hal-00725654/PDF/TOF.pdf

$$d = \frac{c}{2}\frac{\Delta\varphi}{2\pi f} \quad (1)$$

Where:

d is the physical distance from the target object to the image plane of the camera.

c is the speed of light which is roughly three hundred million metres per second (300000000 m/s).

f is the modulated frequency of the IR light.

$\Delta\varphi$ is the phase delay.

The phase shift $\Delta\varphi$ is calculated using the four-phase control signals with 90 degrees delays from each other. The quantities Q1, Q2, Q3, and Q4 represent the electric charge values collected by each signal C1, C2, C3, and C4 in one period (360 degrees). Then the phase shift $\Delta\varphi$ will be:

$$\Delta\varphi = \arctan\left(\frac{Q_3 - Q_4}{Q_1 - Q_2}\right) \quad (2)$$

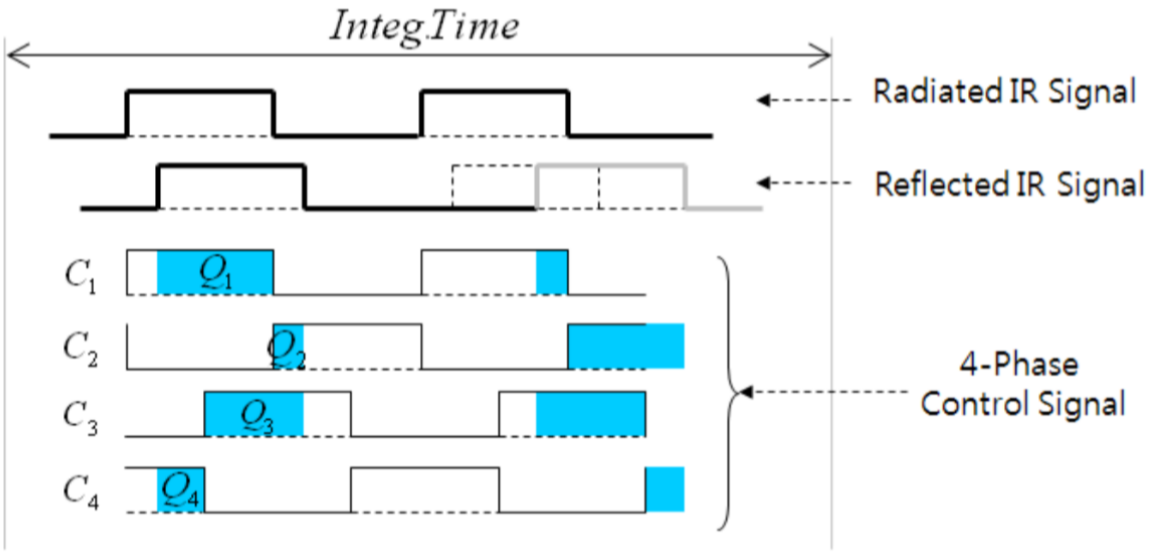


Figure 26 Phase shift determination[31]

3.2.2 Potential

As introduced in section **1.4.1**, the information is sensed separately in every sensor on the image plane of the camera. This means that a TOF camera takes images of information.

[31] (Hansard et al., 2012) https://hal.inria.fr/hal-00725654/PDF/TOF.pdf

The information that is contained in each pixel, such as depth information or intensity information, can be extracted and used depending on the purpose of the user.

Therefore, if a TOF camera is put on the chaser satellite, the physical distance between the chaser and the target is provided through the depth image of the camera.

In the context of rendezvous in the space environment, there is a very high possibility that no any other objects could be around in the picture frame. And by determining the position of the object (which is the target) using the intensity image, with reference to the two-axis coordinate system of the camera on the chaser side, it is possible to know the direction, which is the two LOS angles, of the target.

Besides, the information about the brightness of each pixel in the intensity picture can also be useful to determine the relative attitude of the target. In space, it can be assumed that the CCRs (corner cube reflector) are the objects that most strongly reflect the light and make the brightest points on the intensity image of TOF camera. By doing image processing with the connected-component-labeling algorithm (Wikipedia, 2016c) to find the three brightest points, then by applying some geometry algorithm for those points, the attitude can be theoretically detected.

With all the analyses have been made, it can be seen here a potential for TOF cameras to be brought to the concept of autonomous space rendezvous. With the support from some CCRs and the specific software program, the entire sensing system can be created. Based on the physical characteristics, the advantage is that it would be easy to place the camera in not only big spacecraft, but also small size satellites such as those in the micro class of satellites. This research is going to prove the feasibility of the idea.

3.3 System Design

This section will be showing how the new sensing system is going to be built using the system design methodology learnt from the International Council on Systems Engineering (INCOSE) SE Handbook[32]. It will go throughout the progress, starts with the requirements for the sensing system for autonomous space rendezvous in considering with the research goal and ends up to the physical architecture of the new sensing system for autonomous space rendezvous using TOF camera. There are some important steps that will be described in details. There is, firstly, the requirement analysis **(3.3.1)**, which includes the analysis of the

[32] http://www.incose.org

system boundary **(3.3.1.1),** the use case diagram of the system **(3.3.1.2)** and the set of functional requirement of the system **(3.3.1.3)**; then after the functional requirements of the system are created, there goes the architecture design **(3.3.2)**, which includes the functional design for the system **(3.3.2.1)**, the physical design for the system **(3.3.2.2)** and the architecture diagram of the system **(3.3.2.3)**.

3.3.1 Requirement Analysis

Following the analysis of the role of the relative satellite navigation in the general space rendezvous from section **1.2**, considering the analysis of the measurement parameters of an autonomous space rendezvous mission from section **1.4**, the main states of the rendezvous sensing system in its process to approach its purpose among the progress of a space rendezvous, which in short is to calculate and give parameters, can be described on the life cycle in Figure 27 below:

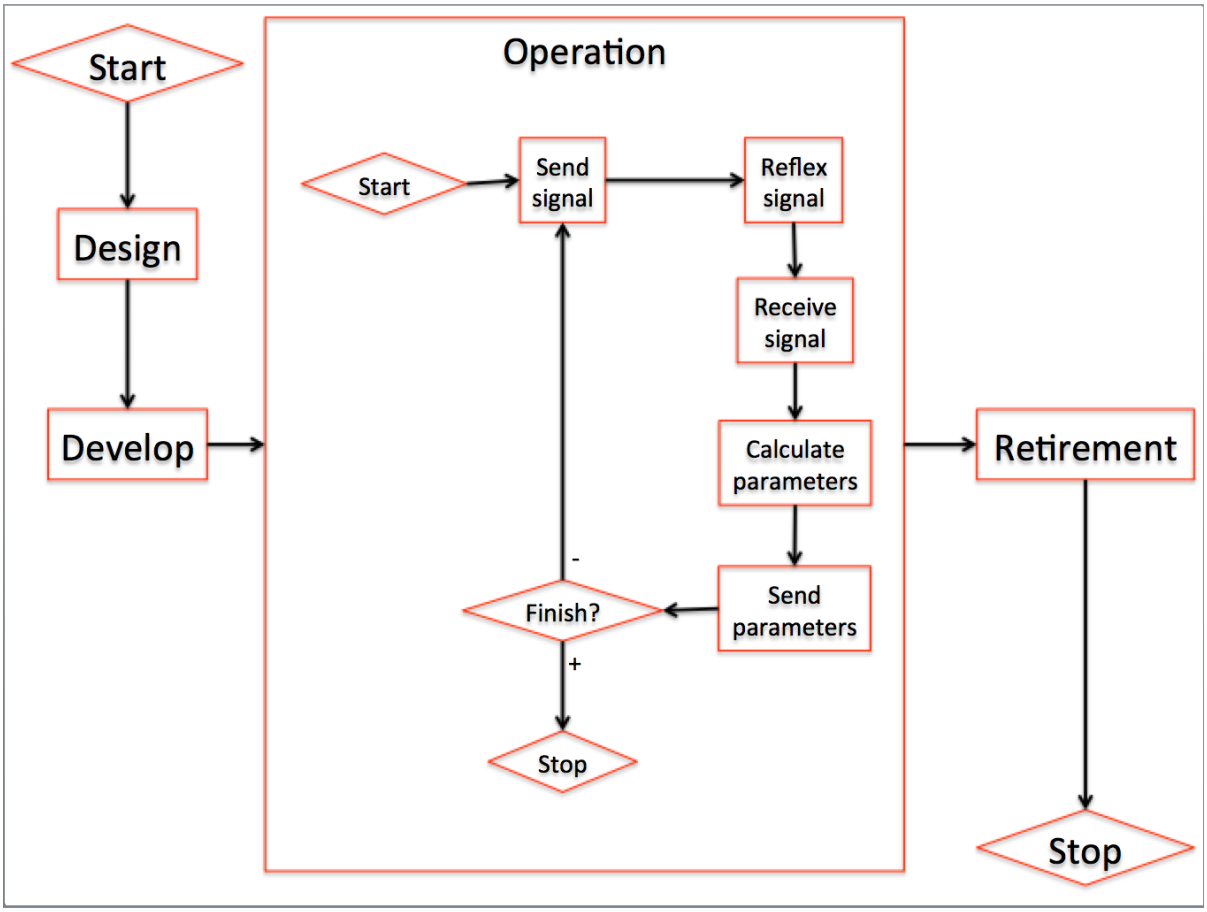


Figure 27 Rendezvous sensing system life cycle

In this life cycle, all the main functions of the sensing system, which is used in autonomous space rendezvous mission, are included in the operation phase. There is also the

need that should not be forgotten: according to the problem definition from section **2.4,** putting it on the circumstance of satellite trend, in the certain extent of applications of the system for the small size satellites especially those in the micro class of satellites, the requirements for the size and the weight of the sensing system are considered as the constraints given by the structure of a satellite. These factors are not mentioned in the life cycle of the system but they appear to be the initial factors in term of the allocation of the system from the design phase of the sensing system, which this research will be proposing.

One more note is that the retirement phase will be decided depends on the results from the operation phase. The sensing system can be reusable, can be improved to be better or will be destroyed somehow. Therefore this retirement phase will not be in among the objectives of this research and will not be discussed further.

3.3.1.1 System Boundary

System boundary is the result of analysing the context diagram in which the sensing system happens to be in the centre. It shows how the sensing system will interact with other systems or subsystems or stakeholders on the satellite, in order to achieve the objective, which is mainly giving the parameters needed for autonomous space rendezvous to the processor in charge of attitude controlling, which in particularly for the micro class of satellites is the onboard computer.

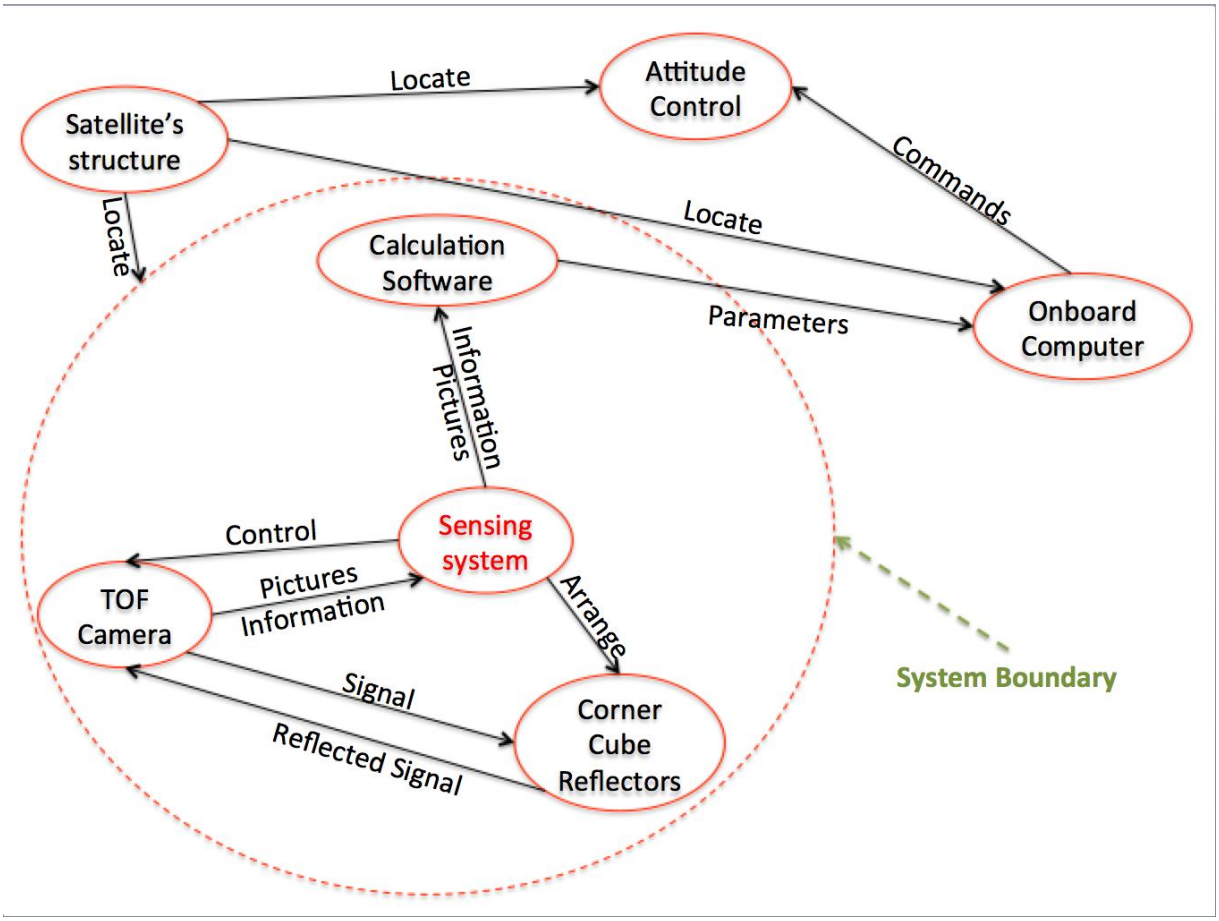


Figure 28 Context diagram and system boundary

As it can be seen in Figure 28 about the context of autonomous space rendezvous, this sensing system uses TOF camera with its existing technique and functions as the main component. The system includes the arrangement of some CCRs and the implementation of the calculation software inside the border or the system boundary. It is also the limitation of this research.

3.3.1.2 Use Case Diagram

From the context diagram, the concept of the sensing system for autonomous space rendezvous purpose can be understandable. And from the boundary of the system, it can be seen clearly the picture of what are needed to be done through the process of relative sensing. More precise about the functions and the design that going to be made in this research will be described in the use case diagram in Figure 29.

In the use case diagram, there are the functions that the sensing system should possess in order to interact with all the stakeholders, which have been brought down from the context diagram above, those have connection(s) directly to the system that is being built, or indirectly

to the stakeholder(s) that is/are inside the system boundary has been already cited inside the context diagram. As can be seen here, the core element of the use case diagram is the sensing system, those stakeholders will be involved in the use case diagram with the corresponding connections are: TOF camera, CCRs, calculation software (directly connect to the sensing system) and satellite structure, onboard computer (indirectly connect to the sensing system through one of the three direct stakeholders).

To avoid the ambiguity and misunderstanding from the lacking of information that can be drawn by the short description of the diagram, the full description should be made. With this combination of the use case diagram and the use case description, the total image of the process that happens inside the sensing system can be easily imagined.

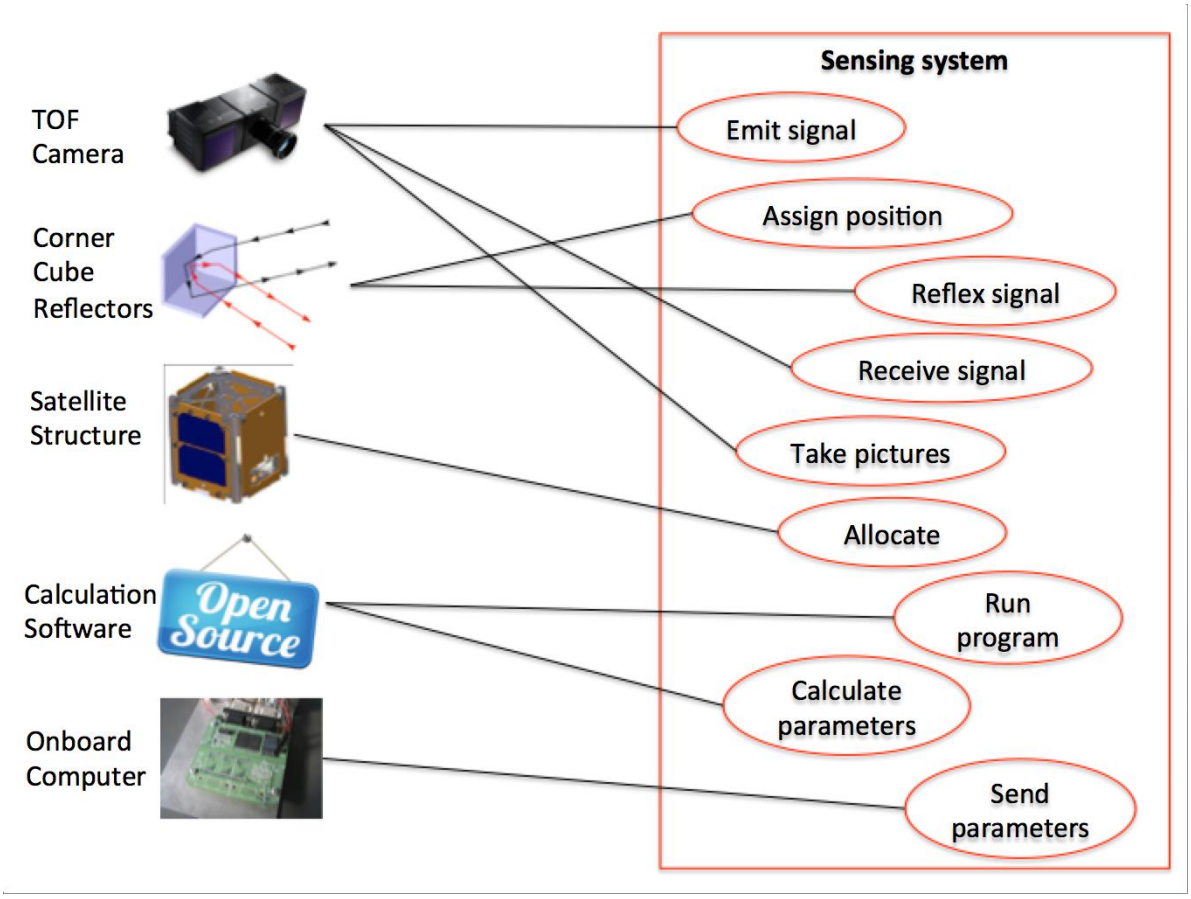


Figure 29 Use case diagram

And for more details of the use case diagram, this is the use case description:

Table 3 Use case description

Use case description	
Use case < … >	Description
1. Emit signal	1. The system controls the camera to set up desired parameters for the signal, which will be transmitted and will be used as the carrier of the information. 2. The system controls the camera to transmit modulated infrared light beams (signals) with known parameters as it was setting from step 1.
2. Assign position	1. Design the number and the shape of the CCRs will be used for the sensing system and their alignments on the target side. 2. The system tells the chaser about the information of these reflectors' positions.
3. Reflex signal	1. The CCRs reflex the signals, which were emitted from use case number 1 automatically to the same, reversal direction with the one the signals came from.
4. Receive signal	1. The sensors inside the camera automatically receive the reflected signals from use case number 3. 2. The sensors inside the camera compare the reflected signals with the one the camera emitted before to get the phase shift value (or phase difference) in each pixel of the camera's image. 3. The system receives this phase shift information from the camera.
5. Take pictures	1. The system controls the camera to take pictures in its field of view. These pictures have information of the brightness of each pixel. 2. The system receives these pictures from the camera with their special information.
6. Allocate	1. The system assures that the size of all its components on the target side is less than 25 cm x 10 cm x 10 cm to

	be able to place neatly inside the structure of the chaser satellite. 2. The system assures that it has the size of less than 15 cm x 15 cm x 1 cm in total for the CCRs to be able to place nicely on one surface of the structure of the target satellite.
7. Run program	1. The system starts the program to manage the procedure of use case number 1.1, 4.3 and 5.2.
8. Calculate parameters	For far range scenario (distance is more than 20 metres): 1. Using the phase shift received from the camera, the system gives it to the software as the input. 2. Using the pictures from the camera, the system gives them to the software as the inputs. 3. The software calculates the distance and direction of the target and places these parameters as its outputs. 4. The software compares the distance calculated above with the threshold of 20 meters. If equal or smaller, goes to step 5. For near range (distance is less than from 20 metres): 5. The software calculates the attitude of the target in advance and places these parameters as its outputs.
9. Send parameters	1. The system takes the outputs of the calculation software and sends them to the onboard computer of the satellite.

3.3.1.3 Set of Functional Requirements

Each description from Table 3, the use case description above, will be referred to one function, which the sensing system will possess, and all together they make the full set of functional requirements for the system.

Use case description 1.1 “The system controls the camera to set up desired parameters for the signal, which will be transmitted and will be used as the carrier of the information”

→ Function 1: Set up signal function.

Use case description 1.2 “The system controls the camera to transmit modulated infrared light beams (signals) with known parameters as it was setting from step 1”

→ Function 2: Transmit signal function.

Use case description 2.1 “Design the number and the shape of the CCRs will be used for the sensing system and their alignments on the target side”

→ Function 3: Design interface function.

Use case description 2.2 “The system tells the chaser about information of these reflectors’ positions”

→ Function 4: Update data function.

Use case description 3.1 “The CCRs reflex the signals, which were emitted from use case number 1 automatically to the same, reversal direction with the one the signals came from”

→ Function 5: Reflex signal function.

Use case description 4.1 “The camera automatically receives the reflected signal from use case number 3”

→ Function 6: Receive signal function.

Use case description 4.2 “The sensors inside the camera compare the reflected signals with the one the camera emitted before to get the phase shift value (or phase difference) in each pixel of the camera’s image”

→ Function 7: Calculate phase shift value function.

Use case description 4.3 “The system receives this phase shift information from the camera”

→ Function 8: Get phase shift information function.

Use case description 5.1 “The system controls the camera to take pictures in its field of view. These pictures have information of the brightness of each pixel”

→ Function 9: Take pictures function.

Use case description 5.2 “The system receives these pictures from the camera with their special information”

→ Function 10: Get pictures function.

Use case description 6.1 “The system has the size of less than 25 cm x 10 cm x 10 cm for all its components on the target side to be able to place neatly inside the chaser satellite structure”

→ Function 11: Develop “less than 25x10x10cm^3” chaser side function.

Use case description 6.2 “The system assures that it has the size of less than 15 cm x 15 cm x 1 cm in total for the CCRs to be able to place nicely on one surface of the structure of the target satellite”

→ Function 12: Develop “less than 15x15x1cm³” target side function.

Use case description 7.1 “The system starts the program to manage the procedure of use case number 1.1, 4.3 and 5.2”

→ Function 13: Implement calculation software function.

Use case description 8.1 “From the phase shift received from the camera, the system gives it to the software as the input” and use case description 8.2 “From the phase shift and the pictures from the camera, the system gives them to the software as the inputs”

→ Function 14: Get inputs function.

Use case description 8.3 “The software calculates the distance and direction of the target and gives these parameters as its outputs”

→ Function 15: Calculate for far range function.

Use case description 8.4 “The software compares the distance calculated above with the threshold of 20 meters. If equal or smaller, goes to step 5”

→ Function 16: Switch function.

Use case description 8.5 “The software calculates the attitude of the target in advance and gives these parameters as its outputs”

→ Function 17: Calculate for near range function.

Use case description 9.1 “The system takes the outputs of the calculation software and sends them to the onboard computer of the satellite”

→ Function 18: Send data function.

Generally, there are eighteen main functions in the set of functional requirements for the new sensing system as shown below in Figure 30:

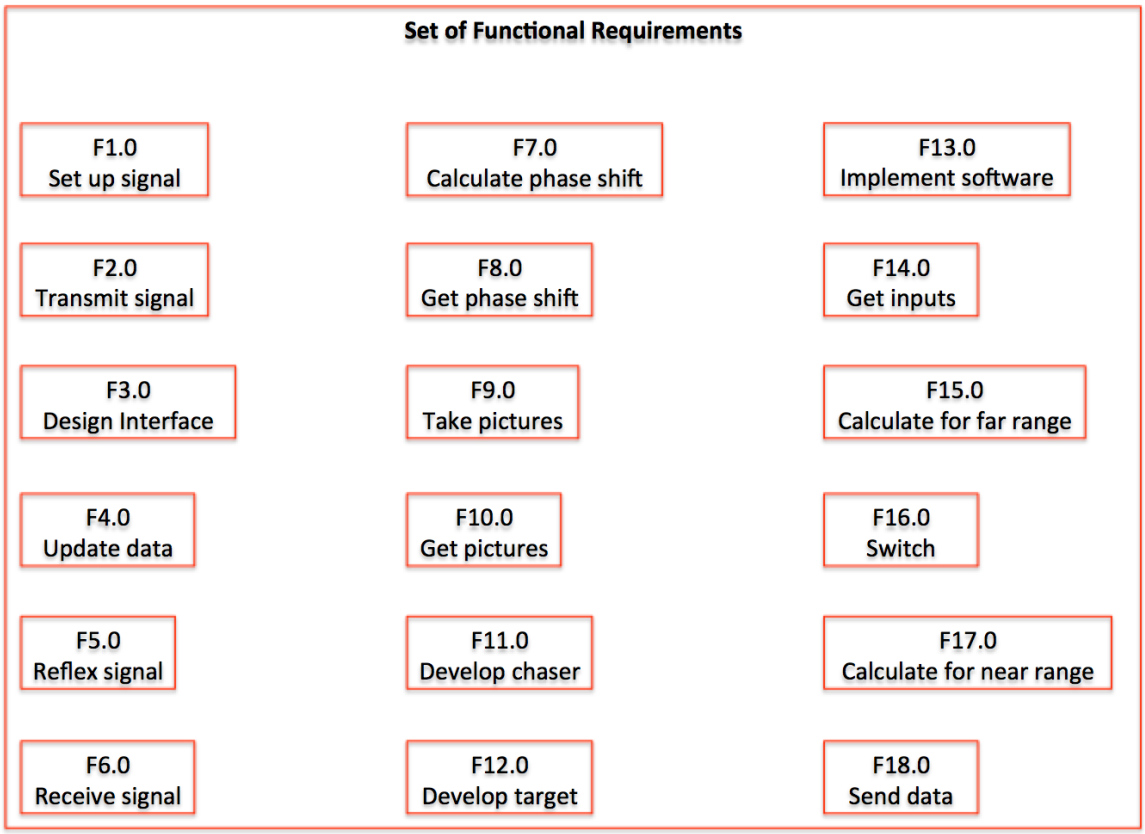


Figure 30 Set of functional requirements

3.3.2 Architectural Design

Architectural design is the designing phase for the real physical system that can be able to adapt all the functional requirements as clarified above. It means that from all the functions that the system need to have, the concept of the system can be made, that includes what physical components can be used, how they are connected with each other and what are their interactions with each other.

3.3.2.1 Functional Design

Functional design is the task of making the Functional Flow Block Diagram (FFBD), which shows how all the functions work from the previous ones to the next ones inside the system, how they are connecting to each other to make the entire system and to help the system achieve its goal.

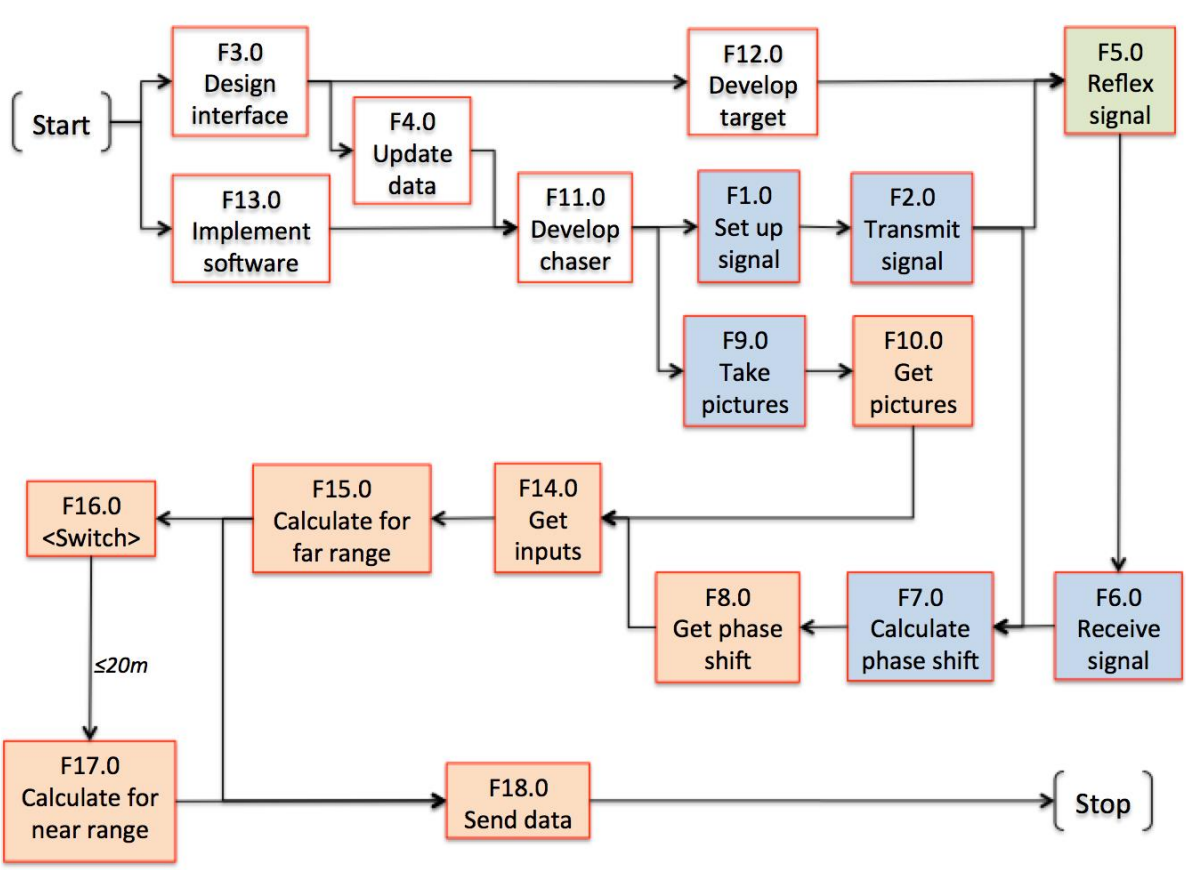


Figure 31 Functional Flow Block Diagram (FFBD)

Look at Figure 31, the FFBD, the function in the box with the green background is the function of the CCRs. This function will automatically work as it is the natural characteristic of the CCR itself, it does not need any effects to get to work; the functions in the boxes with the blue background are the functions of the TOF camera that also are initial and do not need to create or make any changes; the functions in the boxes with the orange background are the functions that need to create by writing the program or the software, which is one part of this research, it means that in this case they will be made inside the calculation software, which is the result of function **F13.0 "Implement software"** that will be discussed more details later.

The function F4.0 "Update data" is the activity that happens mainly in the onboard computer, by that the chaser gets known about the position of each CCR that attached on the target – in assuming that the chaser should achieve to the attitude at which the three-axis relative angles that indicate the relative attitude of the target with reference to the chaser's three-axes coordinate system are all zero degree.

The function **F3.0 "Design Interface"** will create the communication interface between the target and the chaser, through which the chaser will know about the position and

the attitude of the target depends on the requirements of the state where the process of an autonomous space rendezvous is. This function will be also discussed more details later.

For the continuing of the FFBD, Figure 32 and Figure 33 will be showing the lower level functions, which are broken down from some of those functions in the set of functional requirements. The reason of this activity is that sometimes there are several functions need to be made to create a bigger function, and the more details it can be clarified, the better the system will be created.

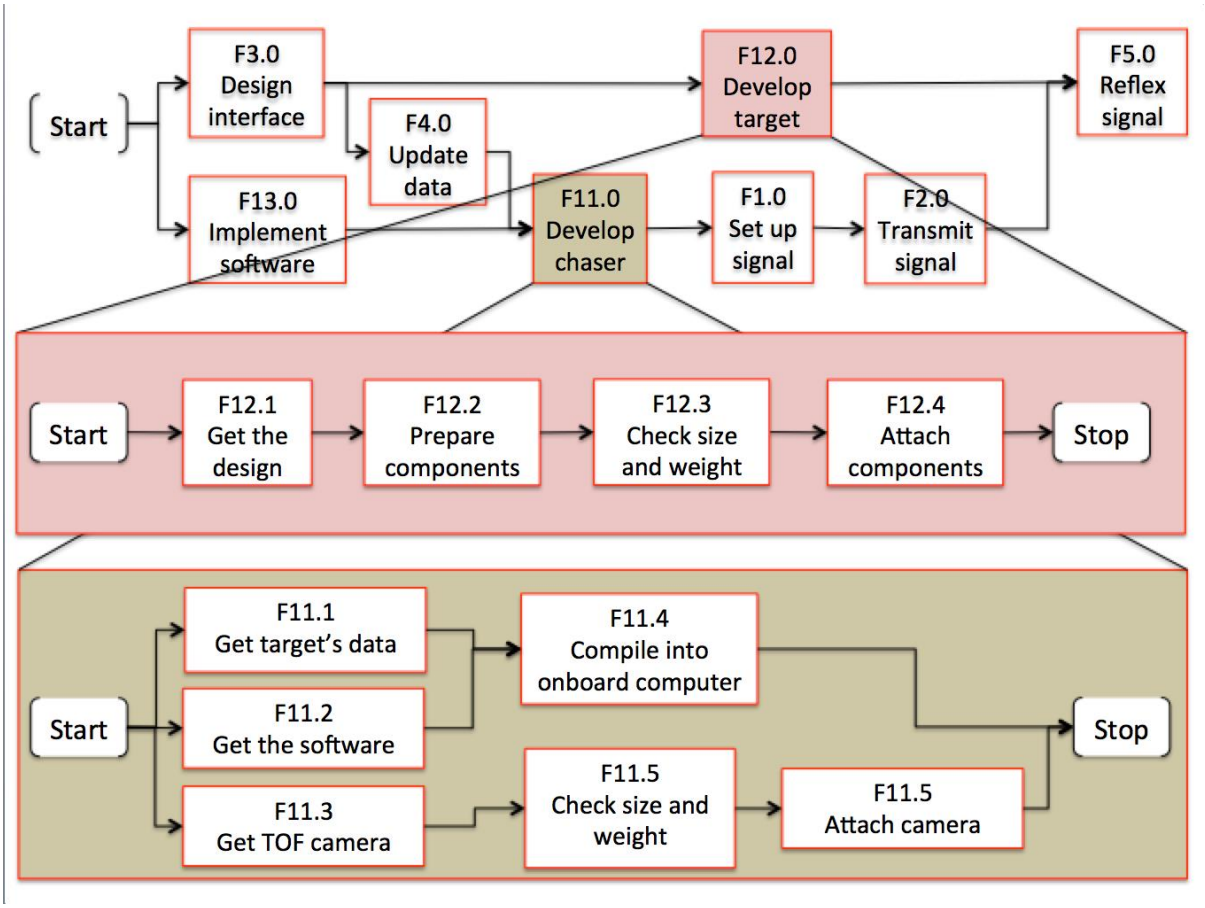


Figure 32 Function break down of F11 and F12

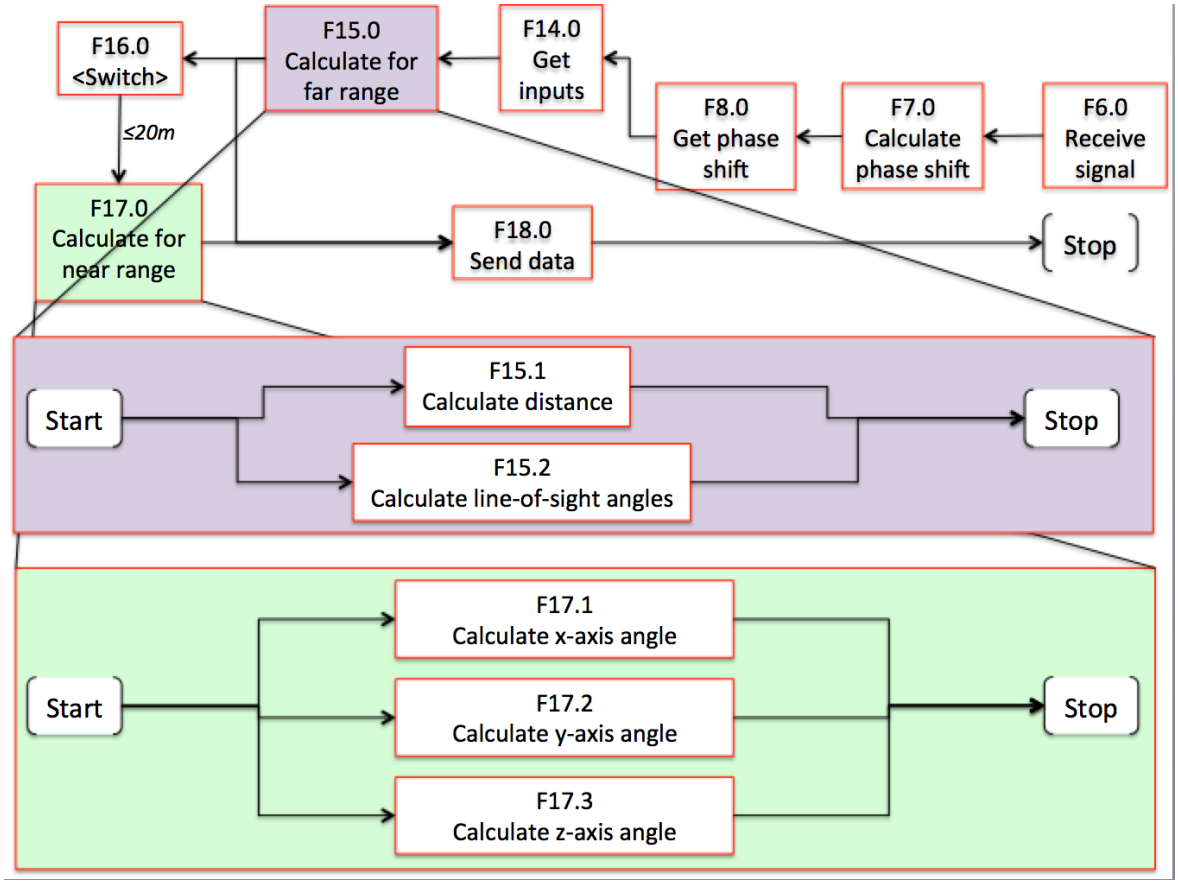


Figure 33 Function break down of F15 and F12

3.3.2.2 Physical Design

Basically, the physical design of the system is the step that locates all the lowest level functions gotten from the functional design, based on the connections amongst them, to the list of physical components or subsystems or organisations that the designer chose to create the entire system.

But before doing so, there are “Design interface” and “Implement software” functions that still need to be designed as already mentioned from the previous section, the **F3.0** and **F13.0**, and they should be designed in the physical design part, so then some functions can be allocated to them.

The real design

a. Interface (alignment of corner cube reflectors on the target side)

In this part, the core factor and component are the corner-cube-reflector (CCR). CCR is the reflector is made of an array of corner-cube reflectors. Each one looks like the one in Figure 34, the size of one piece of CCR varies depending on the

requirement. CCRs are considered one of the best types of reflectors at this moment in term of reflexing light-form signals.

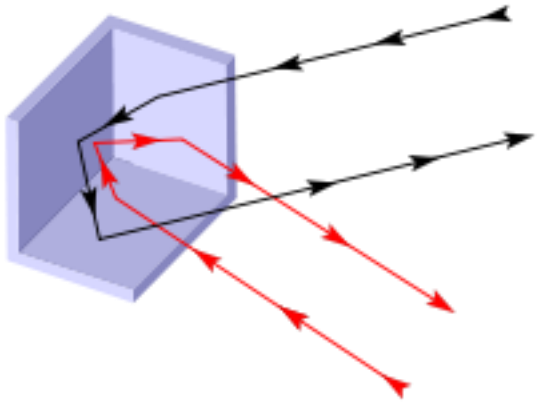


Figure 34 Corner cube reflector

It also has some advanced characteristics that are really useful in the application of autonomous space rendezvous such as (Wikipedia, Retroreflector, 2016):

- It reflexes almost one hundred percent of the power of the incoming light. It means that it prevents the reflected light from losing power due to the scattering phenomenon on the reflexing surface.

  This characteristic helps it to lengthen the effective distance of the light and also it makes the CCR's image to be assumed as the brightest point of the picture, which is important to detect the position of the CCR's image amongst other's on the picture was taken at the focal plane of the camera.
- It reflexes the light to the same but reversal direction of the incoming one as can bee seen by the visualised arrows in Figure 34.

  This helps the determination of the direction or the line-of-sight angles of the reflected light to be easier. And the direction of the reflected light, in this case, is also the direction of the target satellite.

Imagine in space environment, where the density of objects flying is not so high (ESA, 2013), in other words there are not many things can be moving around and especially nearby such a satellite (otherwise there will be a space collision), the only object would reflex the light from the camera and therefore would be taken in the picture in such a circumstance is the target satellite and the strongest reflected lights

that would appear in the picture as the brightest points would come from the CCRs due to its reflexing characteristic, thus on the picture taken at the focal plane of the camera, the light point or points can assumingly be understood as they came from the target and the brightest ones are the ones of the CCRs. Based on these assumptions, the calculation algorithm is made.

The construction of this design uses only three CCRs with the edge size of 5 cm. In Figure 35, the design for the alignment of the three CCRs on one surface of the target is shown, assuming that the satellite has a size of 50x50x50 cm^3.

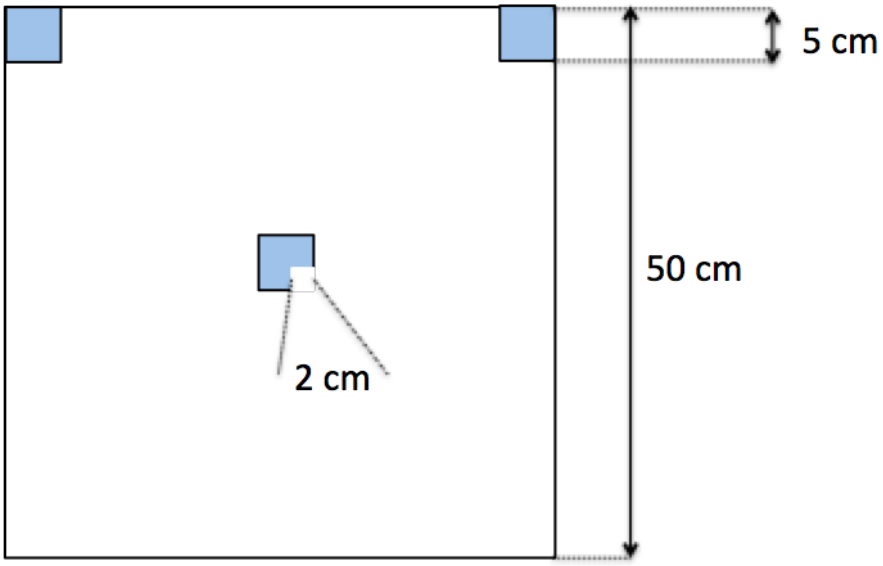


Figure 35 Target interface design

The unique arrangement of three CCRs and the special shape of the central reflector play the important role in the calculation algorithm. The usage of this design will be explained by three range cases: far range scenario (Figure 36), near range scenario (Figure 37), and when the target pattern moves out of the picture frame of the camera (Figure 38).

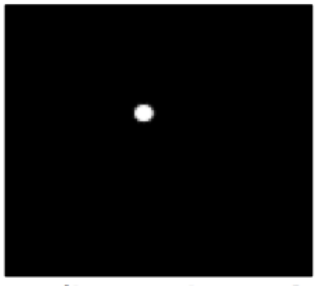


Figure 36 Image taken at far range scenario

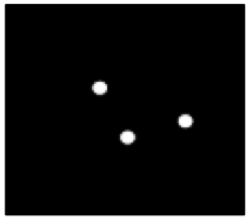


Figure 37 Image taken at near range scenario

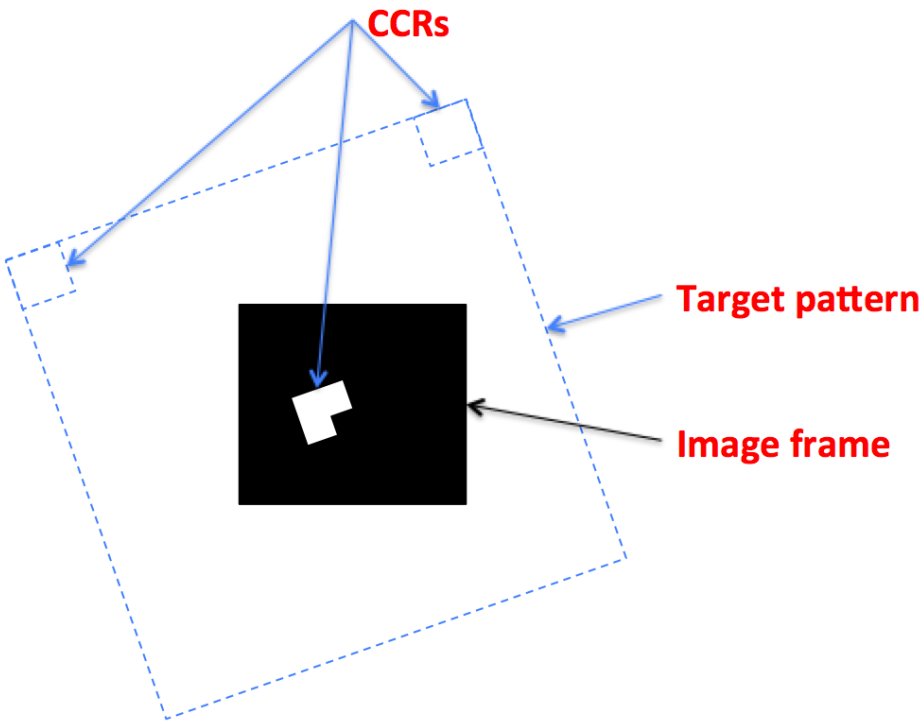


Figure 38 Image taken when the target pattern moves out of the picture frame

In the last case, the design is especially counted for the application in the autonomous docking rendezvous scenario.

Far range scenario: As an object moves with reference to the camera, it can be understood that from a certain distance, the picture which is taken by the TOF camera on its focal plane might not be clear enough to distinguish the three points corresponding to the three CCRs on the target, they might even become just one point. In this scenario, as long as there is still some reflected light comes back to the camera the information, or more precisely, the parameters of the distance and the LOS angles can still be detected using TOF camera and its own functions, and these values are enough to adapt the requirements from the autonomous space rendezvous for this long range scenario.

The algorithm of the calculations for these parameters will be demonstrated and explained afterwards with Figure 39 is how a picture looks like on the focal plane of the TOF camera, and Figure 40 is the geometry method of calculating the LOS angles.

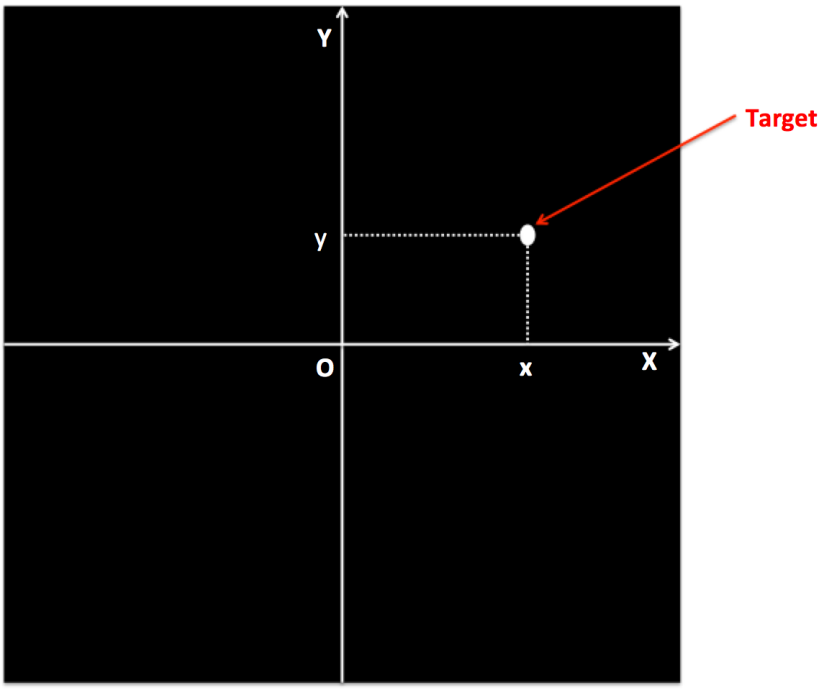


Figure 39 Example of TOF camera picture for far range

So the set of parameters needed for rendezvous in this scenario should be the distance (d) and the direction represented by a pair of LOS angles (α, β).

The distance can be calculated by using the mechanism of TOF camera as introduced in section **1.4**, using equation (1) that once showed above (Hansard et al., 2012).

Assuming there is no another object around the target satellite in the space environment, the picture of TOF camera would look like it is shown in Figure 39. Can be seen here, the light point in the picture is representing the target. Its position in the two-dimensional coordinate (OX, OY) of the picture can be detected as **(x, y)** by the TOF camera itself. From this pair of coordinates, with the known focal length of the camera, the two LOS angles can be calculated as shown below.

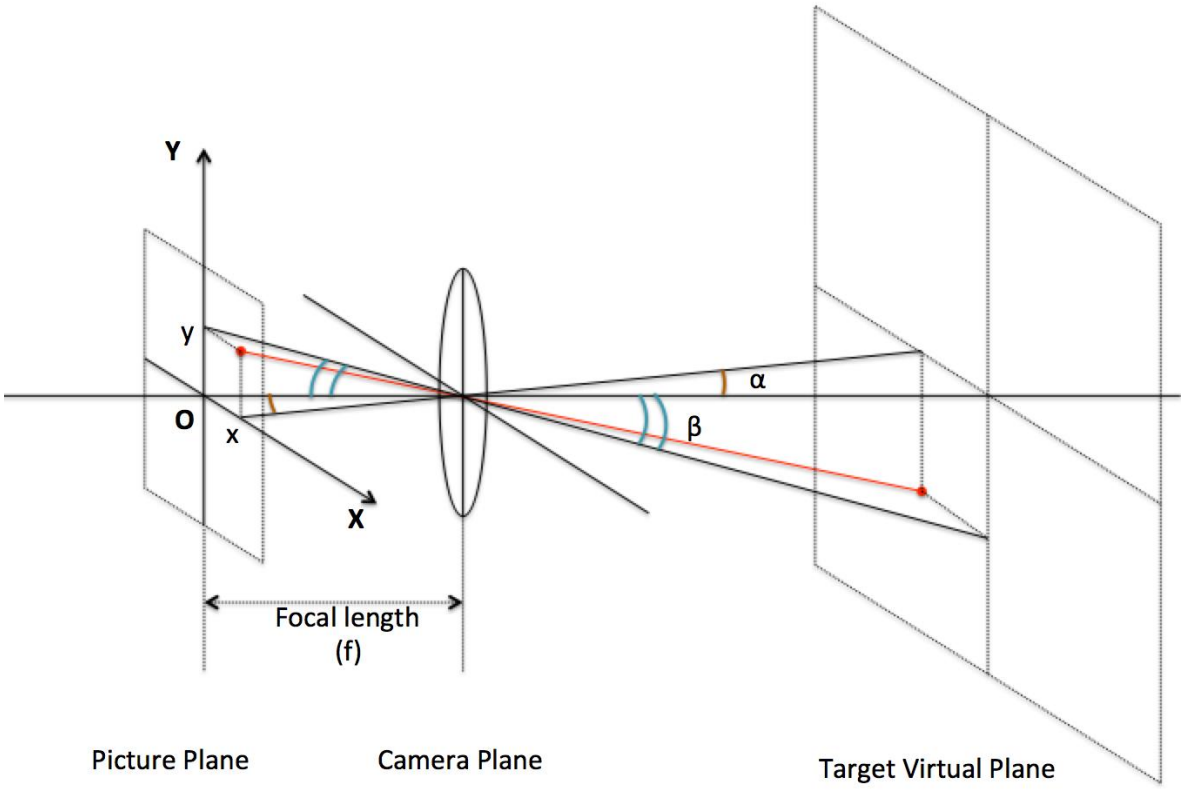


Figure 40 Line-of-sight angles

From the known values of (x, y) and the focal length of the camera (f), Figure 40 explains how the LOS angles α and β will be calculated, in other words, the direction of the target would be detected.

A pair of equations used here are:

$$\alpha = \tan^{-1}(^{x}/_{f}) \tag{8}$$

$$\beta = \tan^{-1}(^{y}/_{f}) \tag{9}$$

From these three equations: (1), (8) and (9), all the required parameters for the far range scenario of space rendezvous progress, which are: the distance (d) and LOS angles of direction (α, β) have been determined. And all those parameters make the relative navigation parameters of the target satellite with reference to the chaser's coordinate system. By knowing those, the chaser will be able to approach the target until it gets really close to each other.

Near range scenario: In this phase, there are some more parameters need to be calculated. Those parameters are: the three-axes rotated angles of the target with reference to the chaser's three-axes coordinate system. Figure 41 demonstrates how

the general rotation of an object would look like around the three axes coordinate system, assuming that the object has the shape of a cube satellite. This demonstration uses the three-dimensional (3D) coordinate system (Stover, Christopher Weisstein, 2016) originally invented by René Descartes (March 31, 1596 – February 11, 1650). By comparing the real attitude of each of the three axes of the coordinate system of the target (the blue coloured coordinate system in the picture) with the corresponding one on the standard attitude (0, 0, 0) which is known by the chaser as the reference coordinates (the red coloured coordinate system in the picture), the three angles (ϑ, θ, φ) can be calculated, they represent the real attitude of the target with reference to the chaser's coordinate system. And by knowing the values of these angles, the chaser then will know how to orient itself (along the three axes) to catch with the correct side (which has the docking or the berthing port) and with the correct attitude, as it continues getting closer to the target. Hence, for this scenario, the set of required parameters should be the distance (d), the LOS angles direction (α, β) and the attitude rotation angles (ϑ, θ, φ). Among this set, the first three parameters d, α and β can still be calculated by the same principle and algorithm as they were done in the case of far range scenario calculation. The task left is how to calculate the last three parameters (angles) by using the picture(s) from TOF camera.

The algorithm for the calculations of these three orientation angles of the attitude of the target will be created below based on the picture and the information taken by the TOF camera with the support of geometry calculation.

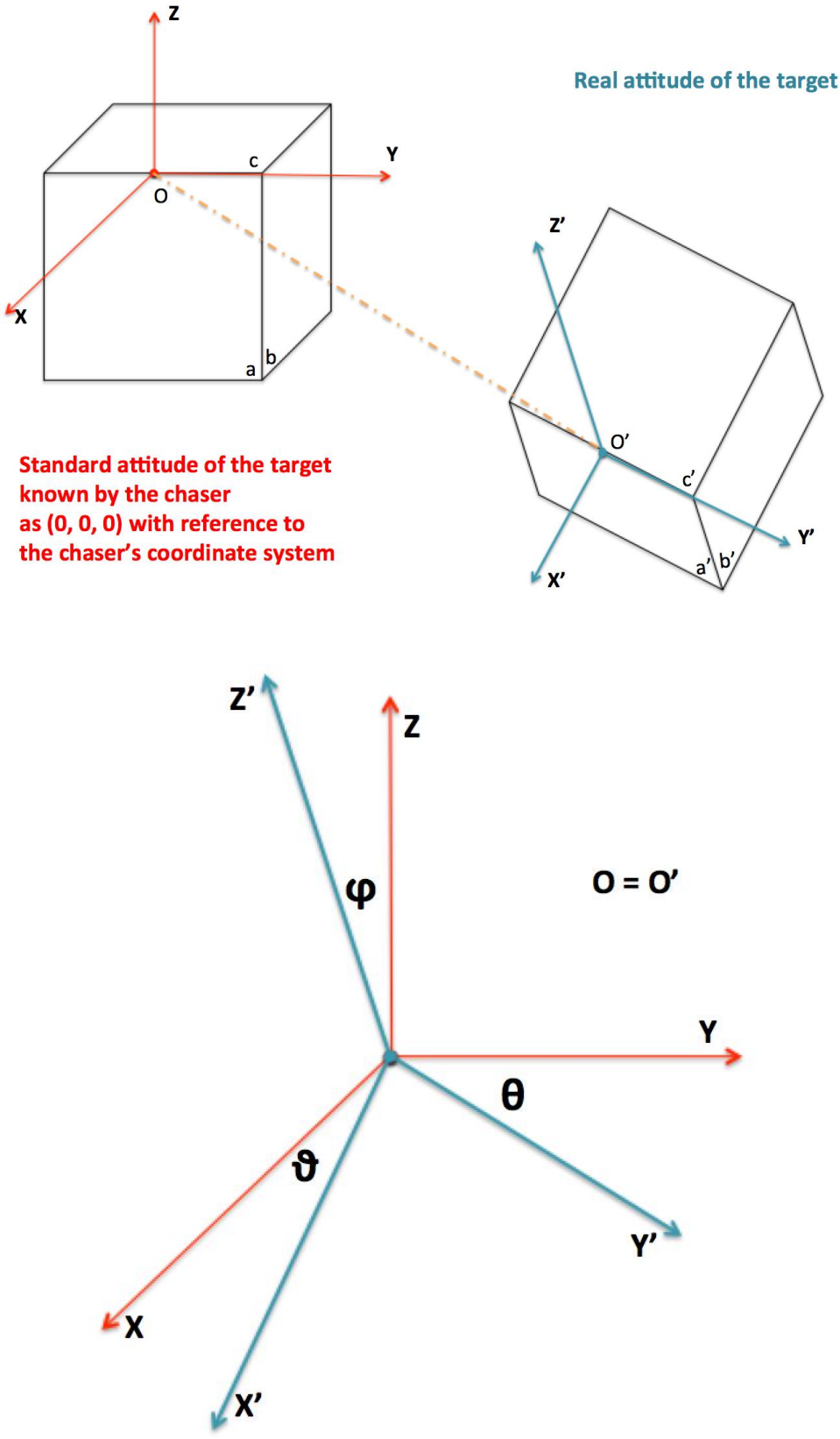
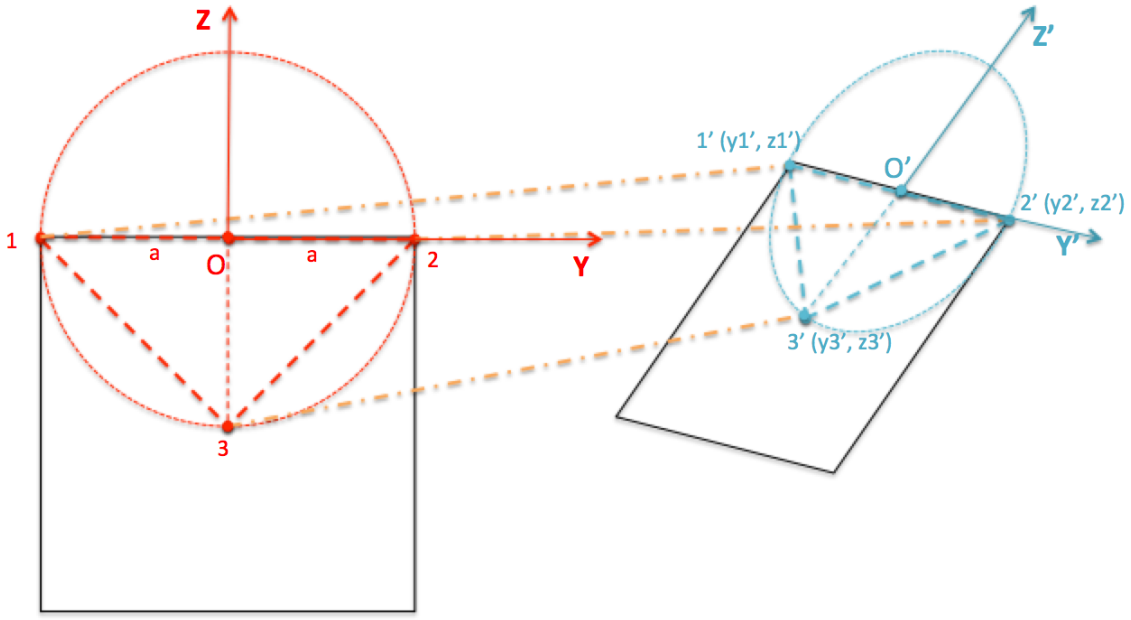


Figure 41 Demonstrate the three-axes rotation

This is the time to use the advantage of the alignment of the three CCRs in such their positions. The reason for choosing three reflectors is that in one plane, three different points define one circle (Wikipedia, Circumscribed circle, 2016). Figure 42 will be showing how those CCRs are projected to the image plane of the camera. As can be seen, the shape of the triangle made by the three reflectors is changed in the image and hence, the circle (red) became the ellipse (blue).



Standard positions of three corner-cube-reflectors known by the chaser with reference to the chaser's coordinate system assuming the viewing direction is perpendicular with the surface made of OY and OZ which is the one that the corner-cube-reflectors are attached, also the image's plane of the camera

Positions of three corner-cube reflectors in real attitude of the target are projected to an image taken by the camera

Figure 42 Image of projected objects

In the figure, the left side image (the red one) has all the known information such as the values of the distance between one to each others as they are decided by the designer, which means the value of the radius of the virtual circle **a** is known. This picture and all the known values are considered as the reference, and are constant for all the calculations afterward; the right side image (the blue one) has the information of the positions of the three CCRs **{1'(y1', z1'), 2'(y2', z2'), 3'(y3', z3')}**, these coordinate values can be detected inside the TOF camera itself. Now the problem has turned to the mathematic issue.

Focusing on the ellipse drawn by these three points (1', 2', 3'), and the fact that these points fall into the image of the TOF camera and the image of the origin O of the coordinate system now falls into O' which will be the middle point between 1' and 2', the value of the major axis |a'| can be known.

Firstly, the coordinates of O' will be calculated using the midpoint formula (Roberts, 2012b) below:

$$O' = (\frac{y1' + y2'}{2}, \frac{z1' + z2'}{2}) \tag{10}$$

Then, the major axis a' is calculated by using the distance formula (Roberts, 2012a) below:

$$\mathrm{a}' = \sqrt{(y3' - \frac{y1' + y2'}{2})^2 + (z3' - \frac{yz + z2'}{2})^2} \tag{11}$$

Now by comparing the real image (blue) and the standard image (red), there can be the possibility to calculate the three angles, which are on demand to be known. Figure 43 shows the method to find out those angles. There goes another mathematic issue.

As described in Figure 42, the viewing direction using here is perpendicular to the plane made of OY and OZ, therefore, the OX axis is not showing up, in other words, only one point of this axis can be seen in the picture, as it is O (or O'). It means when the target rotates around itself, only one angle φ is represented in the figure above that can be seen as its real value, Figure 43. This angle is the change of the orientation of the axis z, or as David A. Forsyth and Jean Ponce described in their book named "Computer Vision – A Modern Approach" page 189 (Ponce & Forsyth, 2012), the tilt angle. This, according to the figure above, can be calculated by:

$$\varphi = \arctan(\frac{y3' - \frac{y1' + y2'}{2}}{z3' - \frac{z1' + z2'}{2}}) \tag{12}$$

There are still two angles ϑ and θ to determine. Look at the change in the shape of the circle, which defined by the three points (1, 2, 3) – the red circle, in the projected image, it has become the ellipse defined by the three points (1', 2', 3') – the blue ellipse. The pair (ϑ, θ) is the cause of this transformation. Imagine if the viewing direction is changed, as if now, in Figure 44, it is perpendicular to the surface made by OX and OZ, therefor, OY now is represented as one point.

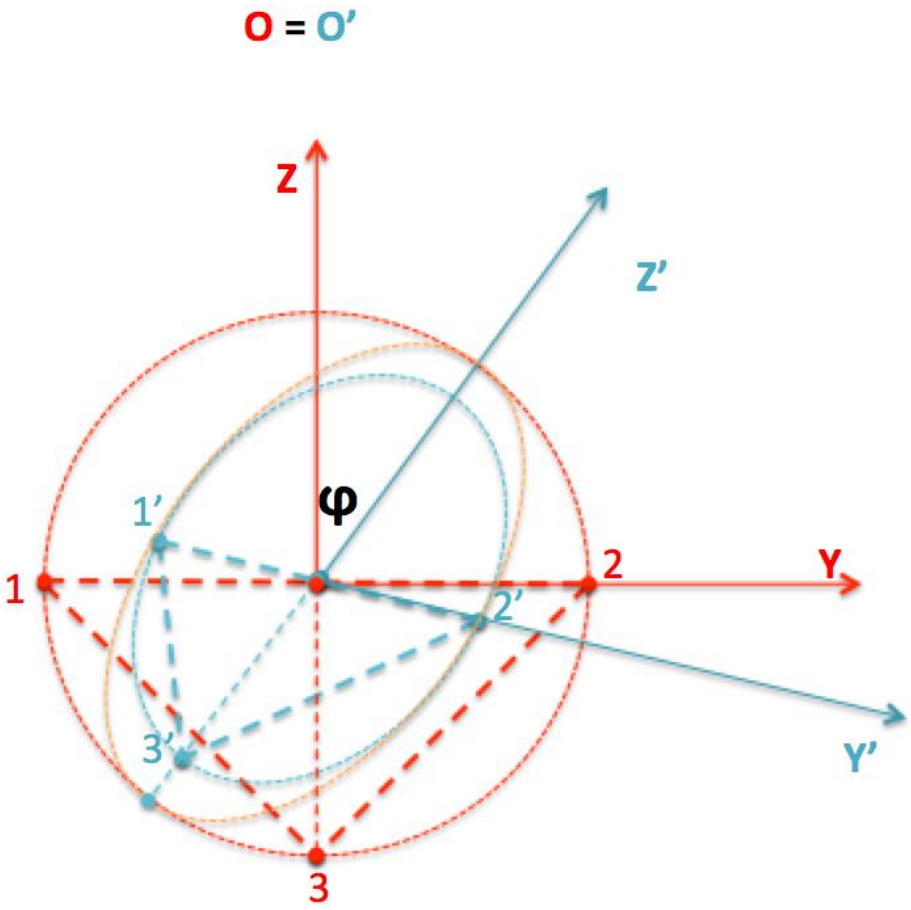
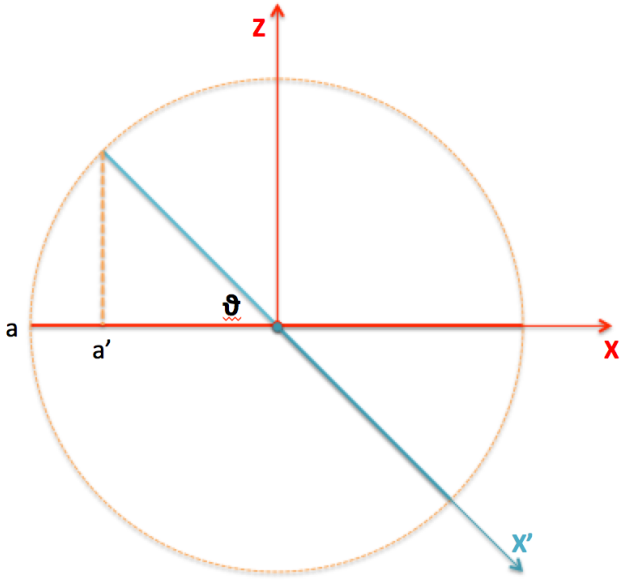


Figure 43 Application of geometry in calculation



Viewing direction is perpendicular
with the surface made of OX and OZ

Figure 44 Another view

This figure shows how the value **a** has become a'. However, the value of a' here is based on the unit of pixel. In reality, a' must be transformed to the real distance (with d was calculated from equation 1 and p is the size of one pixel, depends on the camera):

$$\mathrm{A} = \frac{d}{f} p \sqrt{(y3' - \frac{y1' + y2'}{2})^2 + (z3' - \frac{yz + z2'}{2})^2} \quad (13)$$

Thus the orientation of the axis x, the angle ϑ, is calculated by:

$$\vartheta = \arccos(\mathrm{A}/\mathrm{a}) \quad (14)$$

With a' was calculated by equation (11):

$$\vartheta = \arccos\left(\frac{dp\sqrt{\left(y3' - \frac{y1' + y2'}{2}\right)^2 + \left(z3' - \frac{z1' + z2'}{2}\right)^2}}{f\mathrm{a}}\right) \quad (15)$$

When it comes to the orientation around the axis Y, for more accurate calculation, the simple pinhole model was used. The visualization is shown in Figure 45.

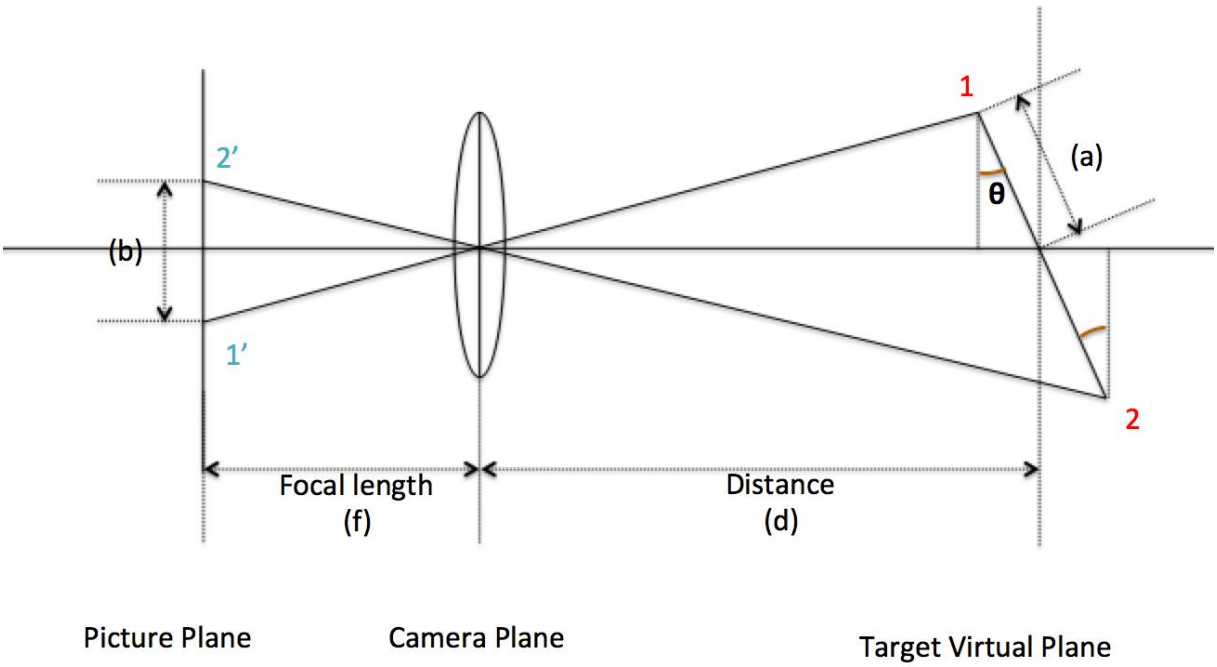


Figure 45 A pinhole model

The principle is:

$$\frac{b}{f} = \frac{acos\theta}{d - asin\theta} + \frac{acos\theta}{d + asin\theta} \tag{16}$$

With b is calculated by:

$$\mathrm{b} = \mathrm{p}\sqrt{(\mathrm{y1}' - \mathrm{y2}')^2 + (z1' - z2')^2} \tag{17}$$

Solving equation (16) with the value of b given by equation (17), known arguments p, a and f, and d is the distance was already calculated, there will be the value of θ:

$$\theta = \arccos\left(\frac{df - \sqrt{|d^2f^2 - p^2(d^2 + a^2)((\mathrm{y1}' - \mathrm{y2}')^2 + (z1' - z2')^2)|}}{ap\sqrt{(\mathrm{y1}' - \mathrm{y2}')^2 + (z1' - z2')^2}}\right) \tag{18}$$

Going throughout from the six equations: (1), (8), (9), (12), (15) and (18), the set of parameters including: the distance (d), the LOS angles direction (α, β) and the attitude rotation angles (ϑ, θ, φ) have been all determined. So then the problem of calculation is solved.

When the target pattern moving out of the picture frame of the camera: In this case, the target pattern is moving out of the filed-of-view (FOV) of the camera, the three CCRs are not all captured. The calculation now has to be done by using the shape of the central CCR. This specific shape is only necessary for the autonomous docking purpose.

The value of the FOV of the camera depends on the lens of the camera itself, this value varies for each lens, and it is indicated in the specification of the camera, also can be changed by the manufacturer if demanding, but the existence of the FOV needs to be taken into account.

The distance D, from that, the algorithm for the attitude calculation needs to be changed one more time, can be detected by this FOV and the size of the target pattern:

$$D = \frac{|a|}{\tan(\frac{FOV}{2})} \tag{19}$$

In this particular scenario, the information that can be known from the image of TOF camera is not the positions of the three CCRs' images, but the position of a block of points, which represent the image of the CCR at the centre with its special shape. Therefore, the technique of finding corners needs to be used to detect the corners of the shape, Figure 46.

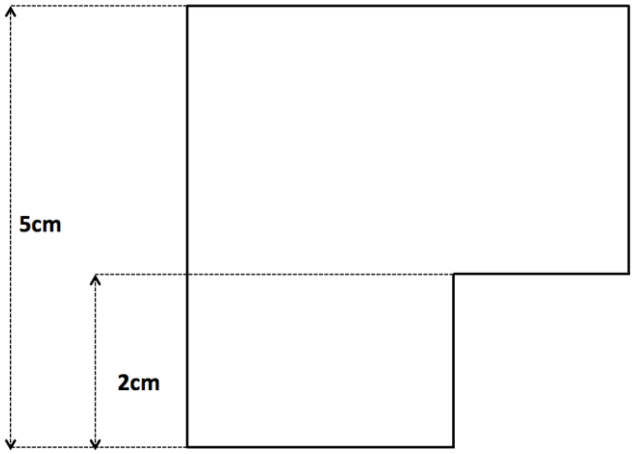


Figure 46 The special shape designed for the central reflector

This technique is, one more time, written by David A. Forsyth and Jean Ponce in section 5.3 of their book (Ponce & Forsyth, 2012). The most needed rotation parameter in this phase is the orientation angle around the vertical axis Z, or the tilt angle, because, at this close distance, the chaser will be moving almost at the same attitude with the target. After those corners are found, the calibration algorithm (Ponce & Forsyth, 2012) can be used to detect the movement of the target. There is a program for this calibration purpose already written by professor Hideaki Uchiyama, Laboratory for Image and Media Understanding, Kyushu University, Japan (LIMU, 2016), and that program might be credited to use to the further extent of this research.

b. Software (for calculation on the chaser side)

Based on the design and the analysis of using the CCRs interface above, combining with the specification of TOF camera, of what the image is made of the array of pixels, each pixel is considered as one sensor that can sense the light, which is reflected from the target, and can read the depth as well as the intensity and some other information of that light, then indicates these information right into that pixel on the image (Hansard et al., 2012). Using the functions which were written specifically for TOF type of cameras (Camera, 2014), especially the manual document from Stanley Electric CO., LTD (www.stanley.co.jp) (Stanley, 2015) to get the needed information, the program is built as showing in Figure 47 and Figure 48 below:

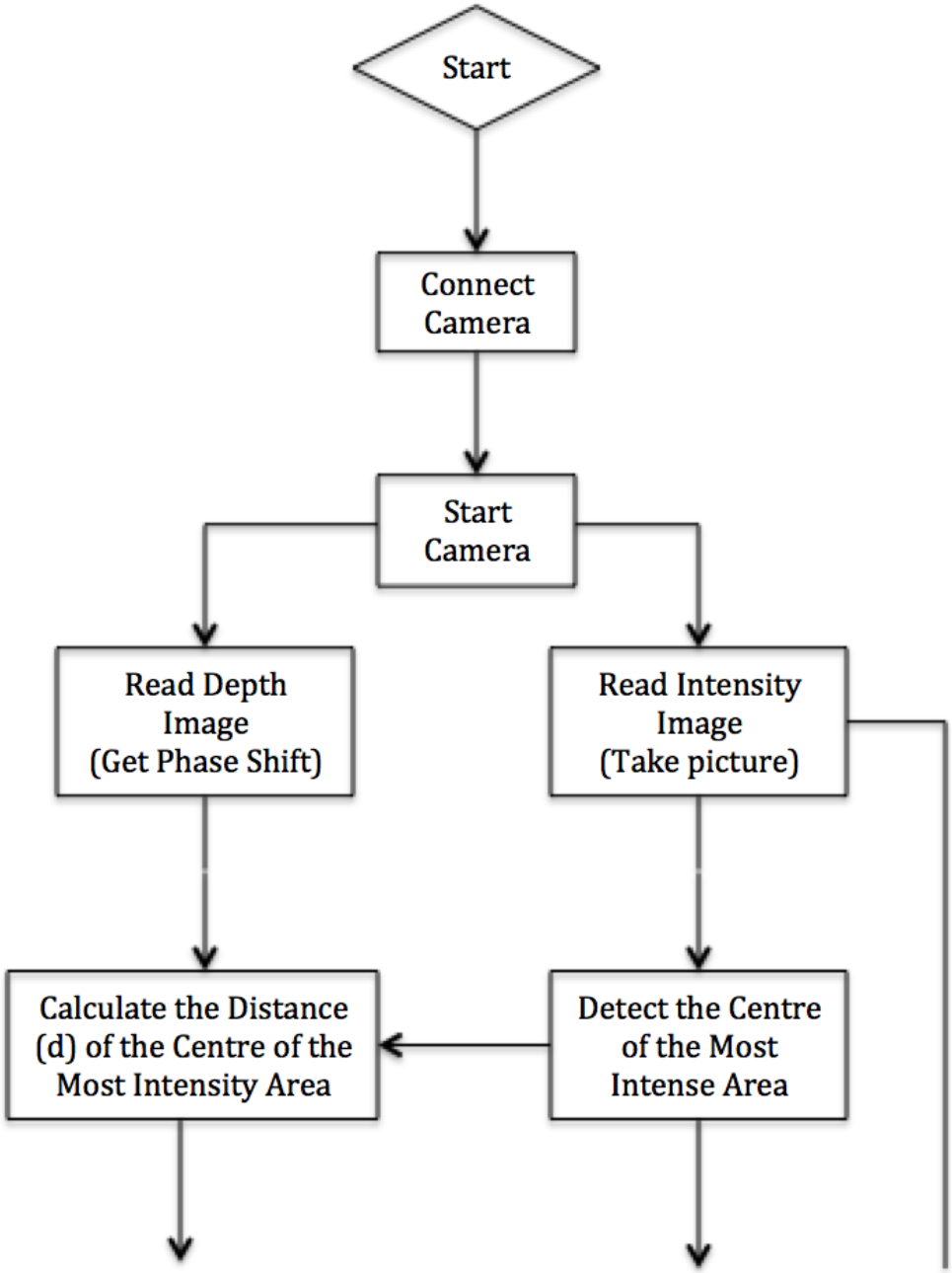


Figure 47 Software implementation diagram (part 1)

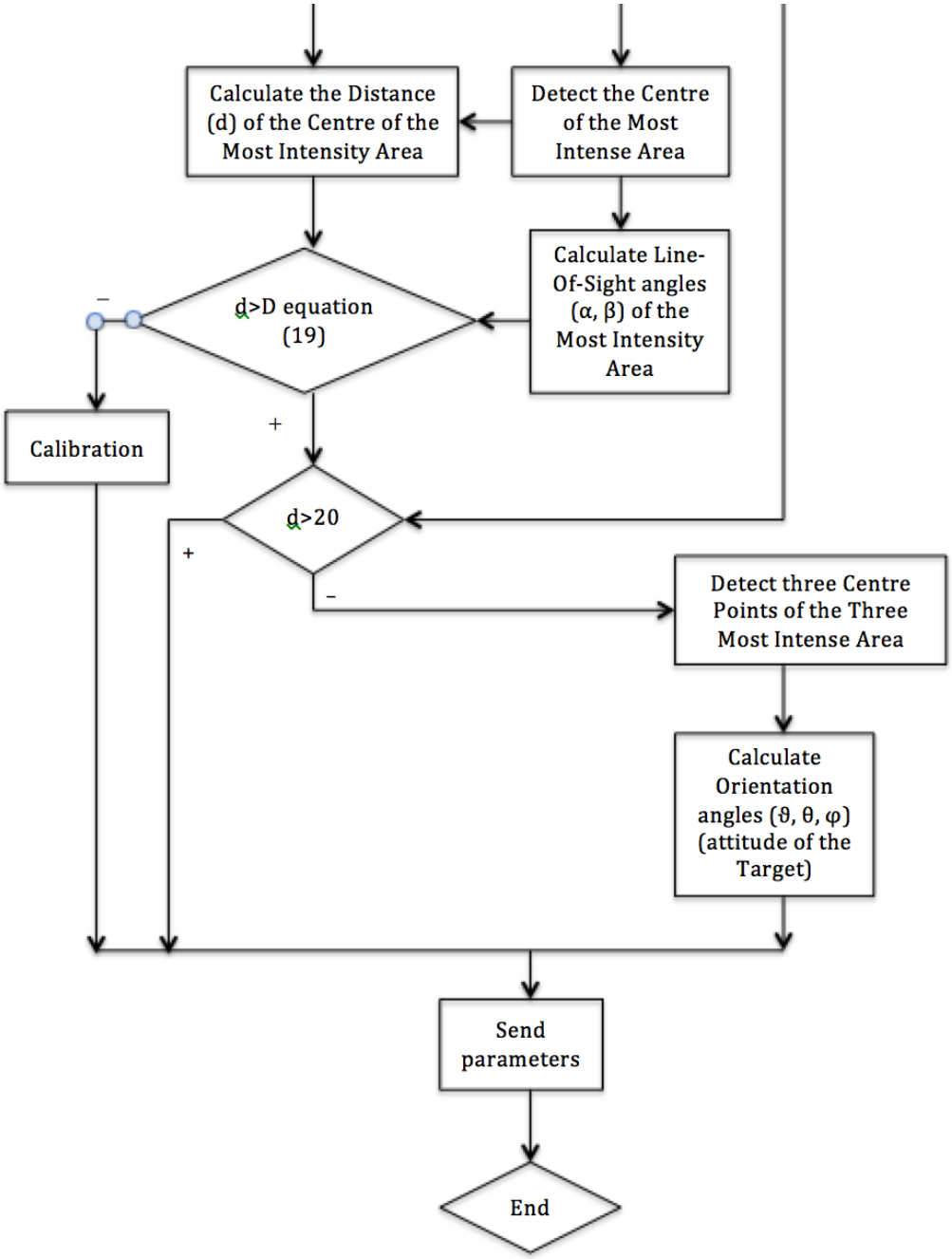


Figure 48 Software implementation diagram (part 2)

The implementation of this software was done using some tools to write a program. These tools are the C++ programming language, the OpenCV external library of image processing for calibration part (if necessary) and the TCS_DLL external library for TOF camera functions. The processor and environmental platform are optional but should be using Microsoft Windows operating system (www.microsoft.com) since there are no supports from TCS library for other operating systems such as MacOS or Linux. Therefore, the software created shall be in the format of the .exe file.

The details of the process of making this software will be described later in the part of verification activities. To be continuing, there comes the physical design of the new sensing system.

The physical design diagram

Once the interface between the target and the chaser and the software program are already completed, following all the analysis of the system, there are three subsystems already created inside the whole sensing system. They are the interface subsystem, the software subsystem and the TOF camera subsystem. And by locating the functions to the subsystems, there is the physical design to be made.

The physical design of the set of functional requirements as it is shown in Figure 49as all the lowest level functions are already allocated to the corresponding physical subsystem under the sensing system.

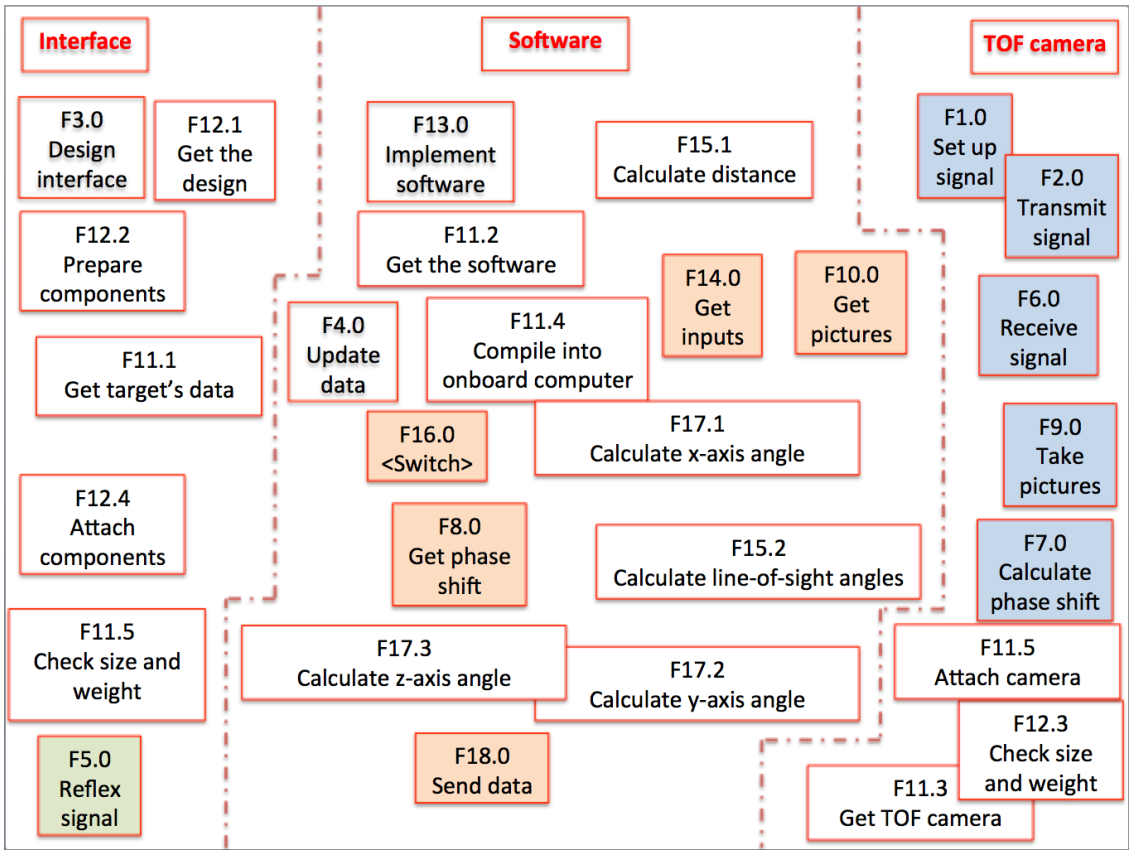


Figure 49 Physical design

3.3.2.3 Architecture Diagram

This architecture diagram will show the final output of the whole design process. In this diagram, the connections and interfaces happen among all three subsystems that were

decided in the step of making the physical design, including the functions located inside them, will be presented.

To avoid the complicated progress, the internal functions, which are the functions only happen inside one subsystem and don't affect or connect to any other functions belonged to other subsystems, will not be shown in the architecture diagram, Figure 50.

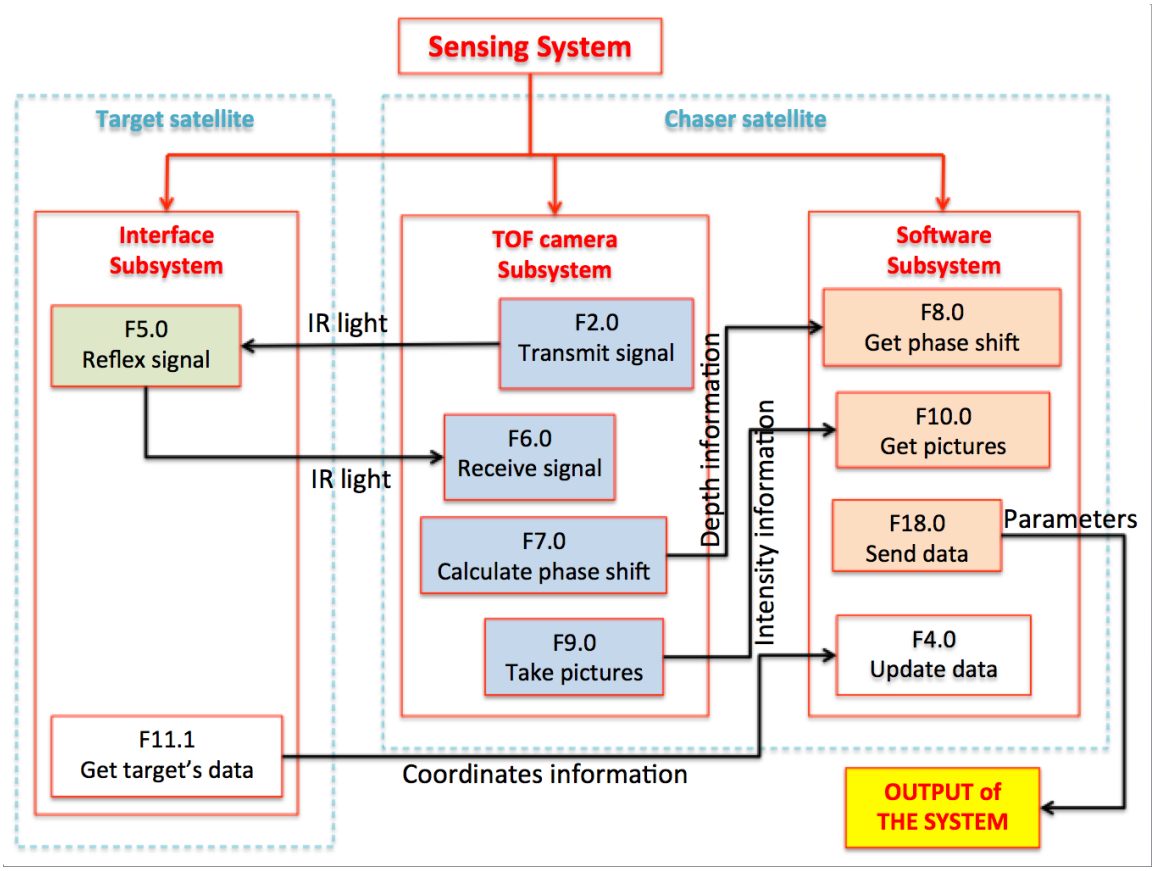


Figure 50 Architecture diagram of the new sensing system

After this architecture diagram was done, the image of how to make the new sensing system can be seen clearly. So the new sensing system is formed by three subsystems: the interface subsystem (on the target side), the TOF camera subsystem and the Software subsystem (on the chaser side). The communications between each subsystem and the two others are described on the diagram by the black arrows. Each arrow, as can be seen, comes from one function inside one subsystem to another function of another subsystem with its description defines the interface of the communication between them. By showing that, the communication between the three subsystems to one another inside the new sensing system can be determined.

And the next chapter will be talking about how this design is demonstrated by doing an experiment.

4 Research Verification by Making Prototype and Testing

This chapter is going to talk about the demonstration of the new sensing system, based on what has been designed in the previous chapter. Its results will be used to evaluate the quality of the method and to verify the feasibility of the application of the new design for the space rendezvous sensing system in reality. The complementation of the experiment has been established at Haruyama laboratory, Nishibekan, Keio University, Hiyoshi, Japan, and will be described throughout the whole chapter, started with the hardware development **(4.1)**, the software implementation **(4.2)**, followed by the experiment and the results **(4.3)**, and finally the result discussion **(4.4)**.

4.1 Hardware Development of Chaser and Target

This section describes the devices and components were used to make a prototyping model of the tests.

4.1.1 Chaser Side

There are some devices to be used in the experiment to imitate the chaser satellite. The description of both the TOF camera subsystem and the software subsystem are shown.

- For the imitation of the TOF camera subsystem:

  An example of TOF cameras used in the experiments was borrowed from Stanley Electric Co., Ltd. Figure 51 shows the camera on the tripod. The sensors array is in the centre (see the picture), and in two sides of the camera, there are the transmitters of the modulated IR lights. Specification of the camera (Stanley, 2015):

  - FOV: 72 (H) x 72 (V) degrees (H = Horizontal, V = Vertical). This means from the origin of the two-axis coordinate system of the focal plane (image frame) of the camera, the view is opened up to **36 degrees** towards all the left-hand side, the right-hand side, the upper side, and the bottom side.
  - Sensors array: 128 x 128 pixels (effective region: 126 x 126 pixels).
  - Image size: 3.84 mm x 3.84 mm (effective region: 3.78 mm x 3.78 mm).

  → From the number of pixels and the size of the image, the size of each pixel, the argument p on equation (13), (15), (17) and (18), can be calculated and given a result of:

$$p = \frac{3.78}{126} = 0.03(mm) \quad (20)$$

- Speed: 30 fps (frames per second), which means the camera transmits signals and creates images every 33.3 milliseconds.
- Focal length (f): 2.64 mm.
- Connection: LAN
    - Cable standard: 10/100Base-TX.
    - MAC address (fixed): 00 80 59 XX XX XX.
    - Protocol: TCP/IP or UDP/IP
    - IP address (default): 192.168.0.80.
    - Subnet mask: 255.255.255.0.
    - Port: 50000 or 50001 for TCP/IP, 50000 for UDP/IP

This information needs to be known and will be used inside the program to connect with the camera, and therefore, be able to get depth or intensity or other types of information from the images of the TOF camera.

Figure 51 TOF camera Stanley

- For the software subsystem:
    - One computer running Microsoft Windows 8.1 (Figure 52 and Figure 53).
    - Microsoft Visual Studio C++ Community 2015 was installed.
    - The configuration of TCS_DLL libraries.
    - Connection: LAN, DHCP client.

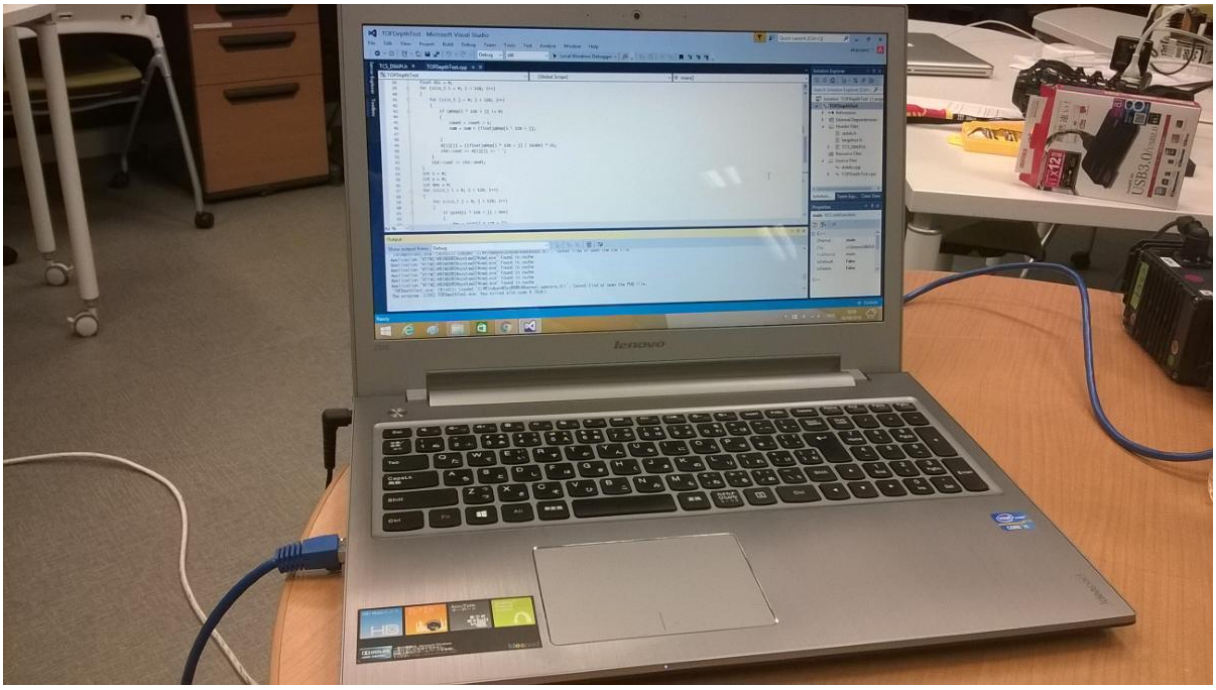


Figure 52 Windows OS 8.1 computer

Figure 53 LAN connection connects to the TOF camera

4.1.2 Target Side

A wooden plane with the size of 50 x 50 centimetres square was used to imitate one surface of the target satellite, which assumingly is a micro class satellite with the size is 50 x 50 x 50 centimetres cube. In this plane, three CCRs are attached to the position as they were designed in the physical design of the interface in **Chapter 3**. Figure 54 is the prototyping of the picture of target pattern put on a tripod.

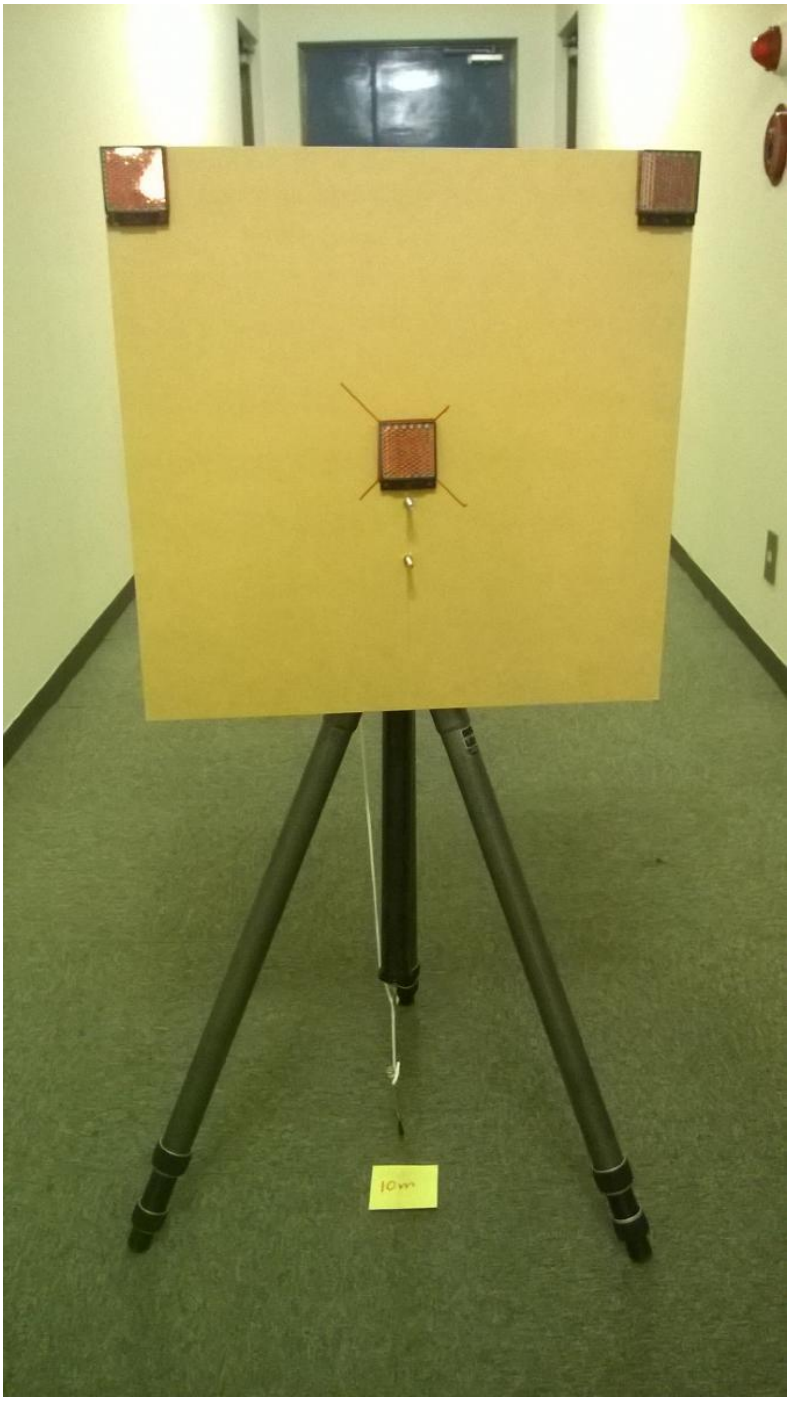


Figure 54 Target prototyping

The edge size of the plane now is known, which is 50 centimetres (0.5 metres). Then the value of a, which is the constant parameter using in equation (14), (15), (16), and (18), will be able to calculated:

$$\mathrm{a} = \frac{0.5}{2} = 0.25(m) \tag{21}$$

Therefore, the distance D calculated from equation (20) can be detected for this real model:

$$D = \frac{a}{\tan(\frac{FOV}{2})} = \frac{0.25}{\tan(36)} = 0.3441(m) \tag{22}$$

It means that with the distance d gets to 0.3441 metres, the calibration algorithm can be used if the autonomous docking was decided to be used as the very last step of the space rendezvous.

Another constant parameter that should be detected before making the experiment is the distance (L) from which the camera can be able to distinguish three different images of the three CCRs, means that if the distance gotten from the camera is smaller than this L value, the algorithm of the calculation inside the software should be switched to the one for near range scenario. This distance was assumed to be 20 metres when analysed the two scenarios in section **1.3.3**. However, it was mentioned as an assumption. This value varies depending on the camera and the requirement of the distinguishable points on the target side. This distance, in the case of this experiment can be calculated from: the edge size of the target, which is 50 centimetres, the design for the arrangement of the CCRs on the target's surface which made the smallest distance that needs to be recognised (or the value 'a' in equation (21)) become 0.25 metres or 250 milimetres, the size of each pixel (p) on the focal plane of the camera or on the image, which is 0.03 milimetres (equation (20)), and the focal length (f) of the camera, which is 2.64 milimetres (camera's specification). The equation used for this purpose is:

$$L = \frac{|a|}{p} f = \frac{250}{0.03} 2.64$$

$$L = 21999(mm) \cong 22(m) \tag{23}$$

This result means that from the distance of 22 metres, the images of the three CCRs will fall into three different pixels, and therefore, the algorithm of the near range scenario can theoretically be used. With the distance further than 22 metres, they might all fall into one pixel and in that case only the calculation algorithm of the far range scenario can be used.

After clarified the two main hold points corresponding to the distance D from equation (22) and the distance L from equation (23), the software can be implemented to do the calculation part as its process was described in Figure 48, section **3.3.2.2** with D now is 0.3441 metres.

4.2 Software Implementation

Firstly, the software application for the far range scenario was implemented using the tools and the devices as shown in section **4.1**, the algorithm was shown in the physical design section in **3.3.2**.

The software developer kit is provided in the form of the dynamic link library (DLL) (Camera, 2014). This library file was taken from Stanley Electric Co., Ltd. The library is the group of functions that were developed by a group of programmers worked on camera and image processing, especially the time-of-flight type of cameras. It was built based on the C++ programming language, so that, it can be used by any C++ program. The file received was compiled under the .dll format, which can only be read by the Microsoft Windows operating system (Wikipedia, 2016). That was why Microsoft Windows computers are recommended to use in this implementation of the software subsystem.

Table 4 explains more details about this toolkit, its files and how they can be implemented into a project written in C++ language. By merging this library into the project, the programmers can be able to use some specific functions that are particularly for the application of the TOF camera to get the information that they need (as once mentioned, for this research, the depth information and the intensity information were used).

Table 4 Software developer kit files list[33]

Supplied files

File required at application execution

TCS_Dll.dll	Library file Copy the file into the same folder as that of the program that uses DLL, the system folder of "Windows", or the folder under the same path.

Files required at application development (Used at development with VC++)

TCS_Dll.lib	Library reference file (used at link under VC++) Normally, copy the files into the same folder as that of the source file that is used for program development.
TCS_DllAPI.h	Library function & constant definition file (used by including with VC++) Normally, copy the files into the same folder as that of the source file that is used for program development.

SDK Manual

TCS_SDK.chml	Help file This file. Use this file for development since it contains function descriptions.

Another information that is also very important for the progress of making the software is the data format of the TOF camera's image, which is described in Table 5. It is important because without knowing the arrangement of the data pieces on the frame of the image, there is no way to determine the position of the light point(s) in the picture, and thus no coordinates can be detected. Look back at the design phase in **Chapter 3**, these coordinates are the basic elements for almost all the calculations of the parameters. That is why the structure of the image, or it should be understood as the way all the sensors in the array are arranged on the focal plane of the TOF camera, has to be understood. This is the essential understanding if one wants to use information from a TOF camera to put into processing.

[33] Credit from the Nippon Signal Co., Ltd (Camera, 2014)

Table 5 TOF image data format[34]

Data format
The Word data has 16 bits per pixel.
The data in one frame is arranged as follows, provided that pixel (x,y) position data is expressed as D(x,y).

D(0,0)	D(1,0)	D(2,0)	….	D(126,0)	D(127,0)
D(0,1)	D(1,1)	D(2,1)	….	:	:
D(0,2)	:	:	….	:	:
:	:	:	….	:	:
:	:	:	….	:	:
:	:	:	….	:	:
D(0,127)	D(1,127)	D(2,127)	….	D(126,127)	D(127,127)

The connection between the program and the camera was created follows the guiding structure from Mr. Yada Yusuke, Research and Development Centre, Stanley Electric Co., Ltd (see Appendix 1).

The formula used for the transformation from the depth information (which is indicated in each pixel of the depth image) to the real distance was also given in Appendix 1.

The calculations for the LOS angles were created in the C++ program, using general geometry algorithm **(Chapter 3).** The known parameters, which are constant, derived from the specification of the camera (also given by Mr. Yada Yusuke) and the design of the interface subsystem, were used inside the program.

The main TOF camera functions were used to write the program are:

- CONNECTINFO structure

  This structure prepares the protocol for the physical LAN connection, that indicates the type of connection, the IP address of the destination, the port number is using, so that, by putting the right numbers, and with the physical connection, as shown in Figure 53, the program will find exactly the TOF camera to connect with.

- TCS_Initial function

  This function uses the information given from the CONNECTINFO structure to look for the device or destination and create the connection at the beginning of the working session.

[34] Credit from the Nippon Signal Co., Ltd (Camera, 2014)

- TCS_CameraStart function

  This function starts the operation of the camera, which includes transmitting the modulated IR lights, receiving the reflected ones, calculating its specific information and putting this information into each pixel in the image.

- TCS_ReadDepthImg function

  This is the function that the programmers call whether the depth information of each pixel in the depth image is needed to use in their program. In this autonomous space rendezvous sensing system software program, the depth information is used to calculate the distance from the focal plane of the TOF camera, which is located on the chaser, to the target.

- TCS_ReadIntensityImg function

  This is the function that the programmers call whether the intensity information of each pixel in the image is needed to use in their program. In this software application for the sensing system for an autonomous space rendezvous, this information is used to detect the brightest point, which is the image of the target (in far range scenario), or the three brightest points, which are the image of the CCRs (near range scenario).

- TCS_CameraStop function

  This function is to stop operating the camera when it is not in use.

- TCS_Close function

  This function is to disconnect or close the connection between the program and the camera. Usually, it is called at the end of the program, when the working session is about to finish.

The code and the explanation of the program were written in the C++ language project with the name is “TOFDepthTest.cpp” (see Appendix 2).

The result of this software shows the distance, the direction (LOS angles), and the relative attitude if the target moves in the near range scenario, of the target on the screen and at the same time write them on the log file (log.txt, see Appendix 3). These values will be refreshed every two seconds. So as the target is moving, they will change on the screen (and will be added to the log file) with the interval time is two seconds. The software will stop running whether the button “Esc” – escape – on the keyboard of the computer is pressed. The act of pressing the “Esc” button is the imitation of the cue that assumingly will be sent into the rendezvous sensing system when the rendezvous mission is completed.

4.3 Experiment and Result

After finished preparing for both sides: the target and the chaser, the experiment was organised at Haruyama laboratory, and the result was analysed for the evaluation.

4.3.1 Testing Environmental Constraint

Due to the scope of this research, which is making the new sensing system for autonomous space rendezvous, there was the assumption that the system was put in the space environment. However, in reality, there are some limits come from the real environment on the ground, in which the test of the system was managed to happen. They are shown and analysed in this section.

- There was no vast enough space for the testing of the far range scenario, as it was expected to be up to 200 metres. It means that only for the test of the calculation algorithm, this range should be changed to the smaller number.

  → The solution was miniaturising the scale of the test.
- There were no any empty enough spaces, within the FOV of the camera (that can imitate the true environment out there in the space) to set up the experiment.

  And because this example of TOF cameras from Stanley has a quite large FOV (36 degrees from the centre towards four directions), it is difficult to find such a place.

  The consequence of this phenomenal is that not only the lights reflected from the target will be caught in the camera. There were also a lot of objects, from different materials, different types of surfaces, exist in the venue where the test took place. The lights could come to them, be reflected and caught in the camera. These noises made a lot of errors that affected the determination of the light spots (the program might have detected the wrong points, which were not the image of the target or the CCRs but other objects existing in the test environment).

  → Therefore if the accuracy was just acceptable, it means that it would get better in the real space environment, which has less noise.
- There were some different light sources existing in the environment that might have affected the recognition ability of the sensors, and might have led to a quantity of errors.

  → Not yet confirmed but the awareness of this problem is necessary).

- There is no testbed to simulate the real six degrees movement of the target. Especially, in the relative attitude determination, it was really difficult to estimate the real parameter at the approximate value.

  → This means that even in the reference parameter, there were already errors. But this factor is unavoidable, since even the testbed with precise measurement also has some tiny errors. However, the testbed would make the measurement more accurate.
- As calculated before that the value of the distance D, at which the calibration could be used, was 0.3442 metres, but as the recommendation from Stanley engineer, the effective range to use this particular camera is from 0.7 metres to 15 metres, then the calibration function was not used. But at the distance of 1 metre, that's enough to do the berthing activity, instead of autonomous docking function.

  → These test results did not cover the autonomous docking situation.

The conclusion is that there was no such approximate space environment to conduct the tests, due to the objective conditions provided to the research, but that is enough to verify the behaviour of the new sensing system in a miniaturised scale. The calibration algorithm was not used.

4.3.2 Experimentation and Test Results

Based on the constraints has clarified, the condition of the experiment was changed. The real simulation environment has become:

- The far range was miniaturised from 8 metres up to 10 metres. The near range (final approach) was started at 8 metres. The closest distance was at 1 metre.
- The LOS angle of the target was set to be ±5 degrees, only horizontally (X axis).
- The orientation angle of the target was set from 0 to 10 degrees only horizontally (it means it rotates around the vertical Z axis).

The configuration of the tests is shown in Figure 55 (including the prototyping of the target, the camera, and the computer).

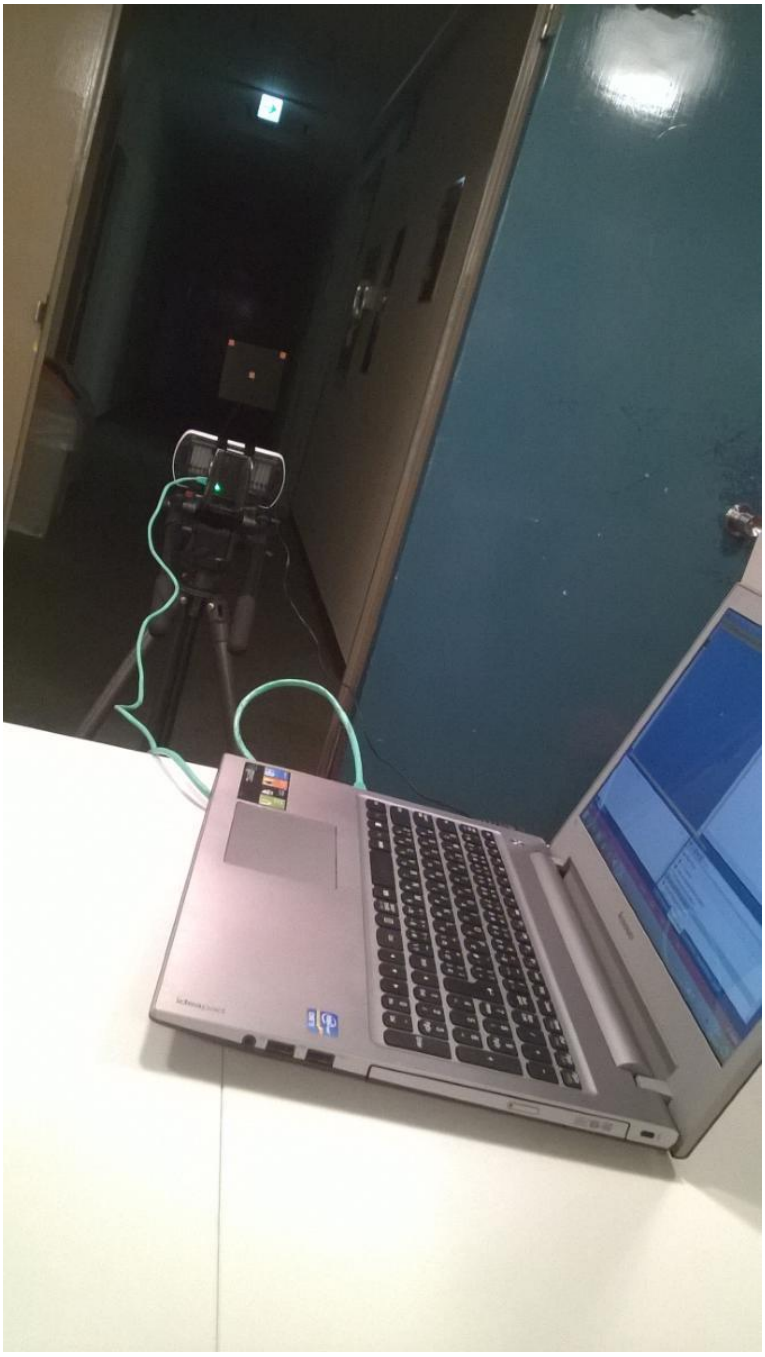


Figure 55 Testing configuration

4.3.2.1 Range Determination Test

Figure 56 shows how the test for the range determination was arranged. The range determination test was conducted with the maximum distance was 10 metres (1000 centimetres), the minimum distance was 1 metre (100 centimetres) and the changing step was 50 centimetres.

A length ruler was used to determine the distance from the camera and some pieces of paper were used to mark the distance.

Figure 56 Range determination test

By changing the position of the target and measure it in 20 seconds then averaging the numbers, the result of the test was shown in Table 6:

Table 6 Range determination

Given distance (cm)	Calculated distance (cm)	Range accuracy (cm)
100	97	3
150	156	-6
200	201	-1
250	246	4
300	297	3
350	349	1
400	395	5
450	456	-6
500	505	-5
550	548	2
600	602	-2
650	658	-8
700	692	8
750	753	-3
800	803	-3
850	858	-8
900	913	-13
950	948	2
1000	1005	-5

4.3.2.2 LOS Angle Determination Test

Instead of moving the target by some angles, which is difficult to literally measure the real value, the chaser was made to tilt itself to make the same effect. An angle ruler was made to measure the angle as can be seen in Figure 57.

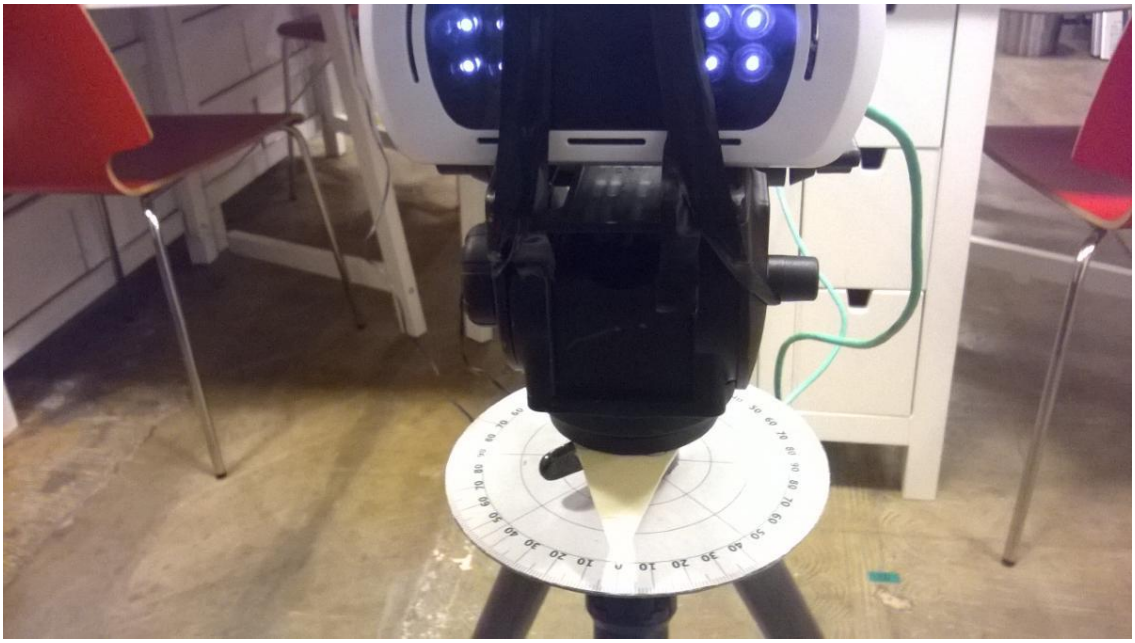


Figure 57 LOS angle determination test

After determining the horizontal axis's LOS angle of 0 degree position to put the target, the chaser was tilted degree by degree from -5 degrees to 5 degrees. For each step, the measurement has lasted for 20 seconds. The results given by the program in Table 7 were the average values of each measurement:

Table 7 LOS angle determination

Given LOS angle (degree)	Calculated LOS angle (degree)	LOS angle accuracy (degree)
-5	-5.83	0.83
-4	-4.54	0.54
-3	-3.25	0.25
-2	-2.6	0.6
-1	-0.65	-0.35
0	0	0
1	0.65	0.35
2	1.95	0.05
3	3.25	-0.25
4	3.9	0.1
5	5.19	-0.19

4.3.2.3 Attitude Angle Determination Test

To be able to measure the attitude angle, which in this test is the angle of the orientation of the target around the vertical axis, the ruler was moved to the target side as can be seen in Figure 58.

Figure 58 Attitude determination test

In the test, the target was put at the distance of 6.5 m to the chaser, and it was tilted degree by degree around its vertical axis with reference to the angle ruler attached to itself, from the value of 0 degree to the value of 10 degree towards the left-hand direction. For each step, the time measurement was 20 seconds. The results were averaged and taken to Table 8:

Table 8 Attitude angle determination

Given attitude angle with reference to the target itself (degree)	Calculated attitude angle (degree)	Adjusted attitude angle by assuming the 0 point as a norm to get the angle with reference to the chaser (degree)	Attitude angle accuracy (degree)
0	11.02	0	0
1	11.46	0.44	0.56
2	14.54	3.52	-1.52
3	16.13	5.11	-2.11
4	16.72	5.7	-1.7
5	17.42	6.4	-1.4
6	18.25	7.23	-1.23
7	18.75	7.73	-0.73
8	19.05	8.03	-0.03
9	19.91	8.89	0.11
10	20.48	9.46	0.54

4.4 Result Discussion

Based on the results, the graphs that show the measurement parameters with their accuracies (for the range, the LOS angle and the attitude angle determinations) of the method were plotted.

For the range (distance) measurement:

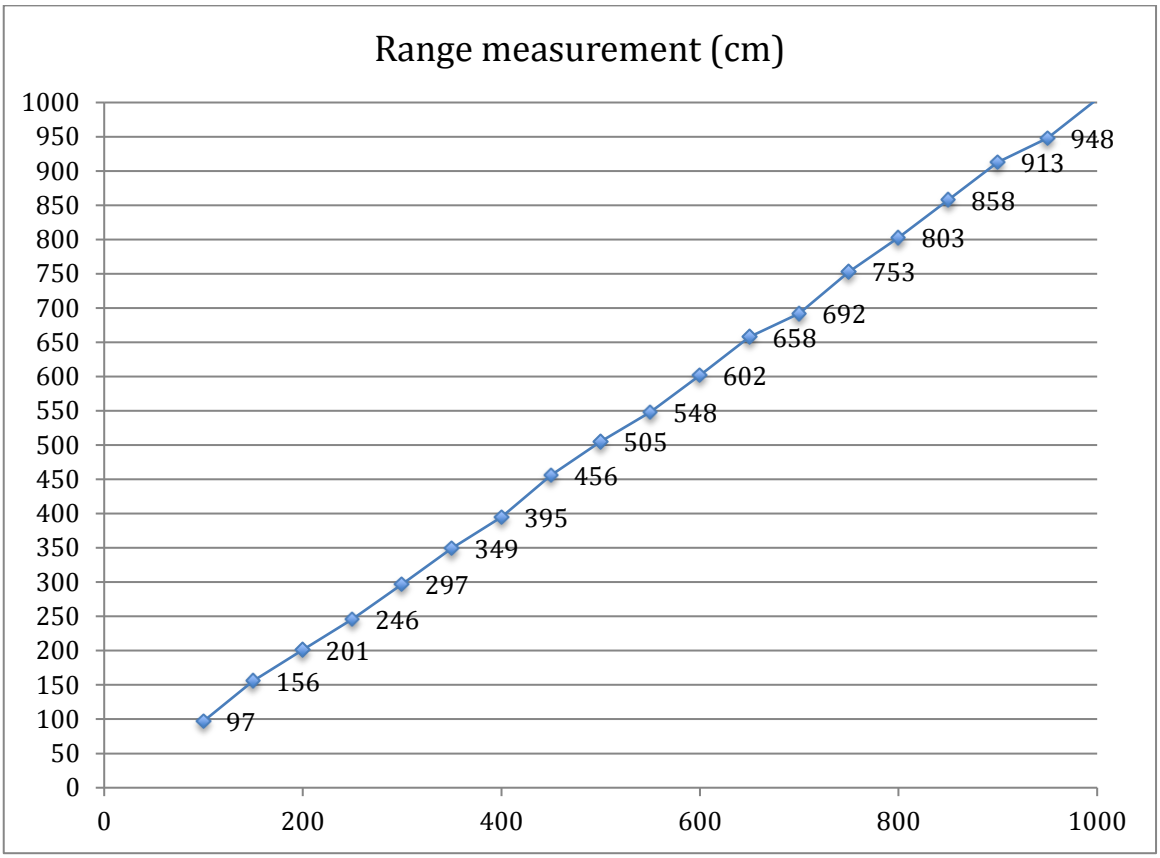


Figure 59 Range measurement

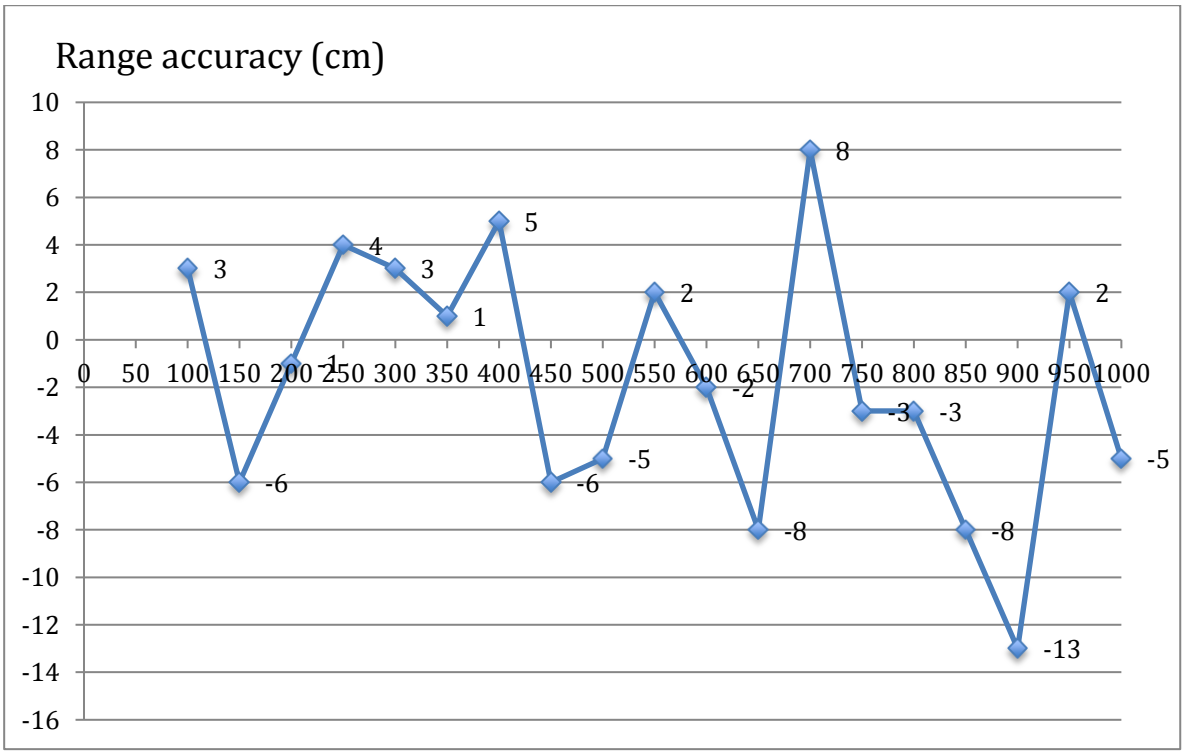


Figure 60 Range determination accuracy

For the direction LOS angle measurement:

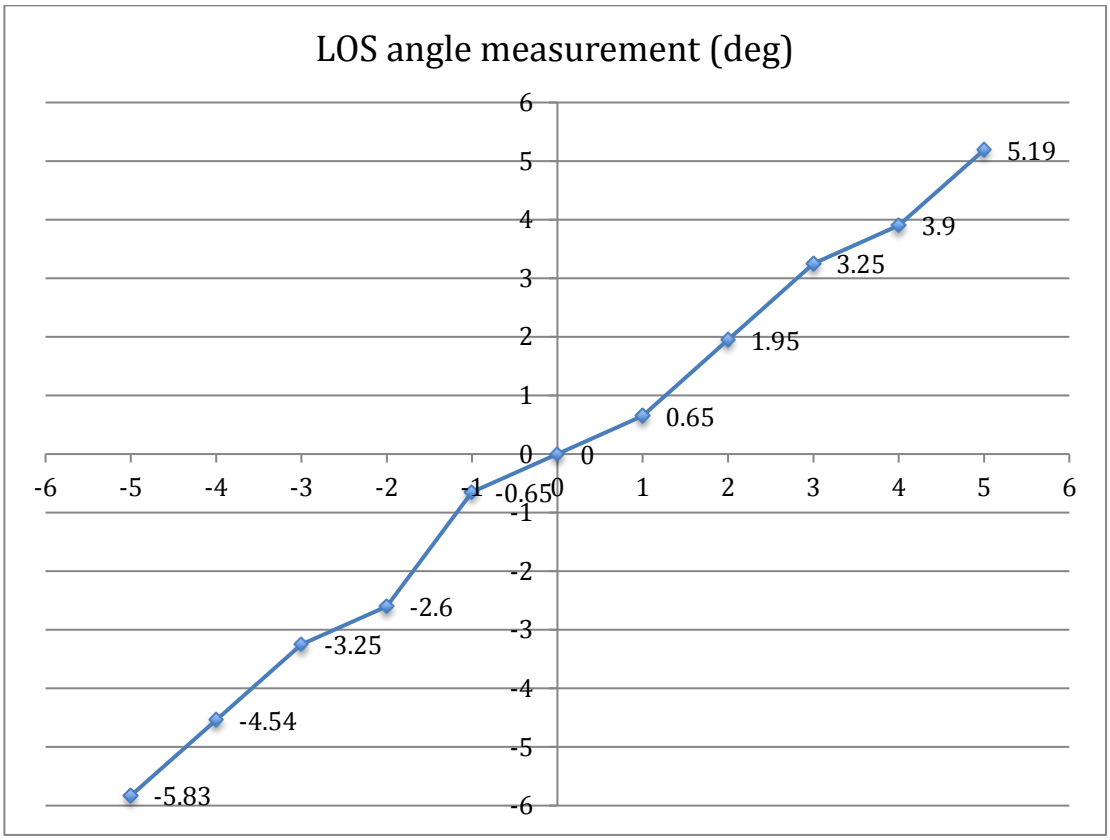


Figure 61 LOS angle measurement

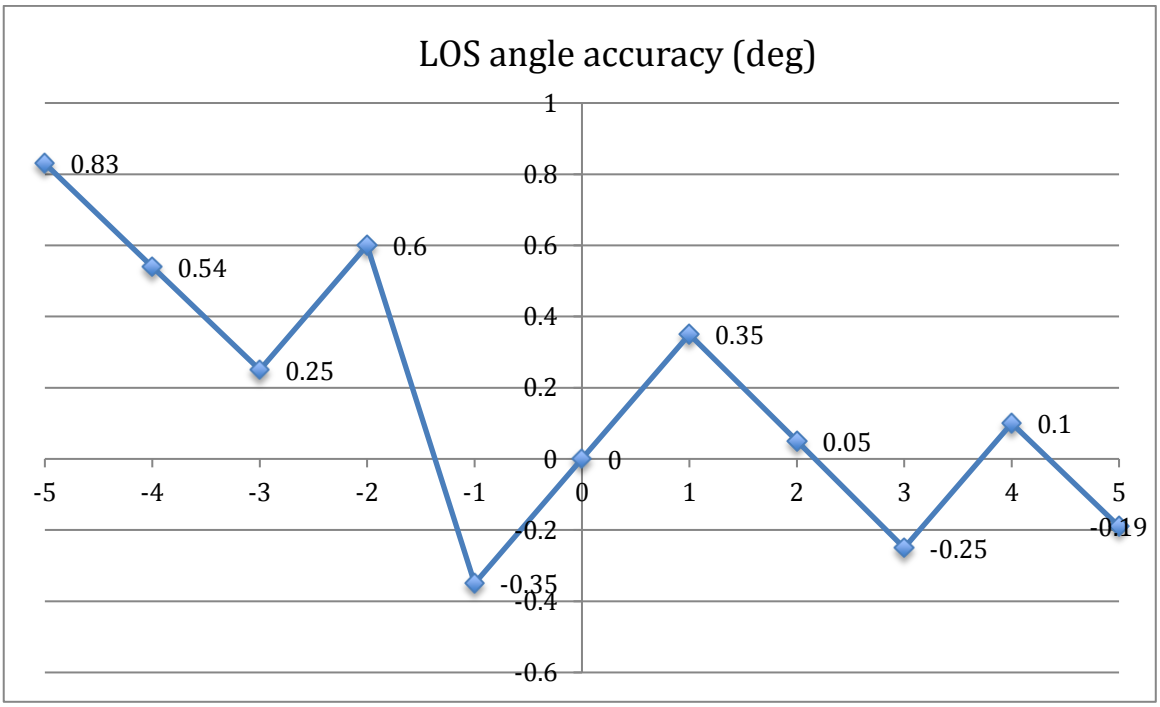


Figure 62 LOS angle (horizontal axis) determination accuracy

For the attitude angle measurement:

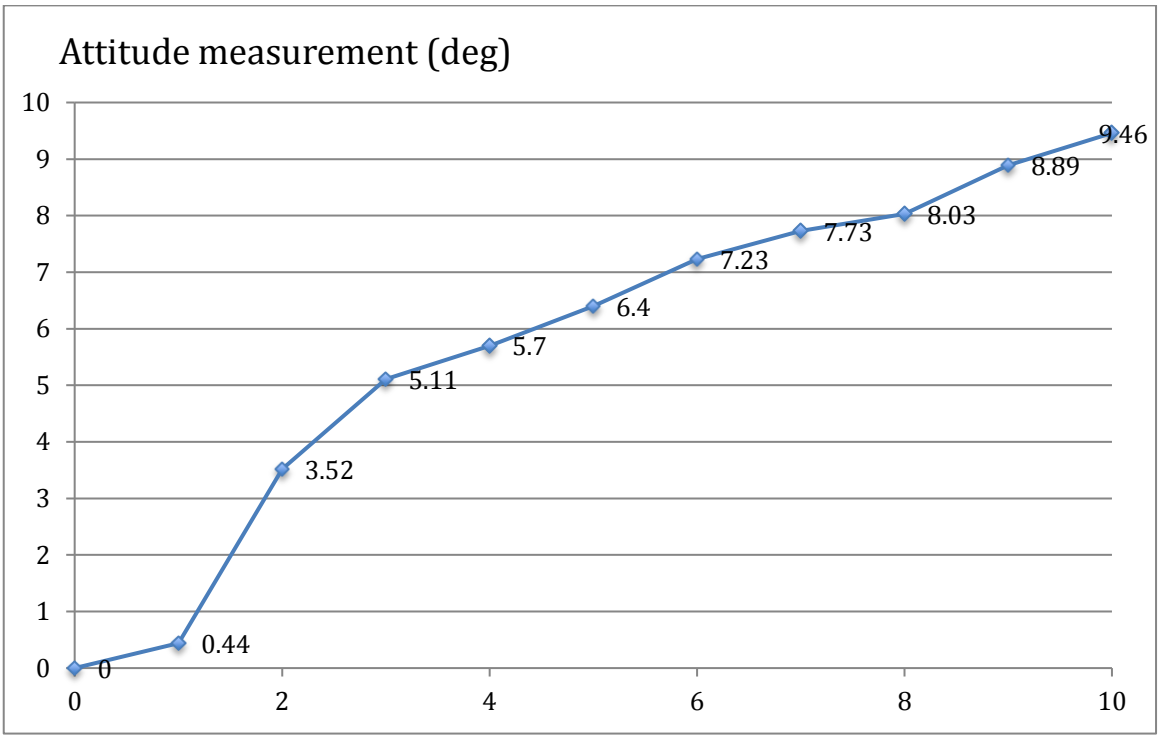


Figure 63 Attitude angle measurement

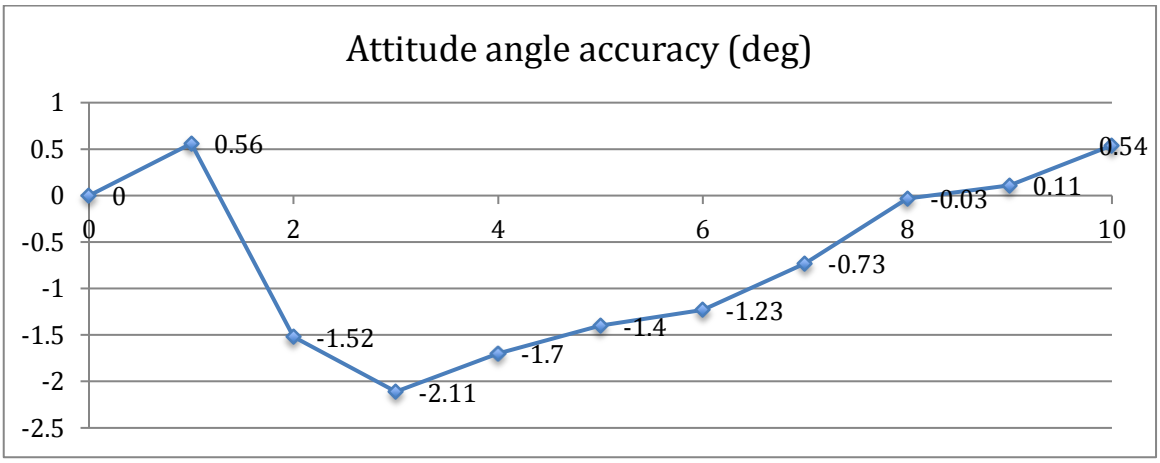


Figure 64 Attitude angle (horizontal rotation) determination accuracy

Look at Figure 60, Figure 62 and Figure 64 above and those tables of their results, there is some information can be extracted:

- In term of the range determination: the accuracy of this sensing system fell to about **4.6** centimetres in average, with the biggest error was about **13** centimetres.
- In term of the LOS angle determination: the accuracy of this sensing system fell to about **0.32** degrees in average, with the biggest error was about **0.83** degrees.

- In term of the attitude angle determination: the accuracy of this sensing system fell to about **0.47** degrees, with the biggest error was about **2.11** degrees.

Some sources were used as the references to evaluate the accuracies of the new sensing system, especially, those evaluations of other experiments from the literature review part **(2.3)**.

Look at the results of the tests of the TOF camera sensing system, the accuracy of the range and LOS angle measurement were good for most of space rendezvous scenarios, in comparative to the references. However, the accuracy of the attitude angle measurement needs to be improved if it wants to satisfy the requirements come from the autonomous docking scenario.

The errors occurred are explainable.

In the distance measurement, the error comes from the noises of the environment. The accuracy fell in the same scale with other methods'.

The LOS angle and attitude errors were caused by the characteristic of the camera itself and those equations were used. The TOF camera from Stanley that was used in the tests has the resolution of 128 x 128 pixels in the image plane of 3.84 x 3.84 milimetres square, each pixel has the size of 0.03 x 0.03 milimetres square, and the focal length is 2.64 milimetres.

In the case of the LOS angle, the error of one pixel can cause the error of **0.03 / 2.64 ≈ 0.11(36)** to the value of the tangent of a LOS angle. It can be seen that the higher resolution of the camera, the better accuracy of the sensing system.

In the case of the attitude, the equation (17) was made based on the simple pinhole model, due to the limitation of time. The result can also be better with a higher accurate image processing method. A deeper study in computer vision is needed to improve the quality of this measurement.

After collecting the results from the tests, the accuracies of the six necessary parameters given by the new sensing system showed that these results could be considered as acceptable, comparing to other results had done by researchers working in the same field. Specifically, these results were good enough to apply for:

- Far range rendezvous scenario.
- Near range rendezvous scenario with berthing method.
- Collecting space debris.
- Landing to space bodies such as comets, asteroids.

5 Conclusion and Future Work

This chapter will first summarise the research in section **(5.1)**. After that, in the second section **(5.2)**, some future works will be proposed to either improve the quality of the system, or to plan the next approaches for a higher study.

5.1 Conclusion

This research is the process of working in navigation for autonomous space rendezvous. Space rendezvous enables the important activities in the space industry, including supplying resources to the ISS and exploring the space environment. One of the important factors in a space rendezvous is the relative navigation. The measurement parameters of the relative navigation can be provided by a rendezvous sensing system. There were some rendezvous sensing systems have already been developed and used for huge spacecraft to do this task.

In recent years, the trend of small satellites brings a challenge to autonomous space rendezvous. The budget, both in finance and time, put on small satellite is limited. It is necessary to do research on small, light, cheap, and fast equipped sensing systems.

At the same time, TOF cameras are considered as cheap products and typical known for the commercial 3D picture with differentiated colours corresponding to different distances. TOF cameras use the time-of-flight principle to detect the distance between the camera and other objects.

There were some researches working on applying of this principle in the space industry, including commercial products and applications of TOF camera. However, as discussed in sections **2.3** and **2.4**, they were not the completed rendezvous sensing systems. Most of them need to work together with other devices, and some of them were not fully developed.

With some advantages in the size, the weight, and the availability, TOF cameras show the potential in space applications, followed the small satellites trend. According to the literature review, the usage of TOF cameras in space rendezvous can be extended up to hundreds of metres or almost one kilometre. It means that TOF camera can be well applied in space rendezvous. The point is: can it make the entire sensing system for autonomous space rendezvous?

This research, by applying computer vision techniques, designed the new sensing system for autonomous space rendezvous from TOF cameras. This system is small, light, simple, cheap, and fast equipped. Moreover, by the specific target pattern, this research derived the advantage of image processing technique, to make the most usage of the corner-cube-reflectors in all distance ranges of a space rendezvous, and to reduce the complexity of the software algorithm.

The tests were made and the requirements were verified.

First of all, the hardware of this sensing system including the TOF camera (5 x 5 x 20 cm cube) and the three CCRs (5 x 5 x 0.3 cm cube each) is considered as the small, light, cheap, and simple set of components, that can easily be mounted to not only huge spacecraft but also the small satellites such as those in the micro class. It means that the structure's requirement from the structure subsystem of the entire satellite, which is **the requirement "Locate"** in Figure 28 Context diagram and system boundary, to the sensing system **was verified**.

The algorithm for calculating the relative navigation parameters was transformed to the code and was easily complemented in the C++ project. The prototyping was made in order to do the tests to evaluate the sensing system with an example of TOF camera provided by Stanley. The new sensing system was able to provide the three space rendezvous relative navigation parameters for far range measurement and this number turned to six for near range measurement. This means that the calculation's requirement from the OBC subsystem of the entire satellite, which is **the requirement "Parameters"** in Figure 28 Context diagram and system boundary, to the sensing system **was verified**.

The tests and the analyses of their results were made to evaluate the accuracy and efficiency of the system. As it was shown the discussion of the test results **(4.4)**, the quality of this new sensing system is good enough for most of space rendezvous scenarios.

In the scenario of autonomous docking between the two satellites, the accuracies of angle measurements still need to be improved. It will also be discussed in the future work section **(5.2)** afterwards. In the scenario of emergency docking with non-cooperative satellite, the target's interface design should be different to provide the chaser the signs to recognise a totally unknown attitude. One possibility is assigning more CCRs to every surface of the target with different patterns.

Even though, the range scale of a space rendezvous was miniaturized, due to the limitation comes from the example TOF camera was used and the constraints of the specific

circumstance of the testing environment, it is believed to be potentially lengthened in term of measuring the distance by modifying the internal characteristic of the camera such as frequency and resolution. This was significantly proved in the literature review.

In conclusion, the new sensing system using TOF camera proposed by this research can solve the gap between the satellite trend and the existing rendezvous sensing system. The physical configuration of the new sensing system is significantly small, light, simple, low cost, and fast equipped. The special target pattern helped the programmer writes the simple code based on computer vision knowledge. The algorithm used to determine all the necessary parameters in space rendezvous navigation provides the high quality result. The application of this sensing system can be widely used in many cases of space rendezvous.

5.2 Future Work

As concluded in section **5.1**, this new sensing system was proved the advantage in autonomous space rendezvous, towards the trend of satellites in the recent years. However, techniques never stop. And there are still some difficulties throughout the research process that need to be full filled.

Theoretically, the range of measurement can be longer, and the accuracies, especially for the attitude determination, can be improved. Here are some proposals for the future works:

1. Continuing to learn deeper about this time of flight 3D sensing camera in order to: lengthen the range measurement, increase the resolution and improve the accuracies.
2. Continuing to study more about image processing and computer vision in order to alter the algorithm of the method, therefore the more accurate result can be given, especially, aim to improve the accuracies of the attitude angles' measurements, for the autonomous docking purpose.
3. Studying the new interface design for the non-cooperative targets of space rendezvous. In addition, studying about the emergency situation.
4. Continuing to find out the way to conduct an approximate space environment with less noise from surrounding objects and lights.
5. Looking for the way to get to or to organise a simulation testing system, or a test bed, that can provide the target six-degree movements, those, at the same time, are precisely measurable.

6. Studying about or cooperating with the people who work on attitude control system to be able to conduct the further tests about the behaviours of the chaser satellite for not only rendezvous but also the merging phase between the two, because eventually, rendezvous in space should be followed by the docking or berthing of the two satellites.

If these works give the positive results, the concept of TOF cameras and CCRs in autonomous space rendezvous can put an advanced point to the revolution of small satellites.

Bibliography

Aeronautics, N. (2010). Components of the ISS Components of the ISS (continued).

Amico, S. D., Pavone, M., Saraf, S., Alhussien, A., Al-saud, T., Buchman, S., … Farhat, C. (2015). Miniaturised Autonomous Distributed Space System for Future Science and Exploration. *International Workshop on Spacecraft Formation Flying (IWSCFF)*, 1–20.

Axelspace Corporation. (2015). Entering into the Satellite Image and Data Service Market with a Micro-Satellite Constellation. Retrieved from https://www.axelspace.com/en/info_/news_/20150916/20150916_/

Boge, T., Benninghoff, H., & Tzschichholz, T. (2013). Visual Navigation for on-Orbit Servicing Missions. *5th International Conference on Spacecraft Formation Flying Missions and Technologies*, (1), 1–11. Retrieved from http://www.sffmt2013.org/PPAbstract/4094p.pdf

Camera, Z. A. T. (2014). The Nippon Signal Co ., Ltd LX9657A-001 LX9657A-002, 1–24.

Dick, S. J. (2008). The Birth of NASA. Retrieved from http://www.nasa.gov/exploration/whyweexplore/Why_We_29.html

ESA. (n.d.). International Berthing Docking Mechanism (IBDM). Retrieved from http://wsn.spaceflight.esa.int/docs/Factsheets/27 IBDM.pdf

ESA. (2013). Space Debris. Retrieved from http://www.esa.int/Our_Activities/Operations/Space_Debris/FAQ_Frequently_asked_questions

Fehse, W. (2003). Automated Rendezvous and Docking of Spacecraft (p. Sensors for Rendezvous Navigation pp. 218–282). Cambridge University Press. http://doi.org/10.1017/CBO9780511543388.008

Garber, S. (2007). Sputnik and The Dawn of the Space Age. Retrieved from http://history.nasa.gov/sputnik/

Hansard, M., Lee, S., Choi, O., Horaud, R., Hansard, M., Lee, S., … Cameras, F. (2012). *Time of Flight Cameras : Principles , Methods , and Applications*.

Holder, L. a. (2013). A Survey of LIDAR Technology and its Use in Spacecraft Relative Navigation. *Journal of Navigation*, *27*, 27. http://doi.org/10.1017/S0373463300025145

IEEE. (2009). 521-2002 - IEEE Standard Letter Designations for Radar-Frequency Bands. Retrieved from http://standards.ieee.org/findstds/standard/521-2002.html

Joseph A. Angelo, J. (2006). *Rockets*. *Frontiers in Space*. Retrieved from

https://books.google.co.jp/books?id=ms6QyacXEZwC&pg=PA8&lpg=PA8&dq=rocketry+pioneers+and+visionaries+nineteenth+century&source=bl&ots=hBfcCZPoaf&sig=bZtBUrThSfO-apdgjJINTiAHUEI&hl=en&sa=X&ved=0ahUKEwjwo829tJXNAhXhUKYKHezyAA4Q6AEIGzAA#v=onepage&q=rocketry

Koelle, D. E., & Janovsky, R. (2007). Development and Transportation Costs of Space Launch Systems. *1st DGLR/CEAS Conference*. Retrieved from http://www.dglr.de/fileadmin/inhalte/dglr/fb/r1/r1_2/06-Raumtransportsysteme-Kosten.pdf

Li, L. (2014). Time-of-Flight Camera–An Introduction. *Larray Li*, (January). Retrieved from http://www.ti.com/lit/wp/sloa190b/sloa190b.pdf\nhttp://www.tij.co.jp/jp/lit/wp/sloa190/sloa190.pdf

LIMU. (2016). Hideaki Uchiyama Profile. Retrieved from http://limu.ait.kyushu-u.ac.jp/e/member/member0030.html

Loff, S. (2015). The Apollo mission. Retrieved from http://www.nasa.gov/mission_pages/apollo/missions/index.html

Marconi, E. M. (2004). Robert Goddard: A Man and His Rocket. Retrieved from http://www.nasa.gov/missions/research/f_goddard.html

May, S. (2013). What Is the Soyuz Spacecraft? Retrieved from http://www.nasa.gov/audience/forstudents/k-4/stories/nasa-knows/what-is-the-soyuz-spacecraft-k-4

Mokuno, M. (2011). ランデブドッキング用光学航法系のシステムデザイン に関する研究. Keio University.

Mokuno, M., Kawano, I., & Suzuki, T. (2004). In-orbit demonstration of rendezvous laser radar for unmanned autonomous rendezvous docking. *IEEE Transactions on Aerospace and Electronic Systems*, *40*(2), 617–626. http://doi.org/10.1109/TAES.2004.1310009

NASA. (2012). Apollo 1. Retrieved from http://www.nasa.gov/mission_pages/apollo/missions/apollo1.html

NASA. (2014). July 20, 1969: One Giant Leap For Mankind. Retrieved from http://www.nasa.gov/mission_pages/apollo/apollo11.html

Ponce, J., & Forsyth, D. (2012). *Computer vision: a modern approach*. *Computer*. http://doi.org/10.1016/j.cbi.2010.05.017

Rast, M., Schwehm, G., & Attema, E. (1999). Payload-Mass Trends for Earth-Observation and Space Exploration Satellites. *ESA Bulletin*. Retrieved from http://www.esa.int/esapub/bulletin/bullet97/rast.pdf

Regoli, L., Ravandoor, K., Schmidt, M., & Schilling, K. (2012). *Advanced Techniques for Spacecraft Motion Estimation Using PMD Sensors*. *IFAC Proceedings Volumes* (Vol. 45). IFAC. http://doi.org/10.3182/20120403-3-DE-3010.00079

Roberts, D. (2012a). Length of a Line Segment (Distance). Retrieved from http://www.regentsprep.org/regents/math/geometry/gcg3/ldistance.htm

Roberts, D. (2012b). Midpoint of a Line Segment. Retrieved from http://www.regentsprep.org/regents/math/geometry/gcg2/lmidpoint.htm

SpaceFlight101. (2016). Automated Transfer Vehicle. Retrieved from http://spaceflight101.com/spacecraft/atv/

SpaceWorks Enterprises, I. (2014). Nano/Microsatellite Market Assessment.

SpaceX. (2015). REUSABILITY: THE KEY TO MAKING HUMAN LIFE MULTI-PLANETARY. Retrieved from http://www.spacex.com/news/2013/03/31/reusability-key-making-human-life-multi-planetary

Stanley. (2015). *TOF 方式 距離画像カメラ DLL 仕様書*.

Stover, Christopher Weisstein, E. W. (2016). Cartesian Coordinates. Retrieved from http://mathworld.wolfram.com/CartesianCoordinates.html

Tzschichholz, T., & Schilling, P. D. K. (2013). *Range extension of the PMD sensor with regard to applications in space**. *IFAC Proceedings Volumes* (Vol. 46). IFAC. http://doi.org/10.3182/20130902-5-DE-2040.00004

Wertz, J. R., Everett, D. F., & Puschell, J. J. (2011). *Space Mission Engineering: The New SMAD*.

Wikipedia. (2016a). Canny edge detector. Retrieved from https://en.wikipedia.org/wiki/Canny_edge_detector

Wikipedia. (2016b). Circumscribed circle. Retrieved from https://en.wikipedia.org/wiki/Circumscribed_circle

Wikipedia. (2016c). Connected-component labeling algorithm. Retrieved from https://en.wikipedia.org/wiki/Connected-component_labeling

Wikipedia. (2016d). Retroreflector. Retrieved from https://en.wikipedia.org/wiki/Retroreflector

Wikipedia. Dynamic-Link Library. (2016). Dynamic-Link Library. Retrieved from

https://en.wikipedia.org/wiki/Dynamic-link_library

Wild, F. (2011). What Is the International Space Station? Retrieved from http://www.nasa.gov/audience/forstudents/k-4/stories/nasa-knows/what-is-the-iss-k4.html

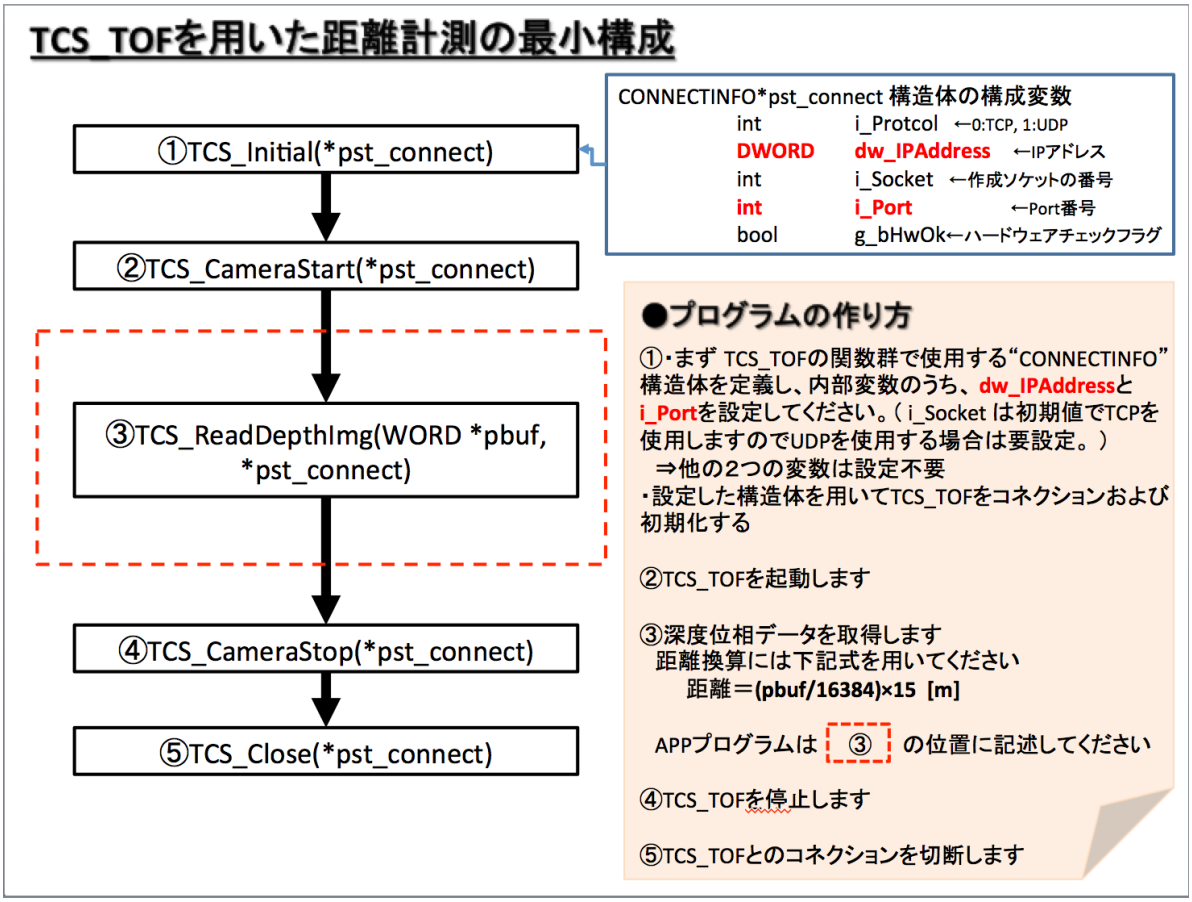
Wilfried Ley, Klaus Wittmann, W. H. (2009). Space Transportation System. In *Handbook of Space Technology* (p. 115).

Woffinden, D. C., & Geller, D. K. (2008). Navigating the Road to Autonomous Orbital Rendezvous. *Advances in the Astronautical Sciences*, *129 PART 3*(4), 2395–2415. http://doi.org/10.2514/1.30734

Wright, J. (2015). Russian Progress Spacecraft. Retrieved from http://www.nasa.gov/mission_pages/station/structure/elements/progress.html

Appendices

Appendix 1 Structure Guide of Using TOF Camera



Appendix 2 TOFDepthTest.cpp

```
#define _USE_MATH_DEFINES
#include<cmath>
#include"stdafx.h"
#include"targetver.h"
#include<math.h>
#include<windows.h>
#include<iostream>
#include<stdio.h>
#include<fstream>
#include<ctime>
#include"TCS_DllAPI.h"

//Create interface for the LAN connection
CONNECTINFO pst_connect;

//Create arrays of size 128x128
//pDep values contain depth information
//pInt values contain intensity information
WORD pDep[128 * 128], pInt[128 * 128];

//Create log file
std::ofstream file;

//Define argument for counting time
int count = 0;

//Define arguments for the labelling algorithm
int label[128][128];
int l = 0;
int count1 = 0, count2 = 0, count3 = 0;
int sumx1 = 0, sumx2 = 0, sumx3 = 0, sumy1 = 0, sumy2 = 0, sumy3 = 0;


//Main program
int main()

{
	//Point to IP adress
	pst_connect.dw_IPAddress = 0xC0A80050;

	//Indicate port
	pst_connect.i_Port = 50000;

	//Choose protocol
	pst_connect.i_Protocol = 0;
```

```
//Create connection to camera
TCS_Initial(&pst_connect);

//Start camera
TCS_CameraStart(&pst_connect);

//Open log file to be ready to write
file.open("log.txt");

//Create a while conditional loop
bool exit = 0;
while (exit == 0)
{
	//Print time by format (minute:second)
	int min = int(count / 60);
	int sec = count - (min * 60);
	file << min << ":" << sec<<"\n-----\n";

	//Get intensity information from TOF image
	TCS_ReadIntensityImg(pInt, &pst_connect);

	//Define an argument for distance
	float dis = 0;

	//Define the arguments for coordinates
	int x = 0;
	int y = 0;

	//Define an argument for one intensity value
	int den = 0;

	//Scan image to find out the most intense point
	for (size_t i = 0; i < 128; i++)
	{
		for (size_t j = 0; j < 128; j++)
		{
			if (pInt[i * 128 + j] > den)
			{
				den = pInt[i * 128 + j];
				x = i;
				y = j;
			}
		}
	}
	//---------------------------------

	//Print out to check some information of the centre
	float cen = 15 * (float)pDep[64 * 128 + 64] / 16384;
```

```
                printf("test\ndepth information of the centre: %d\ndistance of centre point
(metre): %f metres\n", pDep[64 * 128 + 64], cen);

                //Write information of centre point to log file
                file << "Center depth: "<< pDep[64 * 128 + 64] << "\n" << "Centre distance:
"<< cen << "\n\n";

                //Pront out to check the intensity value of the centre
                printf("centre point intensity: %d\n\n", pInt[64 * 128 + 64]);
                //-------------------------------------------------------------

                //Calculate distance of the most intense point

                //Get depth information
                TCS_ReadDepthImg(pDep, &pst_connect);

                //Calculate the distance by the formula from the manual
                dis = 15 * (float)pDep[x * 128 + y] / 16384;

                //Display the distance and the coordinates of the most intense point
                printf("distance (metre): %f metres\n", dis);
                printf("coordinates (x, y): %d, %d\ndepth information: %d\n", y - 64, 64 - x,
pDep[x * 128 + y]);
                //---------------------------

                //Print out to check the most intense value
                printf("most intense value: %d\n\n", pInt[x * 128 + y]);
                //-------------------------------------------------------------

                //Define the arguments for line-of-sight angles
                float alpha, beta;

                //Calculate line-of-sight angles
                alpha = atan(((y - 64)*3.84 / 128) / 2.64) / M_PI * 180;
                beta = atan(((64 - x)*3.84 / 128) / 2.64) / M_PI * 180;

                //Display the values of line-of-sight angles
                printf("line-of-sight (two axes angles - dgree):\nalpha = %f degrees\nbeta = %f
degrees\n\n", alpha, beta);

                //Write information of distance and direction into the log file
                file << "Most intense depth: " << pDep[x * 128 + y] << "\n";
                file << "Most intense coordinate (x, y): (" << y - 64 << ", " << 64 - x <<
")\n\n";
                file << "Distance: " << dis << " metres\n";
                file << "Direction: LOS angles\n";
                file << "Horizontal angle: " << alpha << " degrees\n";
                file << "Vertical angle: " << beta << " degrees\n\n\n";
```

```
//Start calculations for near range senario
if (dis < 20)
{
	//Run the for loop to label three brightest points (3 CCRs)
	label[0][0] = 0;
	for (size_t i = 1; i < 128; i++)
	{
		for (size_t j = 1; j < 128; j++)
		{
			if (pInt[i * 128 + j] <= 20000) label[i][j] = 0;
			if (pInt[i * 128 + j] > 20000)
			{
				if (label[i-1][j-1] != 0)
				{
					label[i][j] = label[i-1][j-1];
				}
				else
				{
					if (label[i - 1][j] != 0)
					{
						label[i][j] = label[i - 1][j];
					}
					else
					{
						if (label[i - 1][j + 1] != 0)
						{
							label[i][j] = label[i - 1][j +
1];

						}
						else
						{
							if (label[i][j - 1] != 0)
							{
								label[i][j] =
label[i][j - 1];

							}
							else
							{
								l = l + 1;
								label[i][j] = l;
							}
						}
					}
				}
				if (label[i][j] == 1)
				{
					count1++;
					sumx1 = sumx1 + i;
					sumy1 = sumy1 + j;
```

```
                                        }
                                        else
                                        {
                                                if (label[i][j] == 2)
                                                {
                                                        count2++;
                                                        sumx2 = sumx2 + i;
                                                        sumy2 = sumy2 + j;
                                                }
                                                else
                                                {
                                                        if (label[i][j] == 3)
                                                        {
                                                                count3++;
                                                                sumx3 = sumx3 + i;
                                                                sumy3 = sumy3 + j;
                                                        }
                                                }
                                        }
                                }
                        }
                //Finish labelling

                //Find the coordinates of three central points
                float y1 = (float)sumx1 / count1;
                float z1 = (float)sumy1 / count1;
                float y2 = (float)sumx2 / count2;
                float z2 = (float)sumy2 / count2;
                float ya = (float)sumx3 / count3;
                float za = (float)sumy3 / count3;

                //Print out the coordinates of three points
                printf("three points:\n1(%f, %f)\n2(%f, %f)\na(%f, %f)\n", z1 - 64, 64 -
y1, z2 - 64, 64 - y2, za - 64, 64 - ya);

                //Write to file
                file << "Point 1: (" << z1 - 64 << " ," << 64 - y1 << "\n";
                file << "Point 2: (" << z2 - 64 << " ," << 64 - y2 << "\n";
                file << "Point 3: (" << za - 64 << " ," << 64 - ya << "\n";

                //Calculate the major axis, the minor axis and the three-axes angles

                //Define argumets
                float a, b, phi, deta, theta;

                //Calculate the major axis
                a = sqrt(pow((ya - (y1 + y2) / 2), 2) + pow((za - (z1 + z2) / 2), 2));
```

```
                        //Calculate the minor axis
                        if (y1 == y2)
                        {
                                b = 0;
                        }
                        else
                        {
                                if (a == 0)
                                {
                                        b = sqrt(pow((y1 - y2), 2) + pow((z1 - z2), 2));
                                }
                                else
                                {
                                        b = a*sqrt(abs((pow(z2, 2) - pow(z1, 2)) / (pow(y1, 2) -
pow(y2, 2))));
                                }
                        }

                        //Print out the major and minor axes to check
                        printf("major axis: %f mm\nminor axis: %f mm\n\n", a*dis * 1000 *
0.03 / 2.64, b*dis * 1000 * 0.03 / 2.64);

                        //Write to file
                        file << "Two axes of the ellipes: " << a*dis * 1000 * 0.03 / 2.64 <<
"mm and " << b*dis * 1000 * 0.03 / 2.64 << "mm\n";

                        //Calculate the phi angle
                        if (za == (z1 + z2) / 2)
                        {
                                phi = 90;
                        }
                        else
                        {
                                phi = atan((ya - (y1 + y2) / 2) / (za - (z1 + z2) / 2)) / M_PI *
180;
                        }

                        //Calculate the deta angle
                        deta = acos((a*dis * 1000 * 0.03 / 2.64) / 250) / M_PI * 180;

                        //Calculate the theta angle
                        theta = acos((b*dis * 1000 * 0.03 / 2.64) / 250) / M_PI * 180;

                        //Print out the attitude
                        printf("attitude (three-axes angles - degree):\nphi = %f\ndeta
= %f\ntheta = %f\n\n", phi, deta, theta);

                        //Write the attitude information into the log file
                        file << "Attitude: Three-axes angles\n";
```

```
                file << "Orientation by X axis: " << phi << " degrees\n";
                file << "Orientation by Y axis: " << deta << " degrees\n";
                file << "Orientation by Z axis:" << theta << " degrees\n";
            }
            file << "-----------------------------------------\n\n\n";

            //Get out of the loop
            std::cout << "press esc to escape!" << std::endl;

            //Wait 2 seconds before starting a new loop
            Sleep(2000);
            count += 2;

            //Clean screen for writing new values
            system("cls");
            if (GetAsyncKeyState(VK_ESCAPE))
            {
                //Stop camera
                TCS_CameraStop(&pst_connect);

                //Close connection to camera
                TCS_Close(&pst_connect);

                exit = true;
            }
        }
        return 0;
}
```

Appendix 3 A sample of log file “log.txt”

0:8

Distance: 5.16083 metres

Direction: LOS angles
Horizontal angle: 3.90049 degrees
Vertical angle: -2.60256 degrees

Attitude: Three-axes angles
Orientation by X-axis: -10.3049 degrees
Orientation by Y-axis: 64.0797 degrees
Orientation by Z-axis: 90 degrees
