

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE
ENGINEERING AND TECHNOLOGY

DESIGN AND CONTROL OF AN AUTONOMOUS BLIMP

M.Sc. THESIS

Ertuğrul BAYRAKTAR

Department of Mechatronics Engineering

Mechatronics Engineering Graduate Programme

JUNE 2013

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE
ENGINEERING AND TECHNOLOGY

DESIGN AND CONTROL OF AN AUTONOMOUS BLIMP

M.Sc. THESIS

Ertuğrul BAYRAKTAR
(518101078)

Department of Mechatronics Engineering
Mechatronics Engineering Graduate Programme

Thesis Advisor: Assist. Prof. Dr. Pinar BOYRAZ

JUNE 2013

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

OTONOM HAVA ARACI (ZEPLİN) TASARIMI VE KONTROLÜ

YÜKSEK LİSANS TEZİ

**Ertuğrul BAYRAKTAR
(518101078)**

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Lisansüstü Programı

Tez Danışmanı: Yrd. Doç. Dr. Pınar BOYRAZ

HAZİRAN 2013

Ertuğrul Bayraktar, a **M.Sc.** student of **ITU Graduate School of Science Engineering and Technology** student ID **518101078**, successfully defended the thesis entitled “**DESIGN AND CONTROL OF AN AUTONOMOUS BLIMP**”, which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Assist. Prof. Dr. Pınar BOYRAZ**
Istanbul Technical University

Jury Members : **Prof. Dr. Eşref EŞKİNAT**
Boğaziçi University

Prof. Dr. Ata MUĞAN
Istanbul Technical University

Date of Submission : 03 May 2013
Date of Defense : 03 June 2013

To my family,

FOREWORD

This thesis was written for my Master of Science degree in Mechatronics Engineering at Istanbul Technical University.

I am grateful to following people and establishments for their responsibility, support and help. Firstly, I would like show my appreciation and respect to my supervisor Assist. Prof. Dr. Pınar BOYRAZ for her encouragement, guidance and heartiness.

Secondly, I would like to give my special thanks to my colleagues Mr. Ziya Ercan, Mr. Hasan Heceoğlu, Mr. Alper Öner and Mr. Volkan Sezer, PhD. for sharing their knowledge, experience and valuable time. Additionally, I am very grateful to Prof. Dr. Metin GÖKAŞAN and Prof. Dr. Ata MUĞAN, co-founders of Mechatronics Education and Research Center, for providing us to benefit from the opportunities of Mechatronics Education and Research Center. I sincerely thank Mechatronics Education and Research Center employees Mr. Alev Keskin and Mr. Reyhan Kenan for their help. Besides I would like to thank Voras Arge Co-Founders Mr. Mehmet Akif Ceylan, Mr. Emre Akça and Mr. Çağrı Dikilitaş.

Finally, I would like to thank my parents Recai & Hüsniye Bayraktar and my sisters Aslı Büşra and Sena Bayraktar for their constant encouragements and goodwill during the time I studied.

May 2013

Ertuğrul BAYRAKTAR
(Mechanical Engineer)

TABLE OF CONTENTS

	<u>Page</u>
FOREWORD.....	ix
TABLE OF CONTENTS.....	xi
ABBREVIATIONS	xiii
LIST OF TABLES	xv
LIST OF FIGURES	xvii
DESIGN AND CONTROL OF AN AUTONOMOUS BLIMP	xix
SUMMARY	xix
ÖZET.....	xxi
1. INTRODUCTION.....	1
1.1 Use of Unmanned Airship in Research.....	1
1.2 Components of an Autonomous Blimp.....	2
1.2.1 Envelope.....	3
1.2.2 Gondola.....	3
1.2.3 Fins.....	3
1.2.4 Remote control device.....	3
1.3 Previous Projects and Their Application Areas	4
1.3.1 Commercial airship projects	4
1.3.2 Academic airship projects	10
1.3.3 Military airship projects	12
1.4 Aim and Objective of This Thesis	15
2. LITERATURE REVIEW.....	17
2.1 Computer Vision in Blimp	17
2.2 Navigation, Localization and Mapping.....	18
2.3 Control and Co-Operative Control.....	19
3. PROBLEM DEFINITION	21
3.1 Purpose of This Thesis	21
3.2 Hardware Integration	22
3.3 The Schematic Workflow of Whole System.....	24
4. MATHEMATICAL MODELING.....	27
4.1 Kinematic Model.....	28
4.2 Dynamical Model.....	32
4.2.1 Mass and inertia	32
4.2.2 Coriolis effect (F_c)	34
4.2.3 Buoyancy and gravity forces (F_g)	35

4.2.4	Aerodynamic forces (F_a)	35
4.2.5	Propulsion forces (F_p)	36
4.3	Longitudinal and Lateral Models, LQR Method.....	37
4.3.1	Longitudinal equations	38
4.3.2	Lateral equations	38
4.3.3	Linear quadratic regulation	39
4.4	Simulation Model of Blimp.....	41
5.	EXPERIMENTS AND RESULTS.....	47
5.1	Vision Experiments	47
5.2	Localization Experiments.....	52
6.	CONCLUSION AND RESULTS.....	55
	Bibliography	57
	APPENDICES	61
	APPENDIX A.1	62
	APPENDIX A.2	63
	APPENDIX A.3	64
	APPENDIX A.4	65
	APPENDIX B.1.....	66
	APPENDIX C.1.....	67
	CURRICULUM VITAE.....	69

ABBREVIATIONS

AHRS	: Attitude and Heading Reference System
APS	: Active-Pixel Sensor
CB	: Center of Buoyancy
CCD	: Charged-Coupled Device
CG	: Center of Gravity
CM	: Center of Mass
CMOS	: Complementary Metal Oxide Semiconductor
CV	: Center of Volume
DCM	: Direction Cosine Matrix
DoF	: Degree of Freedom
GP	: Gaussian Process
GPS	: Global Positioning System
Hz	: Hertz
ICASE	: International Council of Associations for Science Education
IMU	: Inertial Measurement Unit
ITU	: İstanbul Technical University
Li-Po	: Lithium-Polymer
LQR	: Linear Quadratic Regulation
LTA	: Lighter Than Air
MEAM	: Mechatronics Education and Research Center
MPC	: Model Predictive Control
OPENCV	: Open Source Computer Vision Library
PWM	: Pulse Width Modulation
RC	: Remote Control
RF	: Radio Frequency
SICE	: The Society of Instrument and Control Engineers
SIFT	: Scale-Invariant Robust Features
SURF	: Speeded-Up Robust Features
VRML	: Virtual Reality Modeling Language
VS	: Visual Studio

LIST OF TABLES

	<u>Page</u>
Table 1.1 General Specifications of LZ 127 Graf Zeppelin [8]	5
Table 1.2 LZ 129 Hindenburg General Characteristics. [9]	6
Table 1.3 Properties of SkyCat-20 and SkyCat-220. [11]	7
Table 1.4 General Characteristics of Skyship 500 and Skyship 600. [12]	8
Table 1.5 Specifications of Eureka.	9
Table 1.6 AURORA I Airship AS800 Specifications. [13]	10
Table 1.7 General Information About the Airship LOTTE. [14]	11
Table 1.8 General Characteristics of USS Akron [8]	13
Table 1.9 General Characteristics of American Blimp MZ-3A. [15]	14
Table 3.1 Technical Specifications of Ground Stations.	21
Table 4.1 Dimensional Parameters of Blimp.	41

LIST OF FIGURES

	<u>Page</u>
Figure 1.1 Before Its First Flight, LZ 127 Graf Zeppelin is in Front of Its Hanger[8]	5
Figure 1.2 The Explosion Moment of LZ 129 Hindenburg. [9]	6
Figure 1.3 Goodyear Blimp "Spirit of America". [10].....	7
Figure 1.4 World SkyCat-20, Landing to Water. [11]	8
Figure 1.5 A Skyship 600 During Sightseeing. [12]	9
Figure 1.6 AURORA I AS800, During Its Test Flight. [13].....	11
Figure 1.7 Structural Components and Solar Panels of Airship LOTTE. [14]	12
Figure 1.8 MZ-3A Pending in Hanger. [15]	14
Figure 1.9 LEMV During Hovering. [16]	15
Figure 3.1 Schematic illustration of testbed, 45 points which images are grabbed from, and the blimp.	22
Figure 3.2 Remote control-Arduino board connection via an electronic circuit.....	23
Figure 3.3 Onboard hardware integration includes voltage regulator, sensor system, battery and default gondola of blimp.	23
Figure 3.4 Schematic workflow diagram of autonomous blimp system for indoor use.	25
Figure 4.1 Blimp body axes coordinate system showing the CV, CG and linear (u,v,w) and angular (p,q,r) velocities around the X, Y, and Z axes respectively.....	29
Figure 4.2 The location of the thrusters, the length, and the maximum diameter of the blimp.....	36
Figure 4.4 Longitudinal and Lateral models after LQR applied.	43
Figure 4.5 Basic Blimp Model in Simulink.	44
Figure 4.6 Blimp position visualization.	45
Figure 4.7 Blimp's movements in VRML.	45
Figure 5.1 Documentation style and explanation of dataset image entitles.	47
Figure 5.3 Relation between images and titles, which are derived from points that are images, grabbed from.	49
Figure 5.4 Image matching at Windows using SURF for greyscale and colored.	50
Figure 5.5 Keypoint results comparison between colored and greyscale images matching at Windows.	51
Figure 5.6 Image matching at Ubuntu using SURF for greyscale and colored.	51
Figure 5.7 Keypoint results comparison between colored and greyscale images matching at Ubuntu.....	52
Figure 5.8 Board layout and components of Razor 9 DoF IMU.....	52
Figure 5.9 IMU Data Structure.	53
Figure A.1 Wireless Camera Used in This Thesis.	62
Figure A.2 Arduino Mega 2560 Board Front View.	63
Figure A.3 XBee 2mW Chip Antenna - Series 2	64
Figure A.4 9 Degrees of Freedom - Razor IMU.	65

DESIGN AND CONTROL OF AN AUTONOMOUS BLIMP

SUMMARY

Until the Hindenburg disaster in 1937, airships have been used widely both military missions and commercial transportations, including transatlantic flights. Even if this incident had been a misfortune for the usage of airships, since 40 years, because of oil and energy crises, the popularity of airship has been increased.

In recent years autonomous vehicle applications are increasing gradually, besides unmanned aerial vehicles are the most used at these applications that are related to military and defense industry, commercial or academic projects.

Although airships are used in autonomous aerial vehicle projects, some disadvantages, such as bigger structure and lower maneuverability than other aerial vehicles, have made airships less preferred.

In this thesis, a blimp is used for an indoor robotics application. An indoor image dataset consisting 3090 images is recorded by grabbing image frames from a point which are specified before, by rotating the camera 15 degrees for all points for 3 different heights at Mechatronics Education and Research Center building. These images are used to determine the position of the blimp by matching with real time grabbed frames and then to fuse this data with position result which is obtained from IMU data.

Another onboard sensor that is integrated to the blimp is 9DoF IMU. IMU data is obtained wirelessly via an XBee and Arduino board. By double integrating IMU data the positions at three axes are obtained with errors because of drift and integration constants. Besides, IMU data is used to identify the blimp system in Matlab by validating experimental results with the basic dynamical model simulation results.

As a part of this thesis, some electronic circuits are designed and integrated to the blimp. A voltage regulator circuit is designed and used onboard with 3 different voltage outputs to blimp motors, camera and IMU-Arduino-XBee. At the same time, default RC of the blimp is connected to the serial port of ground station via Arduino board and an electronic circuit to increase and regulate the input voltages of remote controller is used. Control signals that are generated in the main program are sent via this part which includes an Arduino board to send PWM signal over 6 relays for direct&reverse rotations of motors to the motors.

In an addition to experimental parts of this thesis, a basic dynamical model is derived and simulated, and then results are validated with experimental results. The simulation is visualized in VRML. Next, the optimized simulation results are compared with experimental results.

The thesis project consists of four different parts that are modeling and simulation of the system, reading IMU data and calculating the position, grabbing images from wireless camera and matching grabbed images with images from dataset, comparing

simulated blimp positions with experimental results that are obtained after hardware integration.

OTONOM HAVA ARACI (ZEPLİN) TASARIMI VE KONTROLÜ

ÖZET

Hava araçları icat edildikleri günden beri kesintisiz bir şekilde askeri, ticari ve akademik çalışmalarda ilgi odağı olmuşlardır. Her ne kadar 1937 yılındaki Hindenburg faciası ve 2. Dünya Savaşı zeplin kullanımı açısından olumsuz bir kırılma noktası olsa da bu durum, zeplinlerin tarihte, hava araçları arasındaki önemini değiştirmez. Günümüzde yolcu taşımacılığı yapan, yolcu kapasitesi bakımından en büyük uçakların görevini, 1920-1940 yılları arasında, okyanus ötesi uçuşlarda zeplinlerin yaptığı gözönünde bulundurulursa zeplinlerin havacılık tarihindeki önemi daha da iyi anlaşılabilecektir.

Bilindiği gibi zeplinler yapıları itibarıyla diğer hava araçlarına göre enerji verimli araçlardır. Bu verimlilik genelde elips şeklinde olan dış yüzeylerinin içinde bulunan, havadan daha hafif taşıyıcı gaz yardımıyla belli ağırlıklardaki donanımları, kapasiteleri ölçüsünde taşımak için dışarıdan herhangi bir enerji girdisine ihtiyaç duymamalarından ileri gelir. Diğer hava araçları teknolojik gelişimleriyle kıyaslandığında geride kaldığı düşünülen zeplinelere olan ilgi, özellikle enerji darboğazlarının yaşandığı son 40 yılda kademeli olarak artış göstermektedir. Bu gelişmelerden en önemlisi zeplinlerde taşıyıcı gaz olarak patlayıcı özelliği bulunan Hidrojen gazı yerine, herhangi yanıcı ve patlayıcı özelliği olmayan bir asal gaz Helyum kullanılmasıdır. Bu ilginin karşılığı olarak, zeplinlere tekrar yolcu ve kargo taşımacılığında ticari görevler yüklenmekte, zeplinler askeri ve akademik projelerde, özellikle farklı tür ve boyutlardaki algılayıcı taşıyabilme özellikleri nedeniyle yer almaktadırlar.

Otonom (insansız) hava araçları, özellikle savunma sanayi ve akademik dünyanın son 15 yıldır en çok ilgi gösterdiği konuların başında gelmektedir. Bu araçlara entegre edilebilen yer istasyonları, koordineli bir şekilde çalışabildikleri otonom veya otonom olmayan yer araçları, farklı tür ve boyutlardaki sensörler bu alanı daha da karmaşık ve ilgi çeker hale getirmiştir. Bu alanda yapılan yenilikçi yaklaşımlara örnek olarak farklı otonom araçların birbirleriyle etkileşimli ve koordineli bir şekilde çalışabilmesi ve bir görevin farklı bölümlerini gerçekleştirebilmeleri gösterilebilir. En yaygın belirlenen görevler olarak gözetleme, takip, nesne tanıma ve uzaktan müdahale gibi örnekler gösterilebilir. Bunların yanında kullanıcı dostu ve kullanıcı etkileşimli sistemler de, yarı-otonom sistemlerde sıkça yer almaktadır. Yarı-otonom sistemler son dönemlerde sıkça otomotiv sektöründe kullanılmakta ve kullanıcılara yardımcı olacak teknolojiler şeklinde sunulmaktadır.

Bu tez kapsamında, iç-mekan kullanımına uygun boyutlardaki uzaktan kumandalı bir zeplin otonom hale getirilmesi, modellenmesi ve kontrolüyle ilgili deneysel, teorik ve benzetim çalışmaları yapılmıştır. Bu çalışmalarda İstanbul Teknik Üniversitesi, Mekatronik Eğitim ve Araştırma Merkezi binası iç ortamda yapılan deneylerde

kullanılmıştır. Bina içinde bir odanın belirlenen noktalarından görüntüler alınarak bir görüntü veri seti oluşturulmuştur. Ayrıca zeplinin standart haline ek olarak kablosuz kamera, 7.4 V gücünde bir Lityum-Polimer pil, dokuz serbestlik dereceli ataletsel ölçüm ünitesi, telemetrik haberleşme cihazı verici XBee, üzerinde Atmega 328 mikro-kontrolör ve dijital giriş-çıkışlar bulunan Arduino Uno marka bir elektronik kart ve özel olarak tasarlanmış farklı voltaj seviyelerinde güç sağlayabilen bir voltaj regülatörü gibi donanımlar eklenmiştir. Bunlara ek olarak yer istasyonunda oluşturulan sinyaller zeplin kumandası ile motorlara gönderilmektedir. Zeplin kumandası ile bilgisayar arasında Arduino Mega ve her üç motorun 2 yönünü de kontrol eden, anahtarlama devre elemanlarından oluşan bir elektronik devre bulunmaktadır. Kablosuz kameradan gerçek zamanlı görüntüler yer istasyonuna, kameranın alıcısı ile aktarılmaktadır. 9 serbestlik dereceli ataletsel ölçüm ünitesinden gelen veriler ise yer istasyonuna bağlı alıcı telemetrik haberleşme cihazı olan XBee ile sisteme aktarılmaktadır.

İç ortamdan toplanan görüntü verisi tezin ilk aşaması olarak düşünülebilir. Bu ilk aşamada bir veri toplama düzeneği ve zeplinin üzerinde bulunacak kablosuz kamera yardımıyla iç-ortamda 3 farklı yükseklikte, belirlenen her bir noktadan görüntü toplama düzeneği saat yönünde döndürülmek şartıyla 15'er derecelik açılarla, uygun bir noktadan 24 adet görüntü alınarak, görüntü veri-seti oluşturulmuştur. Görüntü veri-seti toplama işlemi Matlab programında yazılan bir kod ile gerçekleştirilmiştir. Görüntü veri-setindeki her bir görüntü 3-boyutlu eksene aktarılmış ve görüntülerin pozisyonlarının iç ortamda hangi noktada olduğu bilgisi yer istasyonunda çalışan yazılıma işlenmiştir.

Zeplinin matematik modeli, Matlab programında dinamik model üzerine gerekli doğrusallaştırmalar yapılarak 4'er serbestlik dereceli boylamsal ve yanal modeller halinde 2 ayrı şekilde oluşturulmuştur. Herhangi bir kontrol yöntemi uygulanmadan yapılan benzetim çalışmaları sonucu zeplin modelinin kararsız bir sistem olduğu sonucu elde edilmiştir. Kararsız zeplin sisteminin kontrol edilebilirliği incelenmiş ve kontrol edilebilirlik matrisinin rankı her iki model için de 4 olarak elde edilmiştir. Sistemin kontrol edilebilir olduğu sonucu elde edildikten sonra literatürde yer alan zeplin kontrolüyle ilgili çalışmalarda diğer alanlara göre kullanımı daha yaygın olan, tüm durum geribeslemeli bir kontrol yöntemi olan doğrusal-karesel regülasyon (LQR) yönteminin kullanımı uygun görülmüştür. Doğrusal-karesel regülasyon yönteminin kullanımı zeplin üzerinde bulunan ataletsel ölçüm ünitesinden, zeplinin tüm durumları alınabildiği için de kolayca uygulanabilecek durumdadır. Doğrusal-karesel regülasyon yöntemi uygulandıktan sonra Matlab programında benzetim çalışması yapılmış ve bu benzetim çalışması sonucu zeplin sisteminin kararlı hale geldiği sonucu elde edilmiştir. Ayrıca Simulink programında zeplin modeli 6 serbestlik dereceli hazır blok yardımıyla oluşturulmuş, buna ek olarak rastgele kontrol sinyalleri üreten bir başka alt sistem ile zepline itki kuvveti veren motorlara entegre edilmiştir ve VRML (sanal gerçeklik modelleme dili) arayüzü ile benzetim görsel hale getirilmiştir.

Zeplin üzerine yerleştirilen 9 serbestlik dereceli ataletsel ölçüm ünitesi, 3 eksenli ivmeölçer, 3 eksenli jiroskop ve manyetik alanölçen manyetometre(pusula) içermektedir. Ataletsel ölçüm ünitesinden elde edilen veri, bir adet Arduino Uno, verici XBee ve yer istasyonundaki alıcı XBee vasıtasıyla bilgisayara iletilmektedir. 9 serbestlik dereceli ataletsel ölçüm ünitesine, üzerindeki ATmega328 mikroişlemci vasıtasıyla kosinüs-yön matrisi (DCM) tabanlı AHRS kodu, ivme ve pusula

vektörlerine bağlı jiroskop sapmalarını düzeltecek şekilde düzenlenerek bir firmware olarak yüklenmiştir. Sonuç olarak yer istasyonuna ataletsel ölçüm ünitesi verisi, üç eksenindeki ivmeler, üç eksenindeki açılar, pusulanın yönü ve zaman değerlerini içeren, 8 elemanlı bir vektör şeklinde gönderilmektedir. Alınan ivme verilerinin alınarak üç eksenindeki hız değerleri ve bu hız değerlerinin integralleri alınarak da zeplin pozisyonu integral hesaplarında integral sabitlerinden doğan hatalarla da olsa hesaplanabilmektedir. İntegrasyondan ve sürtünmeden doğan hataları azaltmak için lokalizasyon ve navigasyon içeren akademik çalışmalarda sıkça kullanılan Kalman filtresi tasarımı ve uygulaması ileriki dönemde bu çalışmanın devamı olarak eklenecektir.

Zeplin üzerinde bulunan ve kablosuz görüntü aktarımı yapabilen kamera ile SURF algoritması kullanılarak görüntü işleme ve alınan görüntülerin veri setinde bulunan görüntülerle eşlenmesi uygulamaları yapılmaktadır. Görüntü işleme uygulaması olarak iç ortamda bulunan ve yer tahmininde önemi yüksek olan nesnelerin tanınması işlemi yapılmaktadır. Bu tanıma işleminin ardından gerçek zamanlı görüntüde algoritmalar tarafından bazı tanımlayıcı özellikler sayısal olarak belirlenmektedir ve bu özelliklerle veri setindeki görüntülerin tanımlayıcı özellikleri arasında bir uzaklık ilişkisi kurulmaktadır. Bu uzaklık ilişkisi 0 ile 100 arasındaki sayılara indirgenmiştir. Örneğin birbirinin aynı görüntüler arasındaki uzaklık 0 (sıfır) olacak şekilde indirgeme ve hesaplama işlemi yapılmıştır. Böylece görüntü eşleme algoritması ile zeplin konumu oda içinde başka bir sensör bilgisine ihtiyaç olmadan kabaca bulunabilmektedir. Alınan herhangi bir gerçek zamanlı görüntü ile veri-setinde bulunan bir görüntünün eşleştirilmesi çevrimi yer istasyonunda bulunan yazılımda yaklaşık Ubuntu 12.10 işletim sistemi ve Eclipse yazılım geliştirme ortamı kullanılan yazılımda renkli görüntülerle yapılan çalışmalar için 1600 milisaniye, siyah-beyaz görüntülerle yapılan çalışmalar için 1610 mili saniye, Microsoft Windows 7 işletim sistemi ve Microsoft Visual Studio 2010 yazılım geliştirme ortamı kullanılan yazılımda renkli görüntülerle yapılan çalışmalarda 6598 milisaniye, siyah-beyaz görüntülerle yapılan çalışmalar için ise 6560 milisaniye sürmektedir. Bu süre sisteme verilecek görevler düşünüldüğünde kabul edilebilir durumdadır.

Tez kapsamında deneysel olarak görüntü eşleme ve ataletsel ölçüm ünitesinden gelen verilerle yapılan işlemler sonucu ayrı ayrı bulunan pozisyonlar birleştirilerek daha güvenilir bir pozisyon verisi, her iki yer tayini işleminin hatalarının ortalaması alınarak elde edilmektedir. Elde edilen pozisyon verisine göre yer istasyonundan üretilen kontrol sinyalleri zeplin kumandası aracılığıyla zepline gönderilmektedir.

Son olarak Simulink ve VRML arayüzünde çalışan model ile deneysel sonuçlar karşılaştırılmakta ve matematiksel model doğrulanmaktadır. Bu işlem sayesinde gelecekte tez kapsamında oluşturulan zeplin matematiksel modeli geliştirilerek, model üzerinde yapılacak değişikliklerle benzetim çalışmalarında kullanılan modelin, deneysel sonuçlara daha yakın bir hareket yapması sağlanacaktır.

1. INTRODUCTION

Unmanned vehicle is a remotely controlled or autonomous vehicle which has no person on board to control or drive it. These devices should be capable of analyzing sensory data for the perception and based on the “perceived world” they make decisions to complete their tasks.

In this work, the aim is to design and control of an autonomous aerial vehicle, blimp/, as a part of unmanned aerial vehicle fleet project at Mechatronics Education and Research Center, Istanbul Technical University. The focus of the work is information fusion using multiple sensors located conveniently on the blimp gondola. In addition, using autonomous blimp, the semantic mapping of indoor environments is achieved for intelligent task planning and execution.

1.1 Use of Unmanned Airship in Research

Blimp is a non-rigid airship and it differs from a big-scale airship because it does not have any rigid structure to help the airbag maintain its shape.

Blimps have a wide range of potential applications. Although usage areas had decreased sharply after Hindenburg disaster (1937) and World War II, in recent years blimps became popular in commercial applications, defense industry and reflected in academic research, especially focusing on navigation, perception and control abilities of blimps. Blimps are mostly used as advertising environments commercially. They are also used in TV camera platforms and as a means of transportation both touristic and cargo. In addition, defense industry uses blimps for surveillance, exploration, monitoring, and transportation.

Blimps could also be used in examining the flock behavior of co-operating animals. For example, a blimp has been used to investigate and reconstruct the principles underlying biological navigations systems such as insect flight [1]. Similarly, another study investigates the navigation behavior of bees for the purpose of long-range goal-

directed navigation in 3D environment by using a biological model of motion detector [2].

One of the researches investigate emergency management using unmanned airships investigates emergency management. The research reports several applications on monitoring natural disasters, such as flood, earthquakes, fire, drought, severe storms etc., search and rescue, and communication [3]. Blimps can also be used in unknown environments conveniently to search for the indication of life. In [4], the researchers discuss an approach to field-testing methods relevant to three scientific thrusts in the detection of life and pre-biotic organics on other worlds by deploying a mobile organic laboratory on Earth to demonstrate the required techniques. Additionally, an R&D research introduces results for aerial robot systems for urban search and rescue (USAR). In this research, different types of aerial vehicles are used and these vehicles are combined to obtain quick and continuous service for disaster information. Autonomous helicopters collect disaster situation data from the sky for first decision making. Then, a blimp and a cable-driven mobile robot survey victims by detecting faint signs of life. A captive balloon system monitors the area and relays wireless communication among working teams on the ground [5]. In fact, one of the most promising applications of an unmanned airship is environmental observation system for large-scale gas pipeline network to monitor gas leakage and to cope with other dangerous situations [6]. A similar use of airships is in the field of automatic forest fire monitoring and measurement. In [7], such an application contains several aerial vehicles working co-operatively to detect fires by sensors, and a central station to guide the aerial vehicles (an airship, two helicopters) for forest fire monitoring.

1.2 Components of an Autonomous Blimp

Autonomous blimps have electronic, mechanical and electromechanical components for perception, navigation and motion control. These components may vary according to the different uses of blimps. The size of the blimp is usually the most important specification, which may change the equipment variety because of the available payload due to buoyancy limits and efficiency.

1.2.1 Envelope

The envelope is the outer surface of the blimp and it gives the shape to the blimp. It usually has a shape of ellipsoid but also it could have different shapes considering aerodynamic conditions, such as cigar, spherical etc. The material of the envelope varies depending on the usage area of the blimp. Usually blimps have helium-proof envelope with Mylar foil. In some cases, the material of the envelope may be latex or polyurethane. In some applications calendared ripstop Nylon or calendared, Silicone-coated, and laminate materials, which are usually used to as parachute materials, has been used as envelope material. The ideal properties which are desired from envelope materials:

- High strength to determine the maximum possible size of the envelope,
- High ‘strength to weight ratio’ to minimize the weight of the envelope,
- Resistance against to the environmental properties,
- Low permeability to minimize the lifting gas loss,

1.2.2 Gondola

Gondola is attached to envelope underneath the airship. Electric motors, propellers, cables, battery, and some other components as sensors, microprocessor and microcontroller are carried by gondola. Depending on the requirements, types, specifications, numbers and sizes of components may vary.

1.2.3 Fins

Fins are usually located at the tail of the blimp. The number of fins may vary according to the required navigation and stability of the blimp since they are used to help to rotate and provide a better stabilization to the blimp. In motion control applications, the fins are actuated by servomotors for active control of orientation.

1.2.4 Remote control device

Autonomous blimps are usually controlled by embedded systems but in some cases, the device which processes the data and sends control signals to motors can be an external and ground-based system that controls the blimp remotely.

1.3 Previous Projects and Their Application Areas

As is well known, airships were as popular as aircrafts/airplanes until the Hindenburg Disaster (1937) and World War II. These two incidents decreased the use of airships. From 1970s the researches about airships has increased and so many commercial, military and academic airships have been developed the evolutionary development was in since 1970s has been the change of lifting gas from Hydrogen to Helium. In addition, some extra tasks has been added to airships and today the use of airships are not just carrying people/military personnel but also executing autonomous missions such as surveillance, monitoring, transporting etc. Developments in autonomous blimp research are not limited just to the co-operative missions. There have been also unconventional structural designs and some of them are covered in this section.

Although the airship projects are related to each other to some extent, they can be classified in three groups such that commercial, academic and military projects.

1.3.1 Commercial airship projects

LZ 127 Graf Zeppelin is the most stunning work in the airship history because it is a production, which is identified by its trademark. Many people would understand airship if anybody tells about Zeppelins. LZ 127 Graf Zeppelin was built in Germany and carried passengers during its operating life. Lifting gas of airship was Hydrogen and rigid airship has operated from 1928 to 1937 commercially. The name Graf Zeppelin comes from Ferdinand von Zeppelin who was the father of airships and was a Graf/Count in the German Nobility. In Figure 1.1, LZ 127 Graf Zeppelin is shown while it was leaving the storage area in Friedrichshaven for its First Flight on September 18, 1928.



Figure 1.1 Before Its First Flight, LZ 127 Graf Zeppelin is in Front of Its Hanger. [8]

During its operating life, the airship made 590 flights covering more than a million miles (1.6 million km). It was designed to be operated by a crew of 36 officers and men. In Table 1.1, the general specifications of LZ 127 Graf Zeppelin is given.

Table 1.1 General Specifications of LZ 127 Graf Zeppelin [8]

	LZ 127 Graf Zeppelin
Crew	40
Capacity	20 passengers
Length	236.53 m
Diameter	30.48 m
Volume	105000 m ³
Useful Lift	60000 kg
Powerplant	5xMaybach Engines, 410 kW (550 hp) each
Maximum Speed	128 km/h

There is another German-built airship as famous as Zeppelin, LZ 129 Hindenburg, but its fame comes from an accident and its enormous dimensions. LZ 129 Hindenburg was a large commercial passenger-carrying rigid airship, also the longest class of flying machine and the largest airship by envelope volume. Table 1.2 demonstrates the specifications of the airship.

Table 1.2 LZ 129 Hindenburg General Characteristics. [9]

	LZ 127 Graf Zeppelin
Crew	40 to 61
Capacity	50-72 passengers
Length	245 m
Diameter	41.18 m
Volume	200000 m ³
Powerplant	4xDaimler-Benz diesel engines, 890 kW (1200 hp) each
Maximum Speed	135 km/h

With its dimensions LZ 129 Hindenburg is the biggest flying machine ever built, the moment of the explosion during its landing is shown in Figure 1.2.

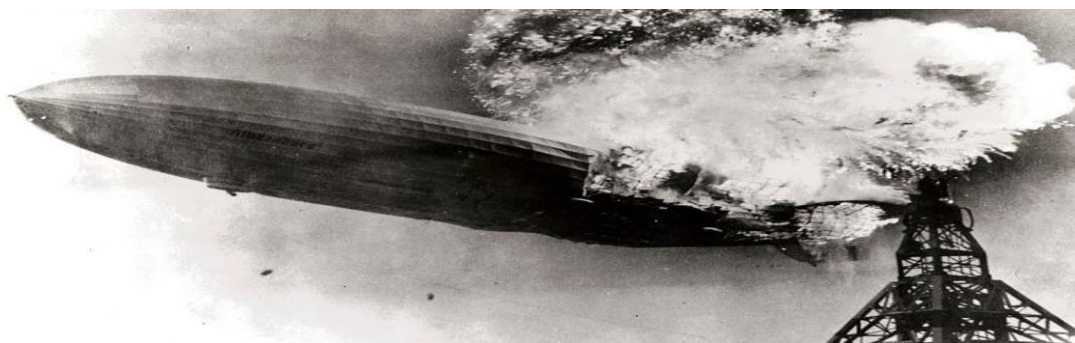


Figure 1.2 The Explosion Moment of LZ 129 Hindenburg. [9]

After first North American transatlantic flight, on 6 May 1937 14 months later from its first take-off, LZ 129 Hindenburg destroyed by fire during landing and 36 people died in the accident. The main reason of the accident was the explosion of lifting gas, hydrogen. The choice of hydrogen is just about the prevalence and the lighter weight of hydrogen comparing with helium.

Since 1925, Goodyear (tire and rubber company) has produced many blimps. The Goodyear Blimp may be the most popular advertisement blimp in the world. In addition to advertising purposes, Goodyear built rigid airships for the US Navy in the 1930s. Furthermore, in the 1940s and 1950s, Goodyear has manufactured a series of large surveillance airships used to protect merchant fleets along the coast. Today Goodyear fleet consists of four airships, these are; Spirit of America, Spirit of Goodyear, Spirit of Innovation and Navigator. In Figure 1.3, Spirit of America named Goodyear blimp is landing and crew are working for a safety landing. This

blimp purposed to transport people as it can be seen from the gondola part of the blimp there are seats.



Figure 1.3 Goodyear Blimp "Spirit of America". [10]

A company, named World SkyCat Ltd., has developed two airships, SkyCat-20 and SkyCat-220 and on 23rd July 2000 their one-sixth fully working model made its first flight at Cardington, UK. This company can modulate these airships for several specific missions such as surveillance and border control, emergency relief, firefighting, luxury tourism, mass passenger transport, passenger and car transport, natural gas transport, pipeline transport, heavy-lift cargo, etc. and the names of the airships may vary according to their mission. In Table 1.3, there are parametric values of SkyCat-20 and SkyCat-220.

Table 1.3 Properties of SkyCat-20 and SkyCat-220. [11]

Overall Dimensions	SkyCat-20	SkyCat-220
Length	81 m	185 m
Height	24.1 m	47 m
Width	41 m	77.3 m
Payload Module		
Length	25.5 m	64 m
Height	2.6 m	4.8 m
Width	3.5 m	7.8 m
Payload		
Standard STOL Mode	20 tons	220 tons
Hover/VTOL Mode	14.5 tons	160 tons
Range		
Max. Payload at Cruise	2400 n.miles	3225 n.miles
Speed		
Cruise	75 kts	80 kts
Sprint	85 kts	95 kts

These airships are hybrid because of lifting is made by the help of lifting gas (Helium) and maneuvers are made by thrusters and the energy sources of the thrusters may be electricity or fuel, in an addition SkyCat-20 has vertical take-off and landing capacity as can be seen in Figure 1.4.



Figure 1.4 World SkyCat-20, Landing to Water. [11]

In the 1980s, Airship Industries was an active company, which manufactured modern non-rigid airships (blimps). There are two remarkable blimps, which are manufactured completely by this company, Skyship 500 and the larger model Skyship 600. The first flight of Skyship 500 was on 28 September 1981, taken place from Cardington. The first Skyship 600 flew from Cardington on 6 March 1984. The general specifications of Skyship 500 and Skyship 600 are given in Table 1.4.

Table 1.4 General Characteristics of Skyship 500 and Skyship 600. [12]

	Skyship 500	Skyship 600
Length	52 m	66 m
Diameter	14 m	22 m
Speed	40 km/h	40 km/h
Volume	5153.666 m ³	7600 m ³

Both Skyship 500 and Skyship 600 were first used in advertising and camera platform, and then tested in touristic transportation in a number of cities worldwide such as Paris, Toronto, Montreal, Sydney, Brisbane, Melbourne, San Francisco, London and Zurich and military services. As can be seen in Figure 1.5, there is a Skyship 600 which is carrying peoples during a sightseeing in 2006 at Swiss Alps [12].



Figure 1.5 A Skyship 600 During Sightseeing. [12]

The Zeppelin Company is a manufacturing company settled in Friedrichshafen, Germany and it builds airships for flying passengers, scientist or TV cameras. These airships are called Eureka. Information about airships, which are manufactured by Zeppelin and have rigid structured bodies, is shown in Table 1.5. Currently, this airship is used regularly on flight-seeing tours and custom charters above the San Francisco Bay Area, San Diego and Greater Los Angeles.

Table 1.5 Specifications of Eureka.

Dimensions		Cabin	
Length	75 m	Number of Seats	2 crew + up to 12 passengers
Width	19.5 m	Restroom	Yes
Height	17.4 m	Volume	26 m ³
Envelope Volume	8400 m ³	Length	10.7 m
Ballonet Volume	2200 m ³	Parameter	
Envelope Surface Area	2630 m ³	Mission Duration/Endurance	High (>12 h)
Mass		Crew/Passenger Comfort	High (noise < 20dB)
Maximum take-off Weight	8040 kg	Noise/Other Emissions	Low
Useful Load	1950 kg	Fuel Consumption	Low (50kg/h)
Performance		Minimum Speed	Hover Capability
3xLycoming IO-360	150 kW(200 hp)	VTOL Capacity	Yes
Maximum Level Flight Speed	125 km/h	Very Low Altitude Mission	Yes
Range	900 km		
Ceiling	2850 m		
Maximum Endurance	~24 h		

1.3.2 Academic airship projects

Generally, academic airship projects support commercial or military airship projects. Therefore, the academic airship projects, which are independent from commercial or military airship projects, are covered in this section.

Autonomous airships provide interesting and attractive research areas to academic world because of difficulty of controlling big volumes of airships with great dimensions such as an aircraft or bigger under different weather conditions, energy efficient movements that are based on lifting gas effects and possible use of solar panels and potential of using sensors separately or together. In particular, fluid mechanics, mechanical design and manufacturing methods of blimps are important areas that require further research and development.

One of the most popular academic works about airships is *Project AURORA* (Autonomous Unmanned Remote Monitoring Robotic Airship) [13]. This research mainly discusses developing the fundamental technology of an autonomous airship which completely includes the subjects such that landing and ground hardware and software infrastructures; airship dynamic modeling and simulation; control and guidance methods; visual servoing strategies; robotic air-ground cooperation; dynamic target recognition; and hybrid airship robotic software architecture. Considering developing technology objective, Project AURORA was conceived as a multi-phase project; at this section of my thesis the first phase of this project, AS800, is covered. AURORA I, AS800, is a non-rigid airship and Table 1.6 demonstrates the parametric specifications of this model.

Table 1.6 AURORA I Airship AS800 Specifications. [13]

	AURORA I Airship AS800
Length	10.5 m
Diameter	3 m
Volume	34 m ³
Payload	10 kg
Maximum Airspeed	50 km/h

In Figure 1.6 AURORA I AS800 is demonstrated. AS800 has three main parts: the onboard components, the ground station and the communication system elements. The onboard station is composed of processors, sensors, actuators and part of

communication system. Sensor unit contains a GPS receiver, an IMU, a wind sensor and cameras. Additionally, there are propeller speed sensors, control surface and vectoring position sensors, engine temperature and fuel and battery level sensors.



Figure 1.6 AURORA I AS800, During Its Test Flight. [13]

The ground station includes a portable computer, human-machine interface (HMI) for visualizing and interaction mechanism between the operator and the airship onboard system. The communication system is based on two radio links which the first one transmits analog video imagery from the airship to ground station and the second one transmits digital sensor telemetry and command data between both stations.

A second important example project is LOTTE which is a remotely controlled airship, built by an Airship Research Group called FOGL (ForscherGruppe Luftschifftechnologie) at the University of Stuttgart, Germany [14]. The aim of this research group, FOGL, is to use LOTTE as a test-bed to get a better understanding and wide information about LTA technologies. In Table 1.7, there is some general information about LOTTE.

Table 1.7 General Information About the Airship LOTTE. [14]

	LOTTE
Length	16 m
Maximum Diameter	4 m
Total Envelope Volume	109 m ³
Installed Ballonet Volume	19 m ³
Maximum Payload	15 kg
Area of Solar Cells	Max. 4.8 m ²
Solar Power	Max. 720 W
Maximum Flight Altitude	Approximately 1000 m
Maximum Flight Speed	45 km/h

LOTTE is not only special because it is a solar powered airship, but it is special because it is equipped with sensors such that an IMU, a GPS, ultrasonic anemometer, compass, pressure gauge and thermometer to have the full perception capabilities. In the project the modeling was given a significant importance. To obtain a reliable theoretical model, experimental investigations and in-flight tests (medium wind tunnel, gust wind tunnel and large water tunnel) have been applied to a scaled model of the airship. The remotely controlled airship has an electric engine which is powered by battery/cell units. In order to recharge these batteries there are solar panels on the airship. There are 12 modules of solar cells and each of them has 0.4 m² area to collect solar energy, the power consumption of the electric engine is 1500 W. In Figure 1.7, the structural components and the solar panels of the airship LOTTE are demonstrated.

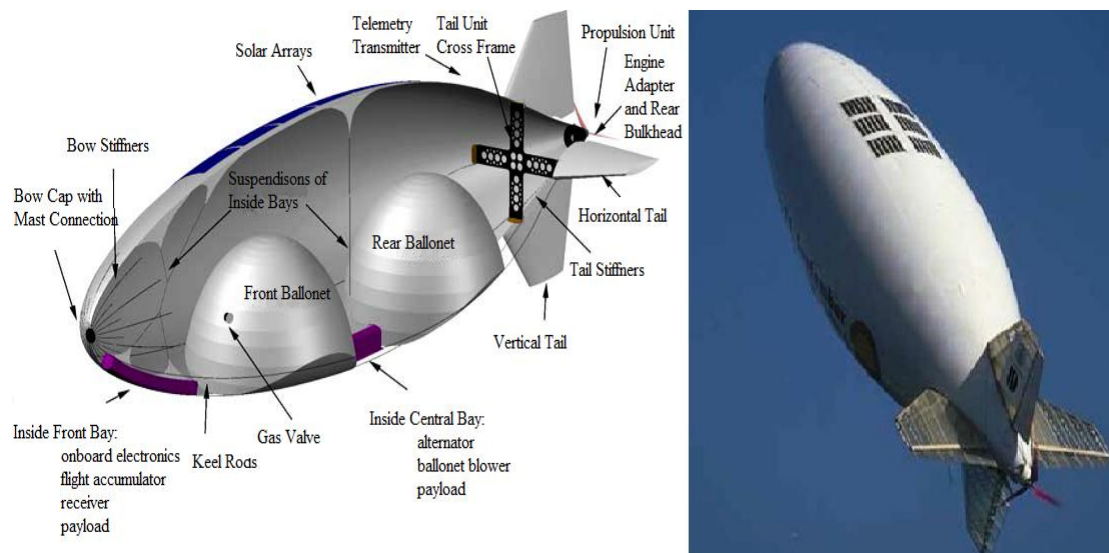


Figure 1.7 Structural Components and Solar Panels of Airship LOTTE. [14]

1.3.3 Military airship projects

Even if major numbers of military airship projects are developed by universities, there are some companies which builds airships for military applications. In this section, the significant projects will be discussed considering historical development.

USS Akron is one of the biggest airship projects in the history that is designed and operated for US Navy. This is a rigid airship and gets involved in memories by a weather-related accident with the loss of 73 of the 76 crew and passengers on board. During its operating life (1931-1933), USS Akron also served as a flying aircraft carrier. USS Macon which is the sister of USS Akron is operated by US Navy for

scouting and carrying aircrafts and the dimensions were same. Both USS Macon and USS Akron are the largest helium filled flying objects ever. In Table 1.8 general characteristics of these airships are shown.

Table 1.8 General Characteristics of USS Akron [8]

	USS Akron
Length	239 m
Diameter	40.4 m
Height	46.5 m
Weight	100 tons
Volume	180000 m ³
Powerplant	8xMaybach VL-2 gasoline 420 kW (560 hp) engines
Speed	Cruising: 93 km/h Maximum: 133 km/h
Range	19590 km
Useful Load	83000 kg
Capacity	89 officers and men
Aircraft Capacity	4xF9C Sparrowhawk biplane fighters
Armament	7xmachine guns

Furthermore, starting from 1910s until 1960s, US Navy had used so many blimps. Their intended uses are mostly about coastal military patrol and rescue, additionally aircraft carrying was one of their activities, too. These blimps are called some alphabetic letter classes such that; B Class, C Class, D Class, E Class, F Class, G Class, H Class, J Class, K Class, L Class, M Class, N Class. These all classes are manufactured by Goodyear-Zeppelin Corporation and/or Goodyear Aircraft Corporation.

Even though YEZ-2A is a military project that Airship Industries could not complete which was originally intended for the US Navy, there is a PhD thesis completed and presented in 1990, at the Cranfield Institute of Technology, College of Aeronautics, Aerodynamics Department, by S. B. V. Gomes. The bankruptcy of Airship Industries is the main reason of this unsuccessful project. The PhD thesis covers construction of

flight dynamics model for computer simulation of YEZ-2A by obtaining aerodynamic data from its prototype.

One of recent blimp projects that are manufactured by military purposes for the US Navy is named American Blimp MZ-3. Being different from previous projects, American Blimp Corporation manufactured this blimp and its modified model of commercial type A-170 series. MZ-3A operates as an advanced flying laboratory used to evaluate affordable sensor payloads, the development of new lighter-than-air (LTA) technologies and general flight support for other related R&D projects for US Navy. In Table 1.9, general information of MZ-3A is shown.

Table 1.9 General Characteristics of American Blimp MZ-3A. [15]

	American Blimp MZ-3
Length	54 m
Volume	4800 m ³
Speed	Max. 93 km/h
Powerplant	2x130 kW (180 hp, Lycoming IO-360)
Capacity	1 pilot, 9 passengers
Ceiling	2900 m
Range	>350 NM

Additionally, MZ-3A is equipped with various sensors to support of technology development for command, control, communications, computers, intelligence, surveillance and reconnaissance (C4ISR) and R&D purposes. In Figure 1.8, MZ-3A is shown in hangar.



Figure 1.8 MZ-3A Pending in Hangar. [15]

The Long Endurance Multi-Intelligence Vehicle (LEMV) is a hybrid airship which is recently undergoing flight tests developed by Northrop Grumman and Hybrid Air

Vehicles for the US Army to provide intelligence, surveillance and reconnaissance support for ground troops. Determined project cost will be \$ 154 million. The unconventional design seems like twin envelopes attached to each other, LEMV is shown in Figure 1.9.



Figure 1.9 LEMV During Hovering. [16]

1.4 Aim and Objective of This Thesis

The thesis focuses on integration of sensors, data fusion and motion control of the blimp to achieve several tasks with semantic mapping. Because of the blimp size and specifications a control system cannot be embedded on it, the gondola carries only a wireless camera and an IMU.

A mathematical model is created in Simulink and connected to VRML for tracking the blimp motion in ground station and comparing it to the real data taken from on-board sensors. The remote control is connected to the PC. The control of blimp is achieved remotely using the RC link between PC and blimp. A wireless camera is attached to the envelope to identify the location of the blimp which compares the frames with an image dataset which is collected before to construct the map/model of indoor environment. Furthermore, IMU is used to identify the location and the outputs are compared with camera localization results.

2. LITERATURE REVIEW

Autonomous airships are used generally for monitoring and surveillance in research projects. However, a major part of research also focuses on improving the perception and cognition capabilities of airships as a part of intelligent cooperative UAV team.

2.1 Computer Vision in Blimp

In order to have good perception capabilities, autonomous blimps often carry a on-board camera employing several computer vision algorithms such as monitoring, surveillance, detection and tracking depending on the task of the robot itself or the UAV team.

In [10], for detecting and tracking of predefined features, an autonomous blimp, which circles around a specified target, is designed and for this purpose, an extension of Lucas-Kanade algorithm is used. However, it was seen that Lucas-Kanade algorithm does not have efficient performance for rotation and scaling of images as mentioned at the same work.

Sometimes autonomous airships are used as a part of the team as they are employed in [18]. They used autonomous blimp for forest monitoring, being a member of a UAV team that contains of 2 helicopters, 1 blimp and a central station. In their work they define the shape, position of the fire front, its rate of spread (how this front evolves with time) and the maximum height of the flames as the most important parameters for the fire detection. The scenario, which is described in the paper, is tested with these vehicles and the mission of the blimp is to patrol the area to look for fire spots. If the fire is detected, one of the helicopters is sent to the fire zone to confirm the situation and localize the fire precisely. After confirmation of the fire, the team generates tasks to take and process pictures of the event from the correct viewpoints at the same time. In that work, two helicopters are just used to confirm and take more pictures from the fire area. Instead of using helicopters for the missions such as confirmation and taking pictures, the airship might have been a

better solution having the accuracy and quality of the localization. Additionally this would prevent the delay which occurs during the confirmation. In this way, some other missions, such as urgent response to the fire or rescuing people from the dangerous area, could be assigned to two helicopters suiting to their agility in motion.

In a different approach, a computer vision based navigation application can be used to estimate the states of an airship from an image database of visual beacons in the image dataset created before and the geometric specifications of the objects that are known [19].

The vision system is not only employed for navigation but it can be also used for tracking and image projection. In [20], an external system tracks the blimp and estimates its states during the flight of the blimp that the blimp flies along a given spatial path to follow a wall. This system is just suitable for the areas where the blimp is clearly seen by the projection system. For this reason, the system is inconvenient in some cases such as outdoor environments since an object or may interfere between the projection system and blimp.

Not all the UAV teams should have a ground station. The blimps can also be designed as a search -rescue or surveillance system operating autonomously. The blimp can carry many sensors such as IMU, GPS, stereo camera and the processor; therefore, without using any external station the blimp can track and estimate its own states. [20]. A study framework only uses one camera to control the blimp and then reference trajectory is generated from images [21] and a different work uses an object detection and tracking algorithm for visual tracking [22].

2.2 Navigation, Localization and Mapping

Although navigation, localization and mapping applications of autonomous blimps can be performed by using cameras and computer vision algorithms, usually IMU, GPS, sonar sensor and similar sensors are used in addition to make the sensing capability better. The necessity of using other sensors comes from obtaining more robust measurement results such as the localization of the blimp or the objects around the blimp, and dynamic states of the blimp such as position, velocity, and orientation to be used in control algorithm.

One of the studies using autonomous blimps which works connected to a ground station, used to acquire information on position and status of the victims in a disaster scenario in the urban area over a rescue communicator recording sound for five seconds in every minute [24]. The experimental outdoor results show that the height of the blimp is the most important factor in obtaining the voice data of victims. This means in order to obtain healthy voice data, the height of the blimp has to be limited according to the capturing capability of the voice sensor.

Similarly, another research uses ultrasound sensors to determine location of an indoor blimp using particle filter [20]. Sonar sensors, which are used at that research, have 6 m measurement range and this capability may limit the area in which the blimp can be accurately localized.

As a mapping application, we may give the example in [25]. Here, they use stereovision, GPS, fluxgate compass and wind sensor. They obtain experimental mapping of the area using optical odometry. As a result of using optical odometry, the traveled distance measurements have errors and accumulation of these errors caused the drift.

2.3 Control and Co-Operative Control

Many control algorithms and techniques are applied to control the autonomous blimps either individually or as a member of an UAV team. The most popular two of these algorithms are predictive control and adaptive control methods. Usually, in order to obtain the dynamic model. A Gaussian process (GP) is used to identify the systems or correct the measurement results. In some cases, two or more algorithms or control techniques can be used in combination.

Usually GP is extended or synthesized with extended Kalman filter (EKF), unscented Kalman filter (UKF) [26], adaptive value iteration [27], Bayesian filtering [22] and reinforcement learning algorithms [29], [30]. Learning and adaptive algorithms do not require pre-defined dynamical models of the blimps. In [27], the approach of applying Monte Carlo reinforcement learning utilizing Gaussian process is used. The advantage of this approach is that the algorithm is able to learn the dynamics of the blimp.

Predictive control techniques are widely used on autonomous blimps. Model predictive control (MPC) and state-predictive control are compared in [31], and in [32] constrained model predictive control (MPC) is studied. Since we can include input and output constraints distinctly in MPC, this method will be considered in this thesis.

Co-operative control of autonomous blimps is a complex problem. This type of control method is inspired from animal aggregations such as fish, bird flocks, and deer herds. These animal teams coordinate their collective motions to perform useful tasks and understanding their behavior may help us build mobile autonomous agents such as AUV (autonomous underwater vehicles) schools, UAV flocks that might be instructed to cooperate in a similar manner. In these teams, the missions of all members can be different or identical.

In [27] two unique experimental setups are introduced to demonstrate the cooperative control of teams of UAVs. The first experimental setup uses eight rovers and four blimps operated indoors to emulate a team of heterogeneous vehicles performing a combined reconnaissance and strike mission. The second experimental setup uses eight small aircraft that are flown autonomously using a commercially available autopilot. This combination of experimental setups provides platforms for both advanced research and realistic demonstrations. Numerous trajectory optimization and team coordination algorithms have been developed to execute these UAV missions. Another research project studies a team which consists of a ground mobile robot and a blimp. The behavior of the vehicles belong to this team for target detection is as follows: The autonomous ground robot moves along a pre-planned path, and, at periodic waypoints along that path, it stops and conducts a 360° visual sweep of the area with the camera, looking for targets. The human-controlled UAV, being less-precisely controllable than the autonomous ground robot, follows its predefined path and, as deemed necessary by the human pilot, rotates and changes altitude to visually scan the surrounding area for targets. Upon detecting a target, the robot or UAV then localizes the target to the best of its ability [28].

3. PROBLEM DEFINITION

This work includes four different parts, which are modeling the blimp, image processing, inertial navigation, data fusion and integrating the hardware. In this section the purpose of this thesis and workflow of all parts of the system are covered. Then, in section 4, detailed information about modeling a 6 DoF blimp and the model of blimp used in this thesis is detailed. In section 5, experimental results for navigation that are obtained from blimp via camera and IMU are given. Finally in section 6, depending on our results the reviews are given.

3.1 Purpose of This Thesis

The main purpose of this thesis is to fuse two different types of data which are obtained from IMU and camera wirelessly to estimate the position of an indoor blimp in a known environment by the whole system.

As being a ground station, one laptop computer and one desktop computer are used alternatively to prevent an unexpected situation and both ground stations are equipped with necessary software to operate for the system in this thesis. In Table 3.1, technical specifications of these computers are given, besides technical specifications of other hardware that are wireless camera, Arduino 2560 Mega and Arduino Uno boards, XBee, Razor 9 DoF IMU are given in Appendix A.

Table 3.1 Technical Specifications of Ground Stations.

	Desktop Computer	Laptop Computer
Operating System	Windows 7 Professional (64-bit)	Windows 8 (64-bit) (Alternatively Ubuntu 12.10 is installed)
Processor	Intel® Core™ i7	Intel® Core™ i7-33632QM
CPU	@ 2.80 GHz	@2.20 GHz
Installed Memory (Ram)	6,00 GB	8,00 GB
Hard Disk Capacity	400 GB	750 GB

Therefore, determining the exact position of indoor blimp is not covered in this thesis but by fusing IMU and camera data the position is estimated such as in a cube which is as shown in Figure 3.1. Additionally, in Figure 3.1, an illustration of all 45 points, testbed and a flying blimp is shown.

At the beginning in order to apply visual algorithms using SURF method, an image dataset consisting 3090 images of indoor environment is created by grabbing images from all appropriate 45 points by rotating the testbed, which is fitted from a representation rack for this work, fifteen degrees for all images for three different heights. The code of grabbing images from indoor environment is written in Matlab and given in Appendix B.

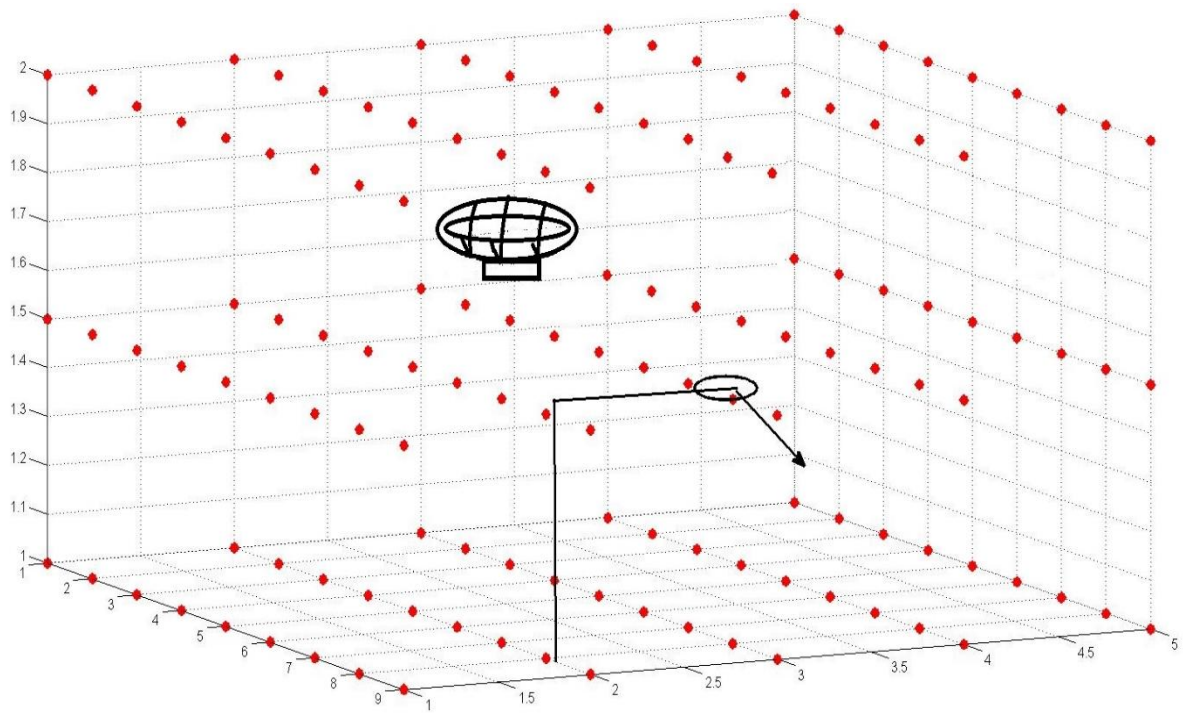


Figure 3.1 Schematic illustration of testbed, 45 points which images are grabbed from, and the blimp.

3.2 Hardware Integration

In order to send control signals to motors of blimp, the default remote control is connected to an Arduino board via an electronic circuit. Arduino board PWM outputs give 3.3V but remote control of blimp needs 9V, therefore our electronic circuit includes relays to regulate and increase the voltage coming from Arduino board to desired voltage level for remote control.

This integration provides to send signals directly from control software via serial port of ground station. In Figure 3.2 this connection is shown.



Figure 3.2 Remote control-Arduino board connection via an electronic circuit.

In this work, an onboard sensor system is integrated to blimp. This system includes a 9 DoF IMU sending data via an XBee connected to an Arduino board, a wireless camera, a 2-cell Li-Po 7.4V, 800 mAh battery, besides blimp's default equipment is connected to this system to be given power. Additionally, a voltage regulator is designed and produced at MEAM to feed this system at three different voltage levels that they are 3V for motors of blimp, 6V for Arduino board, 7.4V for camera. In Figure 3.3, onboard hardware integration is shown.

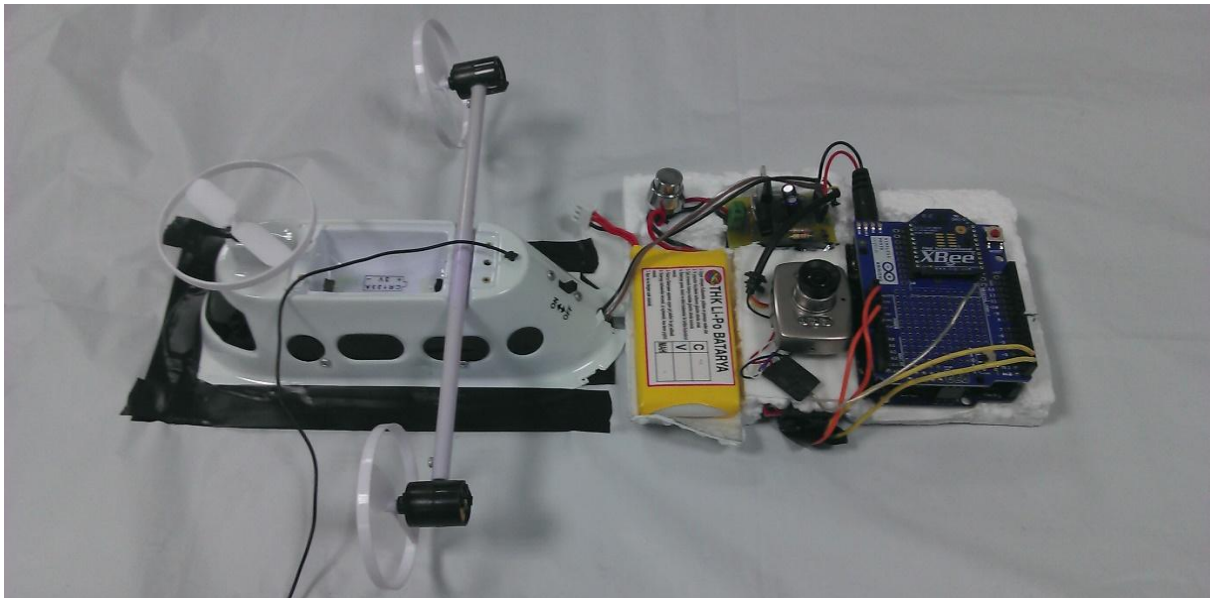


Figure 3.3 Onboard hardware integration includes voltage regulator, sensor system, battery and default gondola of blimp.

3.3 The Schematic Workflow of Whole System

In order to show and explain the work step-by-step covered in this thesis a schematic workflow diagram is created and shown in Figure 3.4. Initially the states of the system are known, and then blimp sends both IMU and grabbed image data to ground station separately. Images are matched by dataset images and this gives identified region and identified object data. On the other hand, by integrating IMU data we obtain velocities at 3 axes then filtering (in order to reduce the effect of errors coming from integration constants), by double-integrating IMU data we obtain estimated position results for 3 axes. After we have 2 different data source reporting on positions, we fuse them to locate blimp in one of cubes which are created virtually by drawing lines between points that are shown in Figure 3.1. When we have results that are obtained by fusing IMU and camera data, we create control signals by comparing experimental results with our model's movements running at the same time and then we send control signals to blimp via remote control. Thus, a self-contained system which can sense the environment and perform localization is obtained. In future work, this system will be used to design an indoor surveillance application and the autonomous blimp will be a part of co-operative robot team which can simulate search-rescue scenarios. In future studies, the robot may perform semantic mapping functions as well, hence the project could be extended to have more artificial intelligence applications.

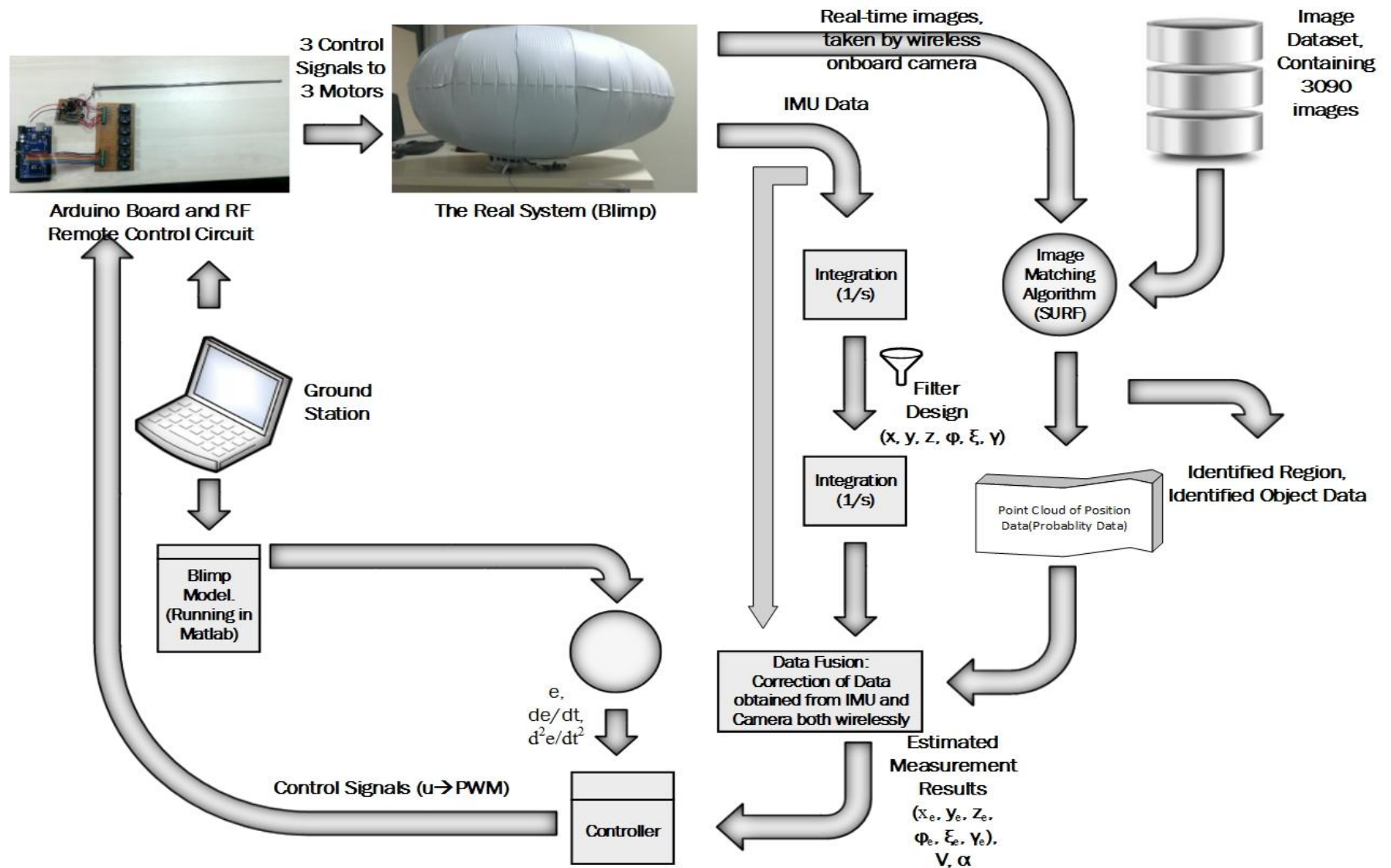


Figure 3.4 Schematic workflow diagram of autonomous blimp system for indoor use.

4. MATHEMATICAL MODELING

Models systematically help humans to understand the functionalities of systems at different areas. In mathematical modeling, these systems are translated into the language of mathematics for our understanding. In another words, mathematical models are very useful to test different situations and parameters those effect directly or indirectly to the systems. In this thesis mathematical model is used to compare real-system with a mathematical model and analyze the differences. Thus, mathematical that is used in this thesis is used to obtain the results of different control methods to stabilize the system.

There are many parameters which effect on the result of the modeled system output and by mathematical modeling some parameters are not included in the model and some parameters have smaller effect on the result than they do in reality. These assumptions made it easier and faster that to have a better adoption and then many simulations can be run to observe the limits of the system model. The meaning of adapting mathematical models to computers through a software is to transform a physical, financial or any modeled situation with all known and considered parameters to a language which software and computer understand. Moreover, this is a very useful and easy way to analyze the simulation results depending on parameters.

This thesis includes kinematic modeling and dynamical modeling parts of the system. The blimp or airship, which is considered for this thesis, is used in an indoor environment; therefore, its model consists of three translational and three rotational degrees of freedom (DOF). The blimp is autonomous and the control signals to propellers is transmitted through a remote control which is generated using Matlab. The blimp has two propellers oriented along the longitudinal axis and a single propeller oriented along the vertical axis of the airship carriage. The mathematical model of the system is created to perform and simulate a 6-DoF rigid body model of the blimp in Matlab/Simulink. The simulation results of this model have been compared with the real system and this comparison is discussed in Section 6.

Two reference frames are considered in the derivation of the kinematics and dynamical equations of motion.

4.1 Kinematic Model

Kinematics is the study of motion of an object without any consideration about the inertial and external forces acting on the object causing the motion. There are several methods to represent rotations move a reference frame to a given referred frame. In this thesis, the most common one of these methods, which is Euler angles, is used to convert earth-fixed reference frame to body-fixed reference frame. This method represents the spatial orientation of any frame of the space as a composition of rotations from a reference frame. [30].

A general spatial displacement of a rigid body consists of a finite rotation about a spatial axis and a finite translation along some vector. The rotational and translational frames in general need not be related to each other. Both frames, earth fixed inertial frame \mathcal{F}_O and body fixed frame \mathcal{F}_B , form a right handed orthogonal frame. The blimp motion has to be referenced to a system of orthogonal body axes fixed in the vehicle with the origin at the CV as can be seen in Figure 4.1. The CV is also assumed to coincide with the gross CB [31], but this leads to a more complicated dynamic model together with a controller that is more complex. Because the blimp used in our study is relatively small and the altitude variation is negligible the blimp is not actuated by any fules which might be consumed through time, the position of the CG will not change considerably and can therefore be used as the origin of the body fixed frame \mathcal{F}_B .

Since there is orthogonality between \mathcal{F}_O and \mathcal{F}_B , it can be said that the rotation $[R]$ is an orthogonal matrix and the inverse of $[R]$ equals to its transpose.

$$[R]^T = [R]^{-1} \quad (4.1)$$

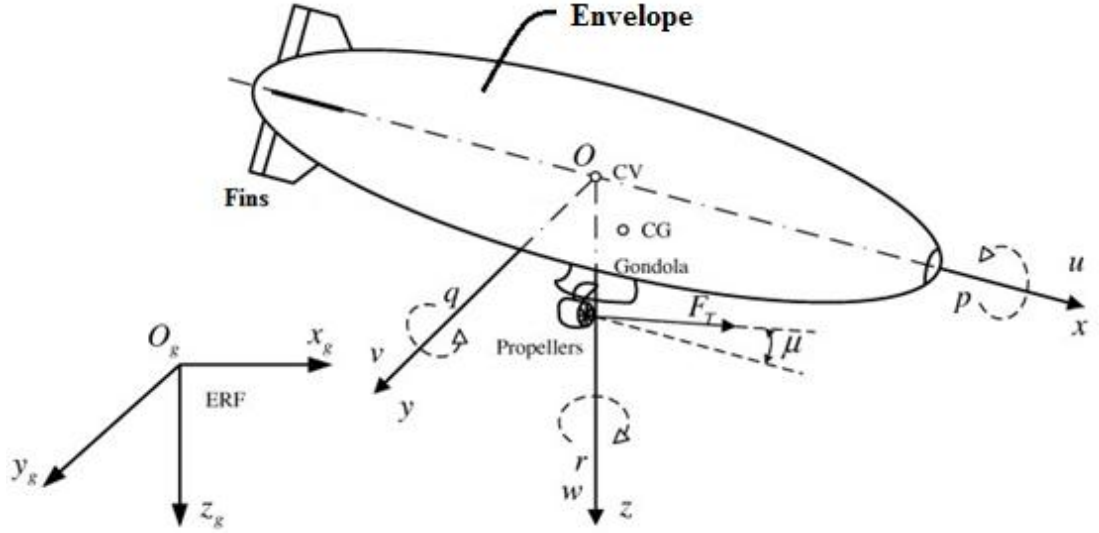


Figure 4.1 Blimp body axes coordinate system showing the CV, CG and linear (u,v,w) and angular (p,q,r) velocities around the X, Y, and Z axes respectively.

The relation between the orientation of the blimp frame \mathcal{F}_B and the inertial reference frame \mathcal{F}_0 can be expressed via a rotation matrix \mathbf{R}^{0B} :

$$\mathcal{F}_0 = \mathbf{R}^{0B} \mathcal{F}_B \quad (4.2)$$

The rotation matrix allows conversion easily between the \mathcal{F}_B and \mathcal{F}_0 back and forth and it consists of cosines and sines of angles of all possible combinations of body and earth fixed frames. Therefore, the rotation matrix is given by:

$$\mathbf{R}^{0B} = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi s\theta c\phi \\ s\psi c\theta & c\psi c\phi + s\psi s\theta s\phi & -c\psi s\phi + s\psi s\theta c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \quad (4.3)$$

where $s\phi = \sin(\phi)$ and $c\phi = \cos(\phi)$. This description is valid in the region $-\frac{\pi}{2} < \theta < \frac{\pi}{2}$. A singularity of this transformation exists for $\theta = \frac{\pi}{2} \pm k\pi; k \in \mathbb{Z}$.

The 6x1-velocity vector contains the three linear (translational) velocities u, v, w and three angular (rotational) velocities p, q, r all written with respect to \mathcal{F}_B in (4.4).

$$\mathbf{V}_{body} = (u, v, w, p, q, r)^T \quad (4.4)$$

Whereas the position and orientation of the blimp is given in (4.5)

$$\mathbf{Q} = (X, Y, Z, \phi, \theta, \psi)^T \quad (4.5)$$

are expressed in the F_0 . The angles ϕ, θ and ψ represent the roll, pitch and yaw rotation of the blimp respectively.

Then by deriving the positions an earth-fixed velocity vector can be written as in (4.6),

$$\dot{r} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \quad (4.6)$$

where r is the position vector which contains the positions of three axial coordinates.

So, multiplying $\begin{bmatrix} u \\ v \\ w \end{bmatrix}$, three linear velocities, by rotation matrix, which is given in (4.2), \dot{r} can be obtained:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = [R] \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (4.7)$$

Also similarly, the linear velocities can be obtained multiplying \dot{r} by the transpose of rotation matrix:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = [R]^T \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \quad (4.8)$$

Euler rates $(\dot{\phi}, \dot{\theta}, \dot{\psi})$ can be transformed from body-fixed frame to earth-fixed frame multiplying angular velocities (p, q, r) by a new transformation matrix $[T]$:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = [T] \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (4.9)$$

where T is:

$$[T] = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & -s\phi/c\theta & c\phi/c\theta \end{bmatrix} \quad (4.10)$$

where $s\phi = \sin(\phi)$, $t\theta = \tan(\theta)$ and $c\phi = \cos(\phi)$.

By the small angle assumption, angular rotations can be inferred from that as it can be seen in (4.11):

$$\begin{aligned}\dot{\phi} &= p \\ \dot{\theta} &= q \\ \dot{\psi} &= r\end{aligned}\tag{4.11}$$

Although the rotation matrix $[R]$ is orthogonal and its inverse is equal to its transpose, transformation matrix $[T]$ is not orthogonal, therefore $[T]^{-1} \neq [T]^T$.

Therefore, the rates of change of Euler angles in terms of earth-fixed frame are equal to multiplication of the inverse of transformation matrix by angular velocity vector which can be obtained from the Equation (4.9) by inverting the transformation matrix:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = [T]^{-1} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}\tag{4.12}$$

Thus, the earth-fixed frame velocities can be written by deriving Q , which is given in the Equation (4.5) to obtain (4.13):

$$V_{earth} = (\dot{X}, \dot{Y}, \dot{Z}, \dot{\phi}, \dot{\theta}, \dot{\psi})^T\tag{4.13}$$

Using rotation and transformation matrices, we can find both the earth-fixed frame velocities and the body-fixed frame velocities transformation easily.

In summary, matrix notation from body to earth transformation can be expressed in (4.14):

$$[V_{earth}] = \begin{bmatrix} [R] & 0 \\ 0 & [T] \end{bmatrix} [V_{body}]\tag{4.14}$$

It is also possible to express the body velocities in terms of earth velocities using (4.15):

$$[V_{body}] = \begin{bmatrix} [R]^T & 0 \\ 0 & [T]^{-1} \end{bmatrix} [V_{earth}]\tag{4.15}$$

Equation (4.14) can easily be expanded by replacing V_{earth} with the Equation (4.13), $[R]$ with the Equation (4.3) and $[T]$ with the Equation (4.10), thus the rates of change of earth-fixed frame positions and Euler angles can be produced as:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} uc\theta s\psi + v(-c\phi s\psi + s\phi s\theta c\psi) + w(s\phi s\psi + c\phi s\theta c\psi) \\ uc\theta s\psi + v(c\phi c\psi + s\phi s\theta s\psi) + w(-s\phi c\psi + c\phi s\theta s\psi) \\ -us\theta + vs\phi c\theta + wc\phi c\theta \\ p + qs\phi t\theta + rc\phi t\theta \\ qc\phi - rs\phi \\ (qs\phi + rc\phi)/c\theta \end{bmatrix} \quad (4.16)$$

4.2 Dynamical Model

In this section dynamic model of the blimp is given subject to three limiting assumptions:

1. The blimp forms a rigid body such that aero-elastic effects can be ignored, because of indoor conditions there is no forces to disturb the shape of the blimp;
2. The airframe is symmetric about the XZ plane, thus both the CV and the CG lie in the plane of symmetry.
3. The CG lies below the CB, so that the blimp is stabilized about the roll and pitch axes.

Adhering to these assumptions the required parameters for dynamic equation of motion is derived. The dynamic model consists of all forces and moments which has effect on the blimp. Therefore, the dynamical model can be stated as:

$$M\dot{V}_{body} = F_c + F_g + F_a + F_p, \quad (4.17)$$

where each of the components is described in the following sub-sections .

4.2.1 Mass and inertia

The 6x6 mass matrix contains all masses and inertias of the blimp with added mass effect accounted for as well. The mass matrix is the sum of the mass and inertia matrix of the rigid-body and added mass matrix. Since buoyant vehicle is under examination, air resistance is also taken in to account. This resistance effect on the blimp is modeled by consisting added mass (also called virtual mass) to the inertia terms.

The mass and inertia matrix of rigid-body M_{RB} is given in (4.18):

$$M_{RB} = \text{diag}(m, m, m, I_x, I_y, I_z) \quad (4.18)$$

In addition, the added mass and inertia matrix \mathbf{M}_A is written in (4.19):

$$\mathbf{M}_A = \text{diag}(m_{A_x}, m_{A_y}, m_{A_z}, I_{A_x}, I_{A_y}, I_{A_z}) \quad (4.19)$$

Due to the low speed of the blimp, the off-diagonal elements can be neglected. By combining these two matrices given in the Equation (4.18) and Equation (4.19), the total mass and inertia matrix can be calculated.

$$\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A = \begin{bmatrix} m'_x & 0 & 0 & 0 & 0 & 0 \\ 0 & m'_y & 0 & 0 & 0 & 0 \\ 0 & 0 & m'_z & 0 & 0 & 0 \\ 0 & 0 & 0 & I'_x & 0 & 0 \\ 0 & 0 & 0 & 0 & I'_y & 0 \\ 0 & 0 & 0 & 0 & 0 & I'_z \end{bmatrix} \quad (4.20)$$

where $m'_x = m + m_{A_x}$, $m'_y = m + m_{A_y}$, $m'_z = m + m_{A_z}$, $I'_x = I_x + I_{A_x}$, $I'_y = I_y + I_{A_y}$ and $I'_z = I_z + I_{A_z}$.

Taking into account of the shape of the envelope $m_{A_x} < m_{A_y} \cong m_{A_z}$, $I_{A_x} \cong 0$ and $I_{A_y} = I_{A_z}$. The virtual masses and inertia can be estimated using a geometrical method based on the kinetic energy of an ideal unbounded liquid around the hull of the blimp in motion [32]. The kinetic energy and the force necessary to accelerate the blimp can be computed by adding to the actual mass of the solid a fictitious mass. This added mass is equal to the density of the fluid multiplied by a volume depending on the geometry of the blimp only. As a result the Lamb's k-factors, k_1 and k_2 are the Lamb's inertia ratios for movements along the longitudinal (OX) and lateral (OY) axes respectively, k' is Lamb's inertia ratio for the rotation about the lateral axis (OY), and l and d are the blimp length and maximum diameter, which are shown in Figure 4.3, respectively stated in [37], [38]. To calculate the added mass terms Lamb's k factors are used as follows:

$$m_{A_x} = k_1 m \quad (4.21)$$

$$m_{A_y} = m_{A_z} = k_2 m \quad (4.22)$$

$$I_{A_x} = 0 \quad (4.23)$$

$$I_{A_y} = I_{A_z} = k' I_{z_h} \quad (4.24)$$

where I_{zh} is the inertia of displaced air and it can be calculated in terms of ellipsoid:

$$I_{zh} = \frac{1}{120} \pi \rho d l^2 (d^2 + l^2) \quad (4.25)$$

The lamb's k -factors can be obtained by using three constants α , β and e the eccentricity of the ellipsoid, which is calculated using the blimp diameters:

$$k_1 = \frac{\alpha}{2-\alpha}, \quad (4.26)$$

$$k_2 = \frac{\beta}{2-\beta}, \quad (4.27)$$

$$k' = \frac{e^4(\beta-\alpha)}{(2-e^2)(2e^2-(2-e^2)(\beta-\alpha))}, \quad (4.28)$$

where the two constants α and β are:

$$\alpha = \frac{2(1-e^2)}{e^3} \left(\frac{1}{2} \ln \left(\frac{1+e}{1-e} \right) - e \right), \quad (4.29)$$

$$\beta = \frac{1}{e^2} - \frac{(1-e^2)}{2e^3} \ln \left(\frac{1+e}{1-e} \right) \quad (4.30)$$

Moreover, the eccentricity e is:

$$e = \sqrt{1 - \left(\frac{d}{l} \right)^2} \quad (4.31)$$

4.2.2 Coriolis effect (F_c)

Coriolis effect (also called the Coriolis force) is an inertial force, which depends on the deflection of a moving object relatively to the reference frame that is earth-fixed frame. F_c contains the fictitious forces, which appear in the non-inertial frames. The Coriolis effect occurs when a motion which consists linear (u , v , w) and rotational (p , q , r) velocities. The Coriolis force can be written as:

$$F_c = C(v) + V_{body} \quad (4.32)$$

where $C(v)$ is Coriolis matrix and it is given as:

$$C(V_{body}) = \begin{bmatrix} O_{3 \times 3} & S(M_{11}v + M_{12}w) \\ S(M_{11}v + M_{12}w) & S(M_{21}v + M_{22}w) \end{bmatrix} \quad (4.33)$$

where $O_{3 \times 3}$ is 3x3 matrix which includes zeros, S is the skew-symmetric matrix operator and M_{ij} are 3x3 submatrices. So the $C(v)$ is:

$$C(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & -m'_z w & m'_y v \\ 0 & 0 & 0 & m'_z w & 0 & -m'_x u \\ 0 & 0 & 0 & -m'_y v & m'_x u & 0 \\ 0 & -m'_z w & m'_y v & 0 & -I'_z r & -I'_y q \\ m'_z w & 0 & -m'_x u & I'_z r & 0 & -I'_x p \\ -m'_y v & m'_x u & 0 & -I'_y q & I'_x p & 0 \end{bmatrix} \quad (4.34)$$

$C(v)$ includes the added-mass terms.

4.2.3 Buoyancy and gravity forces (F_g)

The buoyancy or aerostatic lift force is, as explained by Archimedes' principle, is an upward force which opposes the weight of the air, which is displaced by the blimp.

The magnitudes of the buoyancy and gravitational forces can be explained as follows:

$$F_{gr} = mg \text{ and } F_b = \rho V g, \text{ with the volume } V = \frac{1}{12} \pi l d^2 \quad (4.35)$$

where m is the mass of the blimp, g is the gravitational acceleration, ρ is the density of air and V is the volume of the blimp.

F_g is the net force vector which is the difference between magnitudes of the buoyancy force and the force of gravity.

$$F_g = \begin{bmatrix} -(F_{gr} - F_b) s \theta \\ (F_{gr} - F_b) c \theta s \phi \\ -(F_{gr} - F_b) c \theta c \phi \\ -r_z F_b c \theta s \phi \\ -r_z F_b s \theta \\ 0 \end{bmatrix} \quad (4.36)$$

where r_z is the distance between CG and CB.

4.2.4 Aerodynamic forces (F_a)

Aerodynamic forces vector contains the aerodynamic terms of the model, arising from the aerodynamics of the airship's hull and control surfaces.

$$F_a = D(v)V_{body} \quad (4.37)$$

$D(v)$ contains the non-dimensional linear coefficients of drag D_u and D_p , side D_v and D_q and lift D_w and D_r and quadratic coefficients of yaw D_{u^2} and D_{p^2} , pitch D_{v^2} and D_{q^2} and roll D_{w^2} and D_{r^2} . $D(v)$ can be written as:

$$D(v) = \begin{bmatrix} D_u + D_{u^2}|u| \\ D_v + D_{v^2}|v| \\ D_w + D_{w^2}|w| \\ D_p + D_{p^2}|p| \\ D_q + D_{q^2}|q| \\ D_r + D_{r^2}|r| \end{bmatrix} \quad (4.38)$$

4.2.5 Propulsion forces (F_p)

Propulsion forces are generated by the propellers which are connected to DC motors. There are three thrusters which are attached to the gondola. Two of the thrusters are attached laterally and they provide thrust in the forward/backward direction. These two lateral thrusters are attached to the starboard side and to the port side of the gondola and they are also used to steer the blimp. The third thruster provides thrust in the vertical direction to change the altitude. In Figure 4.2, the schematic display of the propellers of the blimp is given. Also in the model, three thrusters are assumed to have ideal characteristic that the effects of the motor commands are directly proportional.

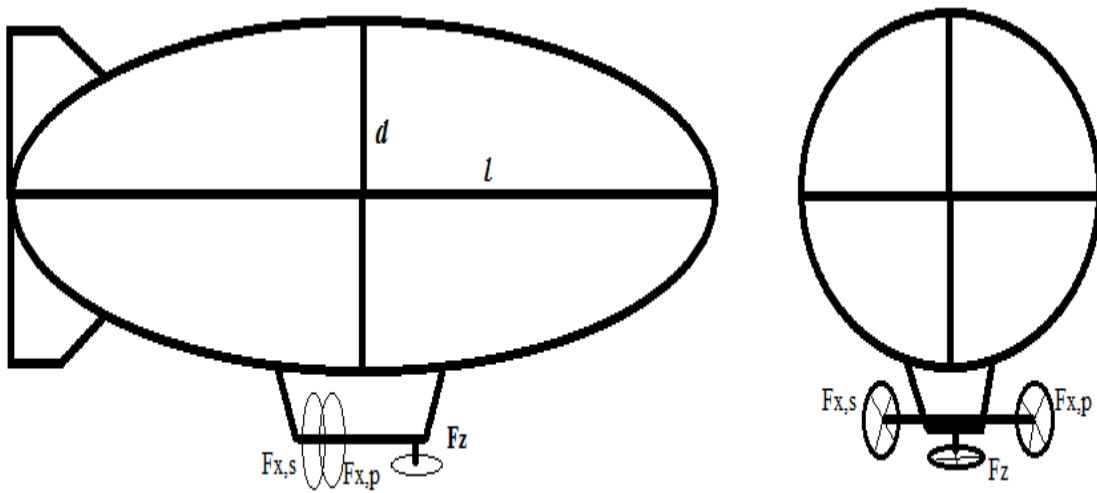


Figure 4.2 The location of the thrusters, the length, and the maximum diameter of the blimp.

Now, the propulsion vector F_p can be derived which contains the terms associated with the propulsive forces and moments, and it is a function of the geometrical arrangement of the propulsive units around the body axes.

$$F_p = \begin{bmatrix} F_{x,s} + F_{x,p} \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix} \quad (4.39)$$

where $F_{x,s} + F_{x,p}$, F_y and F_z indicates the total thrust along the OX, OY and OZ axes respectively. There is no propeller to generate thrust along OY, then $F_y = 0$; M_x , M_y and M_z are the total thrust moments along the same axes and considering the distances, the moments can be written as:

$$M_x = 0 \quad (4.40)$$

$$M_y = (F_{x,p} - F_{x,s})r_x \quad (4.41)$$

$$M_z = (F_{x,s} - F_{x,p})r_y \quad (4.42)$$

where r_y is the horizontal y distance from CB to propeller perpendicular to main axes of the blimp and r_x is the horizontal distance from CB to propeller along the main axis of the blimp. The blimp does not have any thrust vectoring capability and simply relies on the either forward or reverse thrust to maneuver.

4.3 Longitudinal and Lateral Models, LQR Method

In this section linearization of blimp's model underlying longitudinal and lateral models are given. Using these models state-space forms are arranged and detailed transfer function results are given at Appendix C. To obtain the linearized longitudinal and lateral equations of motions for blimp used in this thesis are taken from [38].

Without using a controller, it is seen that the blimp model is unstable and by investigating the results, the controllability is obtained 4 for both linearized longitudinal and lateral state-space models. Since the system can be stabilized, linear quadratic regulation (LQR) method is used in this thesis.

4.3.1 Longitudinal equations

The linearized longitudinal equations of motions for blimp used in this thesis are given in the following equation set in (4.43) to (4.47).

$$\begin{aligned}
m_x \dot{u} + (ma_z - \dot{X}_{\dot{q}}) \dot{q} \\
= X_e + \dot{X}_u u + \dot{X}_w w + (\dot{X}_q - m_z W_e) q + \dot{X}_\delta (\delta_e + \delta_r) + \dot{X}_t \delta_t + T_e \\
- (mg - B)(\sin\theta_e + \theta \cos\theta_e)
\end{aligned} \tag{4.43}$$

$$\begin{aligned}
m_z \dot{w} - (ma_x - \dot{Z}_{\dot{q}}) \dot{q} \\
= Z_e + \dot{Z}_u u + \dot{Z}_w w + (\dot{Z}_q - m_x u_e) q + \dot{Z}_{\delta_e} \delta_e \\
- (mg - B)(\cos\theta_e - \dot{\theta} \sin\theta_e)
\end{aligned} \tag{4.44}$$

$$\begin{aligned}
J_y \dot{q} + (ma_x - \dot{M}_{\dot{u}}) \dot{u} - (ma_x - \dot{M}_w) \dot{w} \\
= -\theta((mga_z + Bb_z)\cos\theta_e - (mga_x + Bb_x)\sin\theta_e) \\
- (mga_z + Bb_z)\sin\theta_e \\
- (mga_x + Bb_x)\cos\theta_e
\end{aligned} \tag{4.45}$$

Thus the longitudinal state-space matrix A_{long} and the control matrix B_{long} are given as follows in (4.46) and (4.47):

$$A_{long} = \begin{bmatrix} \dot{x}_u & \dot{x}_w & (\dot{x}_q - m_z W_e) & -(mg - B)\cos\theta_e \\ \dot{z}_u & \dot{z}_w & (\dot{z}_q + m_x U_e) & -(mg - B)\sin\theta_e \\ \dot{M}_u & \dot{M}_w & (\dot{M}_q - ma_x U_e - ma_z W_e) & -(mga_z + Bb_z)\cos\theta_e + (mga_x \sin\theta_e) \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.46}$$

$$B_{long} = \begin{bmatrix} x_\delta & x_t \\ z_\delta & 0 \\ m_\delta & m_t \\ 0 & 0 \end{bmatrix} \tag{4.47}$$

4.3.2 Lateral equations

Similarly, with longitudinal equations lateral equations are obtained as follows given in (4.48) to (4.52):

$$\begin{aligned}
m_y \dot{v} - (ma_z - \dot{Y}_p) \dot{p} + (ma_x - \dot{Y}_r) \dot{r} \\
= Y_e + \dot{Y}_v v + (\dot{Y}_p + m_z W_e) q + (\dot{Y}_r - m_x U_e) r + \dot{Y}_{\delta_r} \delta_r \\
+ (mg - B) \phi \cos \theta_e
\end{aligned} \tag{4.48}$$

$$\begin{aligned}
J_z \dot{r} - J_{xz} \dot{p} + (ma_x - \dot{N}_v) \dot{v} \\
= N_e + \dot{N}_v v + (\dot{N}_p + ma_x W_e) p + (\dot{N}_r - ma_x U_e) r + \dot{N}_{\delta_r} \delta_r \\
+ (mga_x + Bb_x) \phi \cos \theta_e
\end{aligned} \tag{4.49}$$

$$\begin{aligned}
J_x \dot{p} - J_{xz} \dot{r} - (ma_z - \dot{L}_v) \dot{v} \\
= L_e + \dot{L}_v v + (\dot{L}_p + ma_x W_e) p + (\dot{L}_r + ma_z U_e) r \\
- (mga_z \\
+ Bb_z) \phi \cos \theta_e
\end{aligned} \tag{4.50}$$

Finally, state-space matrix A_{lat} and the control matrix B_{lat} are obtained as follows:

$$A_{lat} = \begin{bmatrix} y_v & y_p & y_r & y_\phi \\ l_v & l_p & l_r & z_\phi \\ n_v & n_p & n_r & n_\phi \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.51}$$

$$B_{lat} = \begin{bmatrix} u_\delta \\ 0 \\ n_\delta \\ 0 \end{bmatrix} \tag{4.52}$$

4.3.3 Linear quadratic regulation

Linear quadratic regulation method is used in different areas and it is a well-known effective optimization problem that aims to minimize the accuracy of control opposite power exacting; $J(u) = \int_0^\infty (x(t)^T Q x(t) + u^T R u(t) dt)$ (where $x(t)$ state of the system, Q is a weighting matrix for the states and R is a weighting matrix for the input signals, $u(t)$ control input) [39] as being a cost function. Additionally, a brief description of LQR state feedback design is given below which is used to stabilize both linearized longitudinal and lateral models:

Considering the linear time invariant system,

$$\begin{aligned}
\dot{x} &= Ax + Bu \\
y &= Cx + Du
\end{aligned} \tag{4.53}$$

where $y(t)$ is output vector. The state feedback is obtained as follows if all states are measurable (through 9 DoF IMU, all states of the blimp used in this thesis are measurable),

$$u = -Kx \quad (4.54)$$

where K is the state feedback gain matrix that can give desirable closed loop properties.

$$\dot{x} = (A - BK)x = A_c x \quad (4.55)$$

where A_c is closed loop plant matrix. Then rearranging the cost function of LQR controller with (4.54) yields:

$$J = \frac{1}{2} \int_0^\infty (x(t)^T (Q + K^T R K) x(t)) dt \quad (4.56)$$

To minimize the cost function, (4.56) must be finite. In order to find the optimal feedback, K , a constant matrix, P , is assumed to be existed such that:

$$\frac{d}{dt}(x^T P x) = -x^T (Q + K^T R K) x \quad (4.57)$$

Substituting (4.57) into (4.56) results in

$$J = \frac{1}{2} \int_0^\infty \frac{d}{dt}(x^T P x) dt = \frac{1}{2} x^T(0) P x(0) \quad (4.58)$$

From this equation, it is clear that J is a constant that only depends on the matrix P and the initial conditions. Hence, substituting the differentiated form of (4.57) into (4.55) yields:

$$x^T (A_c^T P + P A_c + Q + K^T R K) x = 0 \quad (4.59)$$

And thus,

$$A_c^T P + P A_c + Q + K^T R K = 0 \quad (4.60)$$

Substituting (4.55) into (4.60) yields

$$A^T P + P A + Q + K^T R K - K^T B^T P - P B K = 0 \quad (4.61)$$

Assuming that $K = R^{-1}B^T P$ is selected, the following result, which is the Algebraic Riccati Equation, can be obtained.

$$A^T P + PA + Q - PBR^{-1}B^T P = 0 \quad (4.62)$$

4.4 Simulation Model of Blimp

After obtaining linearized longitudinal and lateral trimmed equations, a basic blimp model is formulated in Matlab using real parameters of the blimp used in thesis, which are given in Table 4.1, to simulate and compare results with experimental data. On the other hand, this model is used to analyze the stability of blimp.

Table 4.1 Dimensional Parameters of Blimp.

Mass (kg)	0.255
Distance Between CG and CB (m) (in x-axis)	0
Distance Between CG and CB (m)(in z-axis)	0.21
Distance Between CB and CV (m)(in x-axis)	0
Distance Between CB and CV (m)(in z-axis)	0.14
Maximum Diameter (d) (m)	0.9
Maximum Length (l) (m)	1.16
Vertical Diameter (c) (m)	0.5
Volume (m^3)	0.492
Reference Area (m^2)	0.6232

First of all, it is analyzed that whether the open-loop system (without any control) is stable to see if the system is inherently stable. To investigate the stability, the eigenvalues of the system matrix, A , is used to determine the stability, and the poles for longitudinal system matrix, A_{long} , and lateral system matrix, A_{lat} , are obtained:

$$poles_{long} = \begin{matrix} -32.1675 \\ 0.5032 \\ 0.2573 + 7.3474i \\ 0.2573 - 7.3474i \end{matrix} \quad , \quad poles_{lat} = \begin{matrix} -20.4076 \\ 2.3852 \\ -6.7155 \\ -8.4335 \end{matrix}$$

These values are obtained using “*eig*” function at Matlab, which computes a vector containing the eigenvalues of a square matrix. It is clear for both systems that one of the poles has positive values and this means that these systems are unstable in open-

loop. Briefly, at pole-zero plots, there is real-axis at horizontal and imaginary-axis is at vertical, furthermore left side of imaginary-axis indicates stable situations and right side of imaginary-axis shows unstable situations.

In order to check our system without a controller, initial conditions are set to 0.1 and these results are given in Figure 4.3. From Figure 4.3, it is clear that blimp's longitudinal model states are increasing harmonically to infinite, on the other hand it can be seen that there 3 positive poles and 2 of them are imaginary, as a result this system is unstable. Therefore, lateral states directly go to infinity, lateral model has one positive pole, and there are no lateral model poles at imaginary axis.

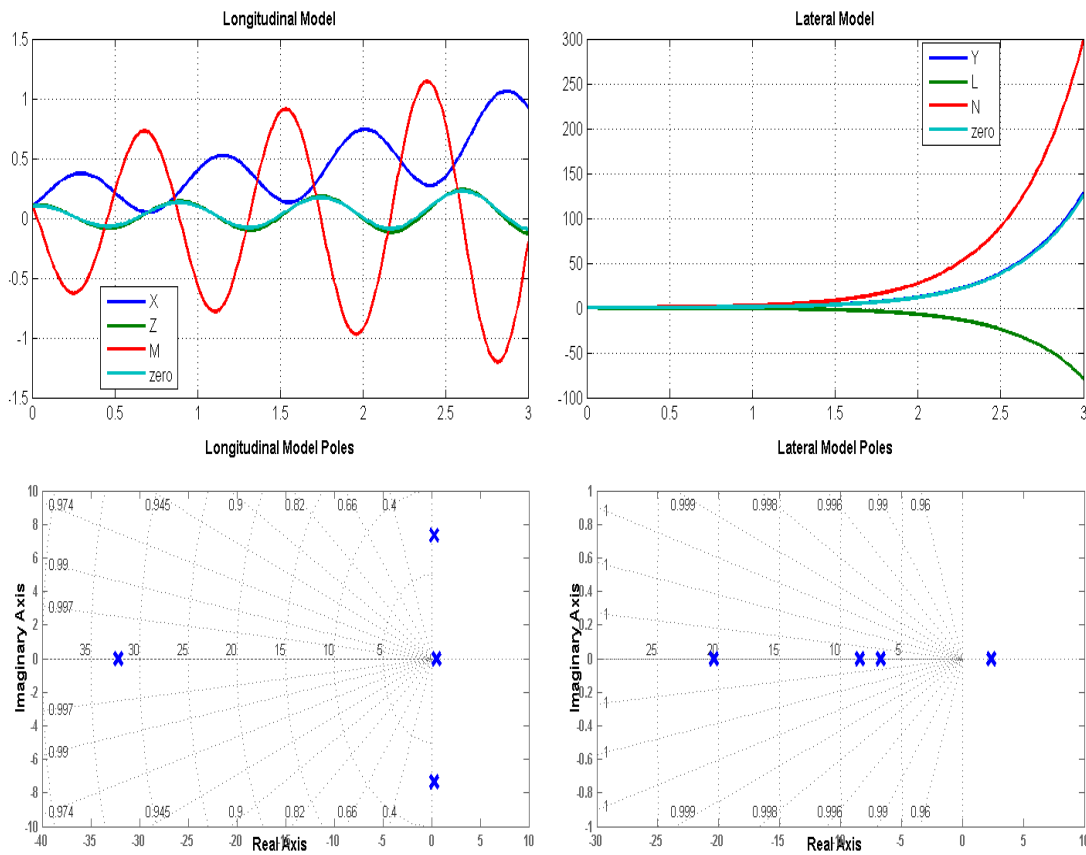


Figure 4.3 Longitudinal and Lateral models without a controller.

Considering our system is a continuous linear time-invariant system, controllability is computed using “*ctrb*” command of Matlab. On the other hand, it is said that supposing A is $n \times n$ state matrix, B is $n \times r$ input matrix R is $n \times n$ controllability matrix which includes $[B \ AB \ A^2B \ \dots \ A^{n-1}B]$, the system is controllable if the controllability matrix has full rank. Since the controllability is obtained four for each models, it is

possible to get these models stable. Thus, as mentioned before, LQR is applied as a controller to our system.

$$poles_{long_LQR} = \begin{matrix} -331.1933 \\ -23.0429 \\ -1.3683 + 6.3771i \\ -1.3683 - 6.3771i \end{matrix} , poles_{lat_LQR} = \begin{matrix} -41.4973 \\ -3.3810 + 0.8310i \\ -3.3810 - 0.8310i \\ -9.2483 \end{matrix}$$

After LQR is applied as a controller method, it can be seen from the poles of each models have negative values, in other words, poles are at the left side of plots and this means these models are stable after applying LQR.

In Figure 4.4, LQR applied longitudinal and lateral positions obtained from simulation are given. It is clear that the states of both models are being stabilized. Even if unstable lateral model has no imaginary poles, after LQR is applied, lateral model has two symmetric (according to the real-axis) poles.

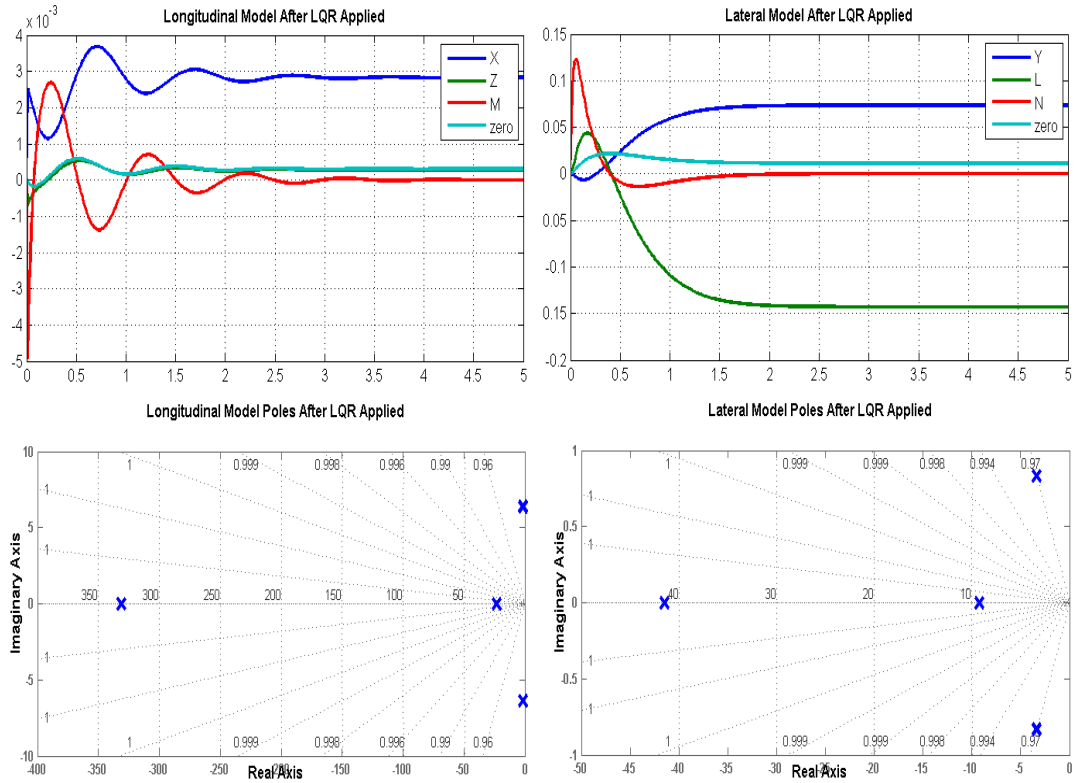


Figure 4.4 Longitudinal and Lateral models after LQR applied.

In an addition to these simulations, a basic Simulink model without a controller of blimp including a VRML visualization is created as shown in Figure 4.5, this model is also unstable. As a future work, an LQR controller with a Kalman filter will be

added to this simulation to obtain better results, additionally this model will be validated after comparing real system data using system identification tools.

In order to run this simulation, a Simulink built-in block called 6 DoF (Euler Angles) is used mainly. This block determines any 6 DoF system as a rigid body and considers the rotation of a body-fixed frame about a flat Earth reference frame.

There are simply 2 inputs to the 6 DoF block that are forces and moments at 3-axes. Three DC motors of blimp generate forces, besides for the simulation motors are triggered by random signal generators. Moments are generated with respect to blimp parameters and forces also. Same as forces, random signal generators to obtain a simulation result generate moments.

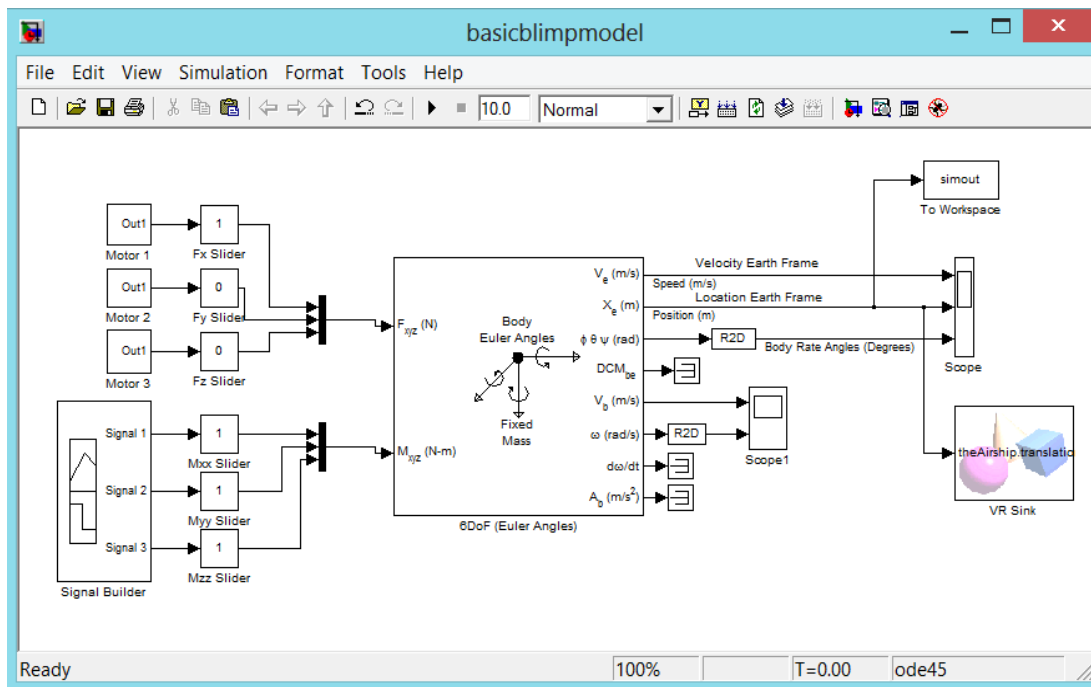


Figure 4.5 Basic Blimp Model in Simulink.

In order to visualize this simulations results, two different tools are used. Firstly, using Matlab's standard "*plot3*" function, the position of blimp is displayed, secondly using VRML interface a 3D simple blimp is created, and the movement due to simulation is shown by this interface translational parameters. In Figure 4.6, blimp position is displayed at three axes. Sampling time of this simulation is 0.2 seconds and these samples are indicated by red triangles at this figure. It is clear from the figure that red triangles are much more at the beginning of the movement because blimp movement is congested and slower at the beginning then blimp moves faster.

As discussed before any controller is applied this simulation and as expected blimp movement is meaninglessly ends at -10 meters at z-axis such as there is an underground movement has been made.

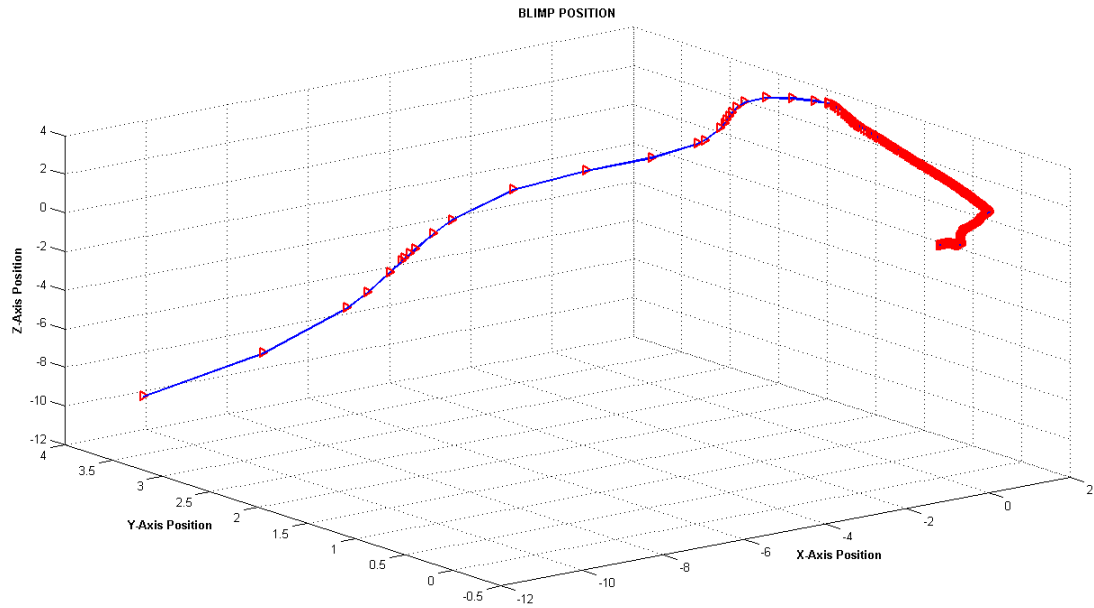


Figure 4.6 Blimp position visualization.

With the help of “VR Tracer”, a Simulink block, blimp’s virtual scene trajectory is traced. In order to connect the blimp as an object to this block, a blimp is created using VRML. Therefore, blimp’s trajectory is made available to trace and visualize by creating marker nodes in regular time steps that is 0.2 seconds for this simulation. In Figure 4.7, this simulation is displayed.

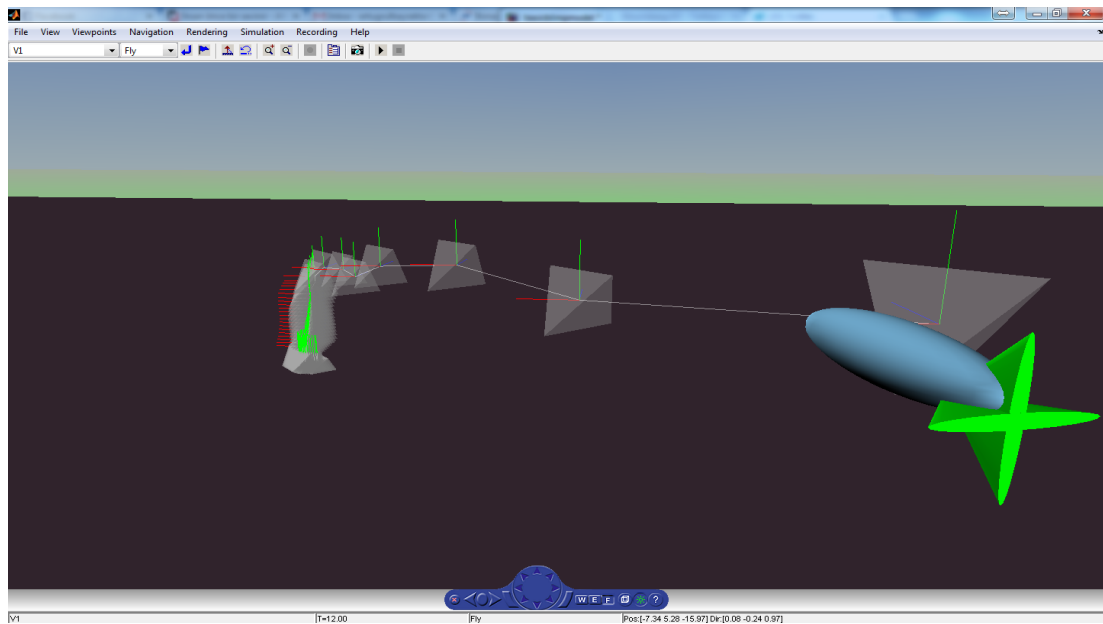


Figure 4.7 Blimp’s movements in VRML.

5. EXPERIMENTS AND RESULTS

At this section, the methods, which are used in this thesis to estimate the position of the blimp, are given. Firstly, in section 5.1 vision experiments are covered by analyzing the compatibility of the method, which is used to match real-time grabbed images from wireless onboard camera with dataset, within this study underlying its theoretical basics. Secondly, in Section 5.2 navigation method and simulation results are given.

5.1 Vision Experiments

To handle the dataset images easily in software running at VS using OpenCV, images are given a special coded name and documented with respect to an order as shown in Figure 5.1. First character of the documentation “p” is used to show the points and 2 digits following point character are used to indicate the point number from 1 to 45 which are created while creating the image dataset. The character “h” is used to specify the next 3 digits shown by “YYY” indicate the height in centimeters from 100 to 200 by increasing 50 and the last 3 digits show the degree from 0 to 345 increasing 15.

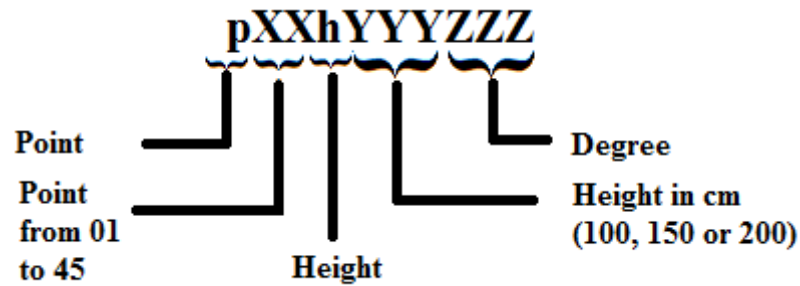


Figure 5.1 Documentation style and explanation of dataset image entitles.

In order to match images or template objects between dataset images and real-time grabbed images, Speeded-Up Robust Features (SURF) method [40] is used. SURF is a method that is mainly used to match images using detectors and descriptor which are specific for images. Additionally, SURF’s novelty comes from to have scale and

rotation invariant detector and descriptor. Because of these specialties of SURF, to avoid any wrong or worse localization while matching images, a distance value, which indicates differences between images, is assigned and lowest distance between matched images is accepted matched images.

This method uses a very basic Hessian-matrix approximation to detect interest points. In an addition, relying on integral images for image convolutions descriptors are achieved distribution-based.

As a sample in Figure 5.2, a collage of images is shown which are grabbed from same point (p9) and same height (h200) by rotating the test-bed with 15 degrees (from 000 to 345) clockwise.



Figure 5.2 A sample of dataset images as a collage.

As a result of using a low-cost frame grabber, sometimes colors of images deteriorate. To avoid any unexpected situations while matching images, both colored and greyscale images are used. Localization experiments are made at both operating systems (Ubuntu 12.10 and Windows) using each color options (colored or greyscale). Additionally, at Windows Visual Studio 2010 Professional IDE and at Ubuntu 12.10 Eclipse 3.8 IDE are used to run image matching and localization using matched image data algorithms.

In Figure 5.33, a brief visualization and interpretation of the relation between titled dataset images with their titles and the localization of points, which titles are derived from, are shown.

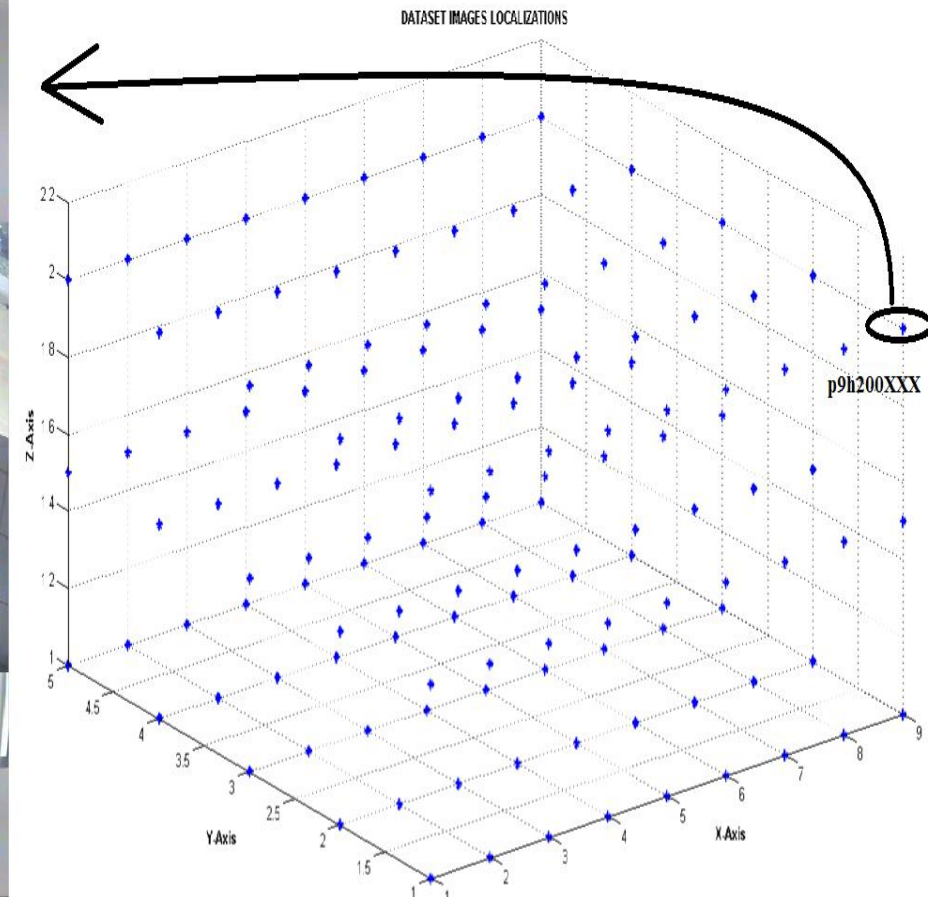


Figure 5.3 Relation between images and titles, which are derived from points that are images, grabbed from.

In Figure 5.4, a matching experiment is shown which is made at Windows for each both color options. Blue circle indicates matched specification between grabbed and dataset images at most number, additionally there are some matching points with other dataset images but blue circles images distance is the lowest to the grabbed image.



Figure 5.4 Image matching at Windows using SURF for greyscale and colored.

Also detector (keypoint 2) and descriptor (keypoint 2) numbers for colored and greyscale images which are generated by image matching algorithm used in this thesis are given in Figure 5.5. Based on these results it is clear that algorithm time for matching colored images is lower than matching greyscale images, at the same time minimum distance for colored images is much than greyscale images likewise the maximum distance is.

Colored Images	Greyscale Images
Max dist : 1.003013	Max dist : 0.990940
Min dist : 0.068058	Min dist : 0.068055
Good Match [0] Keypoint 1: 295 -- Keypoint 2: 35	Good Match [0] Keypoint 1: 22 -- Keypoint 2: 5
Good Match [1] Keypoint 1: 402 -- Keypoint 2: 41	Good Match [1] Keypoint 1: 296 -- Keypoint 2: 35
Good Match [2] Keypoint 1: 407 -- Keypoint 2: 34	Good Match [2] Keypoint 1: 402 -- Keypoint 2: 41
Good Match [3] Keypoint 1: 577 -- Keypoint 2: 47	Good Match [3] Keypoint 1: 576 -- Keypoint 2: 47
Good Match [4] Keypoint 1: 703 -- Keypoint 2: 90	Good Match [4] Keypoint 1: 701 -- Keypoint 2: 89
Good Match [5] Keypoint 1: 910 -- Keypoint 2: 141	Good Match [5] Keypoint 1: 910 -- Keypoint 2: 140
Good Match [6] Keypoint 1: 919 -- Keypoint 2: 88	Good Match [6] Keypoint 1: 916 -- Keypoint 2: 88
Good Match [7] Keypoint 1: 1179 -- Keypoint 2: 141	Good Match [7] Keypoint 1: 1176 -- Keypoint 2: 140
Good Match [8] Keypoint 1: 1682 -- Keypoint 2: 160	Good Match [8] Keypoint 1: 1681 -- Keypoint 2: 159
Good Match [9] Keypoint 1: 1964 -- Keypoint 2: 253	Good Match [9] Keypoint 1: 1961 -- Keypoint 2: 253
Good Match [10] Keypoint 1: 2127 -- Keypoint 2: 386	Good Match [10] Keypoint 1: 2121 -- Keypoint 2: 384
Good Match [11] Keypoint 1: 2358 -- Keypoint 2: 504	Good Match [11] Keypoint 1: 3148 -- Keypoint 2: 467
Good Match [12] Keypoint 1: 3229 -- Keypoint 2: 386	Good Match [12] Keypoint 1: 3222 -- Keypoint 2: 384
Good Match [13] Keypoint 1: 3239 -- Keypoint 2: 421	Good Match [13] Keypoint 1: 3232 -- Keypoint 2: 418
Process Time 6.566000	Process Time 6.598000

Figure 5.5 Keypoint results comparison between colored and greyscale images matching at Windows.

In Figure 5.6, image matching for localization results are shown which are obtained running our program at Ubuntu operating system. Similarly with results which are obtained from Windows, though matches with some other images blue circled images have most matched keypoints for each color options at Ubuntu. The reason of these matched keypoints with other images is that the SURF algorithm detects some keypoints those come from same objects or similarities of different images.

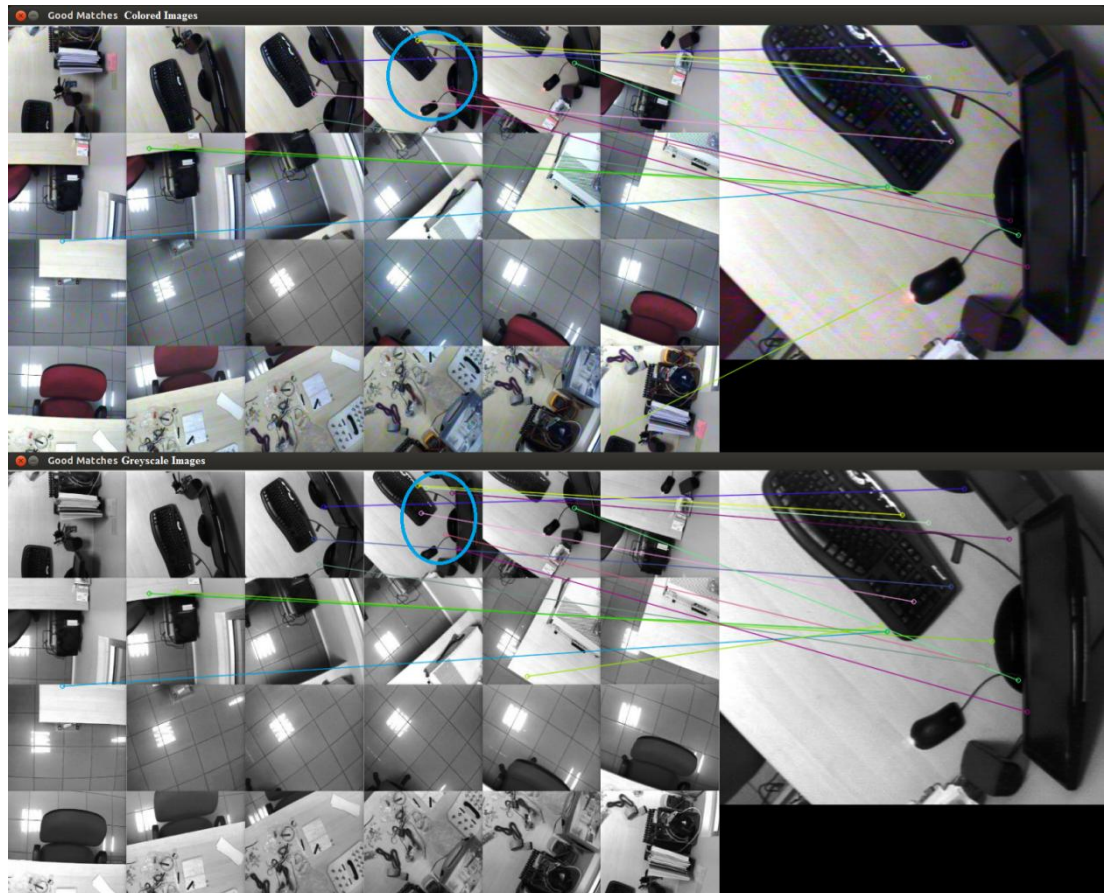


Figure 5.6 Image matching at Ubuntu using SURF for greyscale and colored.

In Figure 5.7 image matching algorithm, which is run at Ubuntu, outputs are given for both colored and greyscale images. Although keypoint and minimum distance values are same with Windows results, maximum distance and process time values are very different.

Colored Images	Greyscale Images
-- Max dist : 0.948940	-- Max dist : 1.019230
-- Min dist : 0.068058	-- Min dist : 0.068055
-- Good Match [0] Keypoint 1: 295 -- Keypoint 2: 35	-- Good Match [0] Keypoint 1: 22 -- Keypoint 2: 5
-- Good Match [1] Keypoint 1: 402 -- Keypoint 2: 41	-- Good Match [1] Keypoint 1: 296 -- Keypoint 2: 35
-- Good Match [2] Keypoint 1: 407 -- Keypoint 2: 34	-- Good Match [2] Keypoint 1: 402 -- Keypoint 2: 41
-- Good Match [3] Keypoint 1: 577 -- Keypoint 2: 47	-- Good Match [3] Keypoint 1: 576 -- Keypoint 2: 47
-- Good Match [4] Keypoint 1: 703 -- Keypoint 2: 90	-- Good Match [4] Keypoint 1: 701 -- Keypoint 2: 89
-- Good Match [5] Keypoint 1: 910 -- Keypoint 2: 141	-- Good Match [5] Keypoint 1: 910 -- Keypoint 2: 140
-- Good Match [6] Keypoint 1: 919 -- Keypoint 2: 88	-- Good Match [6] Keypoint 1: 916 -- Keypoint 2: 88
-- Good Match [7] Keypoint 1: 1179 -- Keypoint 2: 141	-- Good Match [7] Keypoint 1: 1176 -- Keypoint 2: 140
-- Good Match [8] Keypoint 1: 1682 -- Keypoint 2: 160	-- Good Match [8] Keypoint 1: 1681 -- Keypoint 2: 159
-- Good Match [9] Keypoint 1: 1964 -- Keypoint 2: 253	-- Good Match [9] Keypoint 1: 1961 -- Keypoint 2: 253
-- Good Match [10] Keypoint 1: 2127 -- Keypoint 2: 386	-- Good Match [10] Keypoint 1: 2121 -- Keypoint 2: 384
-- Good Match [11] Keypoint 1: 2358 -- Keypoint 2: 504	-- Good Match [11] Keypoint 1: 3148 -- Keypoint 2: 467
-- Good Match [12] Keypoint 1: 3229 -- Keypoint 2: 386	-- Good Match [12] Keypoint 1: 3222 -- Keypoint 2: 384
-- Good Match [13] Keypoint 1: 3239 -- Keypoint 2: 421	-- Good Match [13] Keypoint 1: 3232 -- Keypoint 2: 418
Process Time 1.640000	Process Time 1.600000

Figure 5.7 Keypoint results comparison between colored and greyscale images matching at Ubuntu.

5.2 Localization Experiments

In this section, detailed information about 9 DoF IMU used in this thesis and its firmware, localization algorithm, data structure and data types, wireless communication and hardware are given.

As discussed before Razor 9 DoF IMU is used in this thesis and its open source AHRS firmware [41] is uploaded to it directly via an FTDI interface. In Figure 5.8 board layout of 9 DoF IMU is shown with default directions at 3-axes for sensors.

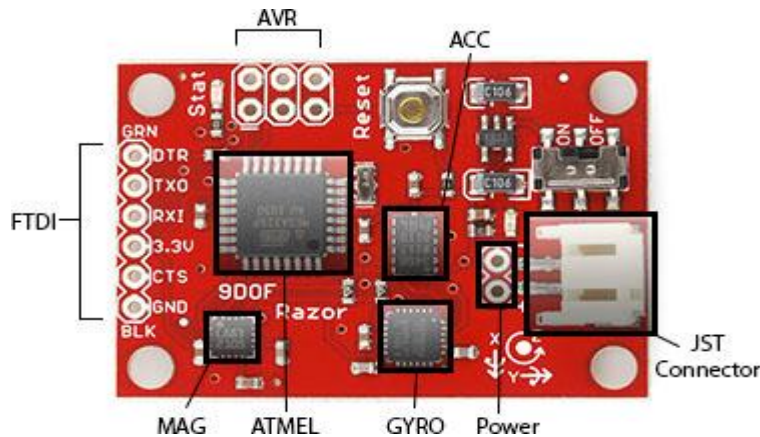
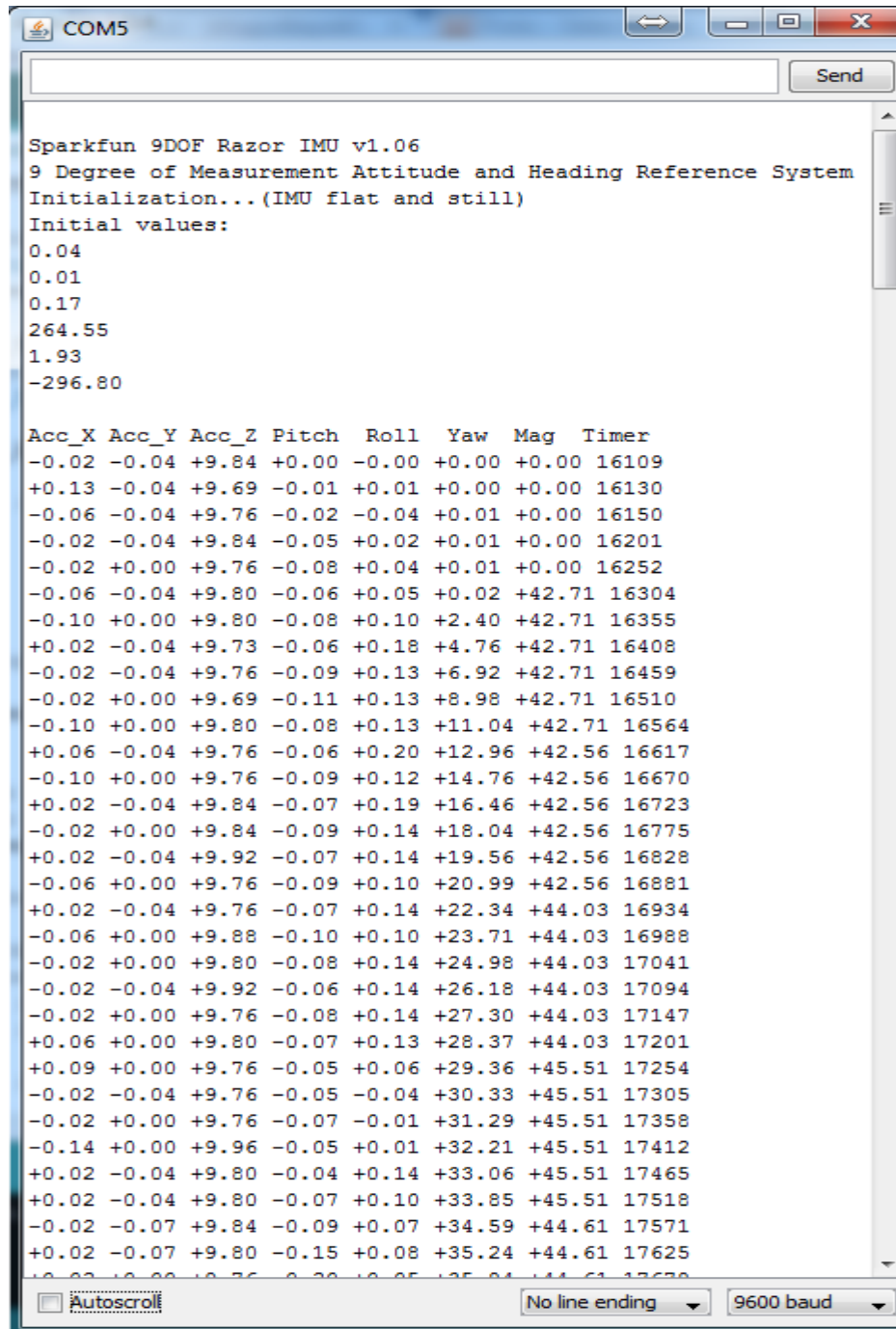


Figure 5.8 Board layout and components of Razor 9 DoF IMU.

Using open source firmware, regulated data is obtained easily being an array, of which first three columns are from accelerometer, second three columns are from gyroscope, seventh column is computed magnetometer heading and last columns is time in milliseconds as shown in Figure 5.9. Besides, when the algorithm is run, IMU gives the initial conditions at first and then waits for 1000 milliseconds (this value can be changed to any desired value) to transmit regular data. This specification is very useful for simple localization applications.



```

COM5
Sparkfun 9DOF Razor IMU v1.06
9 Degree of Measurement Attitude and Heading Reference System
Initialization...(IMU flat and still)
Initial values:
0.04
0.01
0.17
264.55
1.93
-296.80

Acc_X Acc_Y Acc_Z Pitch Roll Yaw Mag Timer
-0.02 -0.04 +9.84 +0.00 -0.00 +0.00 +0.00 16109
+0.13 -0.04 +9.69 -0.01 +0.01 +0.00 +0.00 16130
-0.06 -0.04 +9.76 -0.02 -0.04 +0.01 +0.00 16150
-0.02 -0.04 +9.84 -0.05 +0.02 +0.01 +0.00 16201
-0.02 +0.00 +9.76 -0.08 +0.04 +0.01 +0.00 16252
-0.06 -0.04 +9.80 -0.06 +0.05 +0.02 +42.71 16304
-0.10 +0.00 +9.80 -0.08 +0.10 +2.40 +42.71 16355
+0.02 -0.04 +9.73 -0.06 +0.18 +4.76 +42.71 16408
-0.02 -0.04 +9.76 -0.09 +0.13 +6.92 +42.71 16459
-0.02 +0.00 +9.69 -0.11 +0.13 +8.98 +42.71 16510
-0.10 +0.00 +9.80 -0.08 +0.13 +11.04 +42.71 16564
+0.06 -0.04 +9.76 -0.06 +0.20 +12.96 +42.56 16617
-0.10 +0.00 +9.76 -0.09 +0.12 +14.76 +42.56 16670
+0.02 -0.04 +9.84 -0.07 +0.19 +16.46 +42.56 16723
-0.02 +0.00 +9.84 -0.09 +0.14 +18.04 +42.56 16775
+0.02 -0.04 +9.92 -0.07 +0.14 +19.56 +42.56 16828
-0.06 +0.00 +9.76 -0.09 +0.10 +20.99 +42.56 16881
+0.02 -0.04 +9.76 -0.07 +0.14 +22.34 +44.03 16934
-0.06 +0.00 +9.88 -0.10 +0.10 +23.71 +44.03 16988
-0.02 +0.00 +9.80 -0.08 +0.14 +24.98 +44.03 17041
-0.02 -0.04 +9.92 -0.06 +0.14 +26.18 +44.03 17094
-0.02 +0.00 +9.76 -0.08 +0.14 +27.30 +44.03 17147
+0.06 +0.00 +9.80 -0.07 +0.13 +28.37 +44.03 17201
+0.09 +0.00 +9.76 -0.05 +0.06 +29.36 +45.51 17254
-0.02 -0.04 +9.76 -0.05 -0.04 +30.33 +45.51 17305
-0.02 +0.00 +9.76 -0.07 -0.01 +31.29 +45.51 17358
-0.14 +0.00 +9.96 -0.05 +0.01 +32.21 +45.51 17412
+0.02 -0.04 +9.80 -0.04 +0.14 +33.06 +45.51 17465
+0.02 -0.04 +9.80 -0.07 +0.10 +33.85 +45.51 17518
-0.02 -0.07 +9.84 -0.09 +0.07 +34.59 +44.61 17571
+0.02 -0.07 +9.80 -0.15 +0.08 +35.24 +44.61 17625
-0.02 -0.07 +9.76 -0.15 +0.05 +35.84 +44.61 17678
  
```

Figure 5.9 IMU Data Structure.

With the aim of having localization result from IMU data, a simple algorithm is created for both C++ and Matlab.

In order to compute velocity following equation is used and as a future work an appropriate Kalman filter will be developed and this filter will be supported with experiments to decrease integration and drift based errors.

$$Velocity = (Acc_{(t)} - Acc_{(t-1)}) * (timer_{(t)} - timer_{(t-1)}) \quad (5.1)$$

Similarly, position can be computed as follows:

$$Position = (Acc_{(t)} - Acc_{(t-1)}) * (timer_{(t)} - timer_{(t-1)})^2 \quad (5.2)$$

Afterwards computed velocity and position data are used to have a plot at 3-axes to visualize system outputs. Additionally position data is used to both fuse with image matching algorithm localization output and comparison with it.

6. CONCLUSION AND RESULTS

In this thesis a blimp is modified to be an autonomous aerial vehicle by equipping with camera for localization by matching real-time grabbed frames with a 3090-image dataset that are titled the locations, and IMU for computing the position of the vehicle. Additionally, an onboard power regulator board designed and manufactured to give power to default motors of blimp (3V), to camera (7.5V) and to IMU (6V). Besides, power regulator board is directly connected to the Li-Po battery.

In order to obtain IMU data wirelessly, 9 DoF IMU is connected to an XBee with the help of Arduino Uno board and XBee shield on it. At this part of onboard components, Arduino Uno board and XBee shield are just used to regulate the data from any interrupters and to avoid any noise which may cause from power levels. Before this connection is completed, direct IMU-XBee connection is tried and it has seen that it is impossible to obtain regular data, besides there is some missing data.

As discussed before, after completing flight experiments two localization outputs obtained from this additional equipment will be fused and in order to have better results Kalman filter algorithm will be applied to the IMU data and image-matching algorithm will be appropriate for using in semantic object recognition applications.

We have run image-matching algorithm on two operating systems that are Ubuntu and Windows. Although the keypoint numbers and minimum distances are exactly the same, which may be related to the robustness of SURF algorithm for images matching, there are some crucial differences between results. Such as while maximum distance at Windows for greyscale images is 0.990940 and for colored images is 1.003013, maximum distance at Ubuntu for greyscale images is 1.019230 and for colored images is 0.948940. Additionally, process time, which indicates the time, while matching algorithm has run, has passes, at Windows for colored images is 6.566000 seconds, for greyscales images is 6.598000, and at Ubuntu for colored images is 1.640000 seconds, for greyscale images is 1.600000 seconds.

After obtaining these results for each operating system and colored or greyscale options, it is clear that the Ubuntu and Eclipse performance is much better to run image matching and localization algorithm to than Windows and Visual Studio option.

Bibliography

- [1] S. B. Badia, P. Pyk and P. F. M. J. Verschure, "A Biologically Based Flight Control System for a Blimp-based UAV," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, 2005.
- [2] I. F., "Biologically Inspired Visual Odometer for Navigation of a Flying Robot," *Robotics and Autonomous Systems*, vol. 44, pp. 201-208, 2003.
- [3] J. Rao, Z. Gong, J. Luo and S. Xie, "Unmanned Airships for Emergency Management," in *Proceedings of the 2005 IEEE International Workshop on Safety, Security and Rescue Robotics*, Kobe, 2005.
- [4] M. Coleman, D. Rodgers and J. Jones, "A Buoyant Life Investigating Mobile Platform (BLIMP)," *Advances in Space Research*, vol. 38, pp. 1198-1208, 2006.
- [5] M. Onosato, F. Takemura, K. Nonami, K. Kawabata, K. Miura and H. Nakanishi, "Aerial Robots for Quick Information Gathering in USAR," in *SICE-ICASE, 2006. International Joint Conference*, Bexco, Busan, 2006.
- [6] M. Shiho, K. Horioka, G. Inoue, M. Onda, W. C. Leighty, K. Yokoo, S. Ono, K. Ohashi and M. Hirata, "Proposal for Environmental Observation System for Large Scale Gas Pipeline Networks Using Unmanned Airship," *American Institute of Physics Conference Proceeding*, vol. 702, pp. 522-533, 2004.
- [7] L. Merino, F. Caballero, J. R. Martinez-de-Dios, I. Maza and A. Ollero, "An Unmanned Aircraft System for Automatic Forest Fire Monitoring and Measurement," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1-4, pp. 533-548, 2012.
- [8] D. Grossman, "Airship," [Online]. Available: <http://www.airships.net/lz127-graf-zeppelin/history>. [Accessed 26 March 2013].
- [9] D. Grossman, "Airships," [Online]. Available: <http://www.airships.net/hindenburg/lz129-hindenburg-detailed-history>. [Accessed 26 March 2013].
- [10] "Good Year Blimp," 1925. [Online]. Available: <http://www.goodyearblimp.com/cfm/web/blimp/history/>. [Accessed 26 March 2013].
- [11] "World SkyCat," World SkyCat Ltd, 23 July 2000. [Online]. Available: <http://www.worldskycat.com/index.html>. [Accessed 26 March 2013].
- [12] "Skyship," Skyship Cruise, 3 September 1996. [Online]. Available: <http://www.skyship.ch/>. [Accessed 2013 March 26].
- [13] A. Elfes, S. S. Bueno, M. Bergerman, J. J. G. Ramos and S. B. V. Gomes, "Project AURORA: Development of an Autonomous Unmanned Remote Monitoring Robotic Airship," *Journal of the Brazilian Computer Society*, 1998.
- [14] P. S. M. Kungl, D. A. Wimmer and B. H. Kröplin, "Instrumentation of Remote Controlled Airship "Lotte" for In-Flight Measurements," *Aerospace Science and Technology*, vol. 8, pp. 599-610, 2004.
- [15] D. Parry, "United State Naval Research Laboratory," 2 July 1923. [Online]. Available: <http://www.nrl.navy.mil/media/news-releases/2011/navys-modern-airship-receives-historical-identification>. [Accessed 26 March 2013].
- [16] "Northrop Grumman," Northrop Grumman Corporation, [Online]. Available:

<http://www.northropgrumman.com/Capabilities/lemv/Pages/default.aspx>.
[Accessed 26 March 2013].

- [17] T. Fukao, K. Fujitani and T. Kanade, "An Autonomous Blimp for a Surveillance System," in *International Conference on Intelligent Robotics Systems*, Las Vegas, Nevada, 2003.
- [18] L. Merino, F. Caballero, J. R. Martínez-de-Dios, I. Maza and A. Ollero, "An Unmanned Aircraft System for Automatic Forest Fire Monitoring and Measurement," *Journal of Intelligent and Robotic Systems*, vol. 65, no. 1-4, pp. 533-548, 2011.
- [19] L. de Coelho, M. Campos and V. Kumar, "Computer Vision-based Navigation for Autonomous Blimps," in *SIBGRAPHI '98 Proceedings of the International Symposium on Computer Graphics, Image Processing, and Vision*, Washington, DC, 1998.
- [20] J. Müller, A. Rottman, L. M. Reindl and W. Burgard, "A Probabilistic Sonar Sensor Model for Robust Localization of a Small-size Blimp in Indoor Environments using a Particle Filter," in *IEEE international conference on Robotics and Automation*, Piscataway, NJ, 2009.
- [21] T. Fukao, T. Oshibuchi, K. Osuka, T. Kohno and T. Y., "OutdoorBlimpRobotsforRescueSurveillanceSystems," in *SICE Annual Conference*, Tokyo, 2008.
- [22] T. Fukao, K. Fujitani and T. Kanade, "Image-based Tracking Control of a Blimp," in *Conference on Decision and Control*, Hawaii, 2003.
- [23] D. Luu and S. D., "Scene Analysis for Autonomous System Control," in *8th International Symposium on Signal Processing and Its Applications (ISSPA)*, Sydney, 2005.
- [24] Y. Hasa, O. Takizawa, K. Kawabata and H. Kaetsu, "Information Acquisition Using Intelligent Sensor Nodes and an Autonomous Blimp," in *SICE Annual Conference*, Tokyo, 2008.
- [25] J. C. Zufferey, A. Guanella, A. Beyeler and D. Floreano, "Flying over the Reality Gap: From Simulated to Real Indoor Airships," *Autonomous Robots*, vol. 21, no. 3, pp. 243-254, 2006.
- [26] J. Ko, D. Klein, D. Fox and D. Haehnel, "GP-UKF: Unscented Kalman Filters with Gaussian Process Prediction and Observation Models," in *International Conference on Intelligent Robots and Systems*, California, 2007.
- [27] A. Rottman and W. Burgard, "Adaptive Autonomous Control Using Online Value Iteration with Gaussian Processes," in *IEEE International Conference on Robotics and Automation*, Kobe, 2009.
- [28] J. Ko and D. Fox, "GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models," *Autonomous Robots*, vol. 27, no. 1, pp. 75-90, 2009.
- [29] J. Ko, D. Klein, D. Fox and D. Haehnel, "Gaussian Processes and Reinforcement Learning for Identification and Control of an Autonomous Blimp," in *IEEE International Conference on Robotics and Automation*, Washington, 2007.
- [30] A. Rottmann, C. Plagemann, P. Hilgers and W. Burgard, "Autonomous Blimp Control using Model-free Reinforcement Learning in a Continuous State and Action Space," in *IEEE/RSJ INTERNATIONAL CONFERENCE ON*

- [31] H. Fukushima, K. Kon, Y. Hada, F. Matsuno, K. Kawabata and H. Asama, "State-Predictive Control of an Autonomous Blimp in the Presence of Time Delay and Disturbance," in *IEEE International Conference on Control Applications*, Singapore, 2007.
- [32] H. Fukushima, K. Kon, F. Matsuno, Y. Hada, K. Kawabata and H. Asama, "Constrained Model Predictive Control: Applications to Multi-Vehicle Formation and an Autonomous Blimp," in *SICE-ICASE International Joint Conference*, Busan, 2006.
- [33] L. E. Parker, C. M. Reardon, H. Choxi and C. Bolden, "Using Critical Junctures and Environmentally-Dependent Information for Management of Tightly-Coupled Cooperation in Heterogeneous Robot Teams," in *IEEE International Conference on Robotics and Automation*, Kobe, 2009.
- [34] E. A. Bender, *An Introduction to Mathematical Modelling*, New York: Wiley, 1978.
- [35] G. Slabaugh, "Computing Euler Angles from a Rotation Matrix," 1999.
- [36] S. B. V. Gomes and J. J. G. Ramos, "Airship Dynamic Modeling for Autonomous Operation," in *International Conference on Robotics & Automation*, Leuven, 1998.
- [37] H. Lamb, "The Inertia Coefficients of an Ellipsoid Moving in Fluid," British Aeronautical Research Committee Report and Memoranda No. 623, 1918.
- [38] M. M. Munk, "Aerodynamics of Airships," *Aerodynamics Theory*, vol. 6, 1936.
- [39] Y. B. Sebbane, *Lighter Than Air Robots*, Paris: Springer, 2011.
- [40] F. L. Lewis, "Advanced Controls and Sensors Group," 31 October 2008. [Online]. Available: <http://arri.uta.edu/acs/Lectures/lqr>. [Accessed 20 April 2013].
- [41] H. Bay, A. Ess, T. Tuytelaars and L. V. Gool, "SURF: Speeded Up Robust Features," *Computer Vision and Image Understanding (CVIU)*, vol. 110, no. 3, pp. 346-259, 2008.
- [42] P. Chaubey, "Pranjal b.i.t.Wise," 14 April 2011. [Online]. Available: <http://pranjalchaubey.wordpress.com/2011/04/14/razor-imu/>. [Accessed 25 March 2013].
- [43] L. Merino, F. Caballero, J. Martines-de-Dios, I. Maza and A. Ollero, "An Unmanned Aircraft System for Automatic Forest Fire Monitoring and Measurement," *JOURNAL OF INTELLIGENT & ROBOTIC SYSTEMS*, pp. 533-548, 2012.
- [44] S. Oh, S. Kang, K. Lee, S. Ahn and E. Kim, "Flying Display: Autonomous Blimp with Real-Time Visual Tracking and Image Projection," in *International Conference on Intelligent Robots and Systems*, Beijing, 2006.
- [45] S. Lacroix and K. Jung, "High Resolution Terrain Mapping with an Autonomous Blimp," in *International Conference on Intelligent Robots and Systems*, Lausanne, 2002.
- [46] E. King, Y. Kuwata, M. Alighanbari, L. Bertucelli and J. How, "Coordination and Control Experiments on Multi-vehicle Testbed," in *American Control Conference*, Boston, 2004.

APPENDICES

APPENDIX A.1: Technical Specifications of Wireless Camera

APPENDIX A.2: Technical Specifications of Arduino Mega Board

APPENDIX A.3: Technical Specifications of Arduino Uno Board

APPENDIX A.4: Technical Specifications of XBee

APPENDIX A.5: Technical Specifications of Razor 9 DoF IMU

APPENDIX B.1: Matlab Image Grabbing Code

APPENDIX C.1: Transfer Functions of the System

APPENDIX A.1

Technical Parameters of Transmitting Unit

- Video Camera Parts: 1/3" 1/4" Image Sensors
- System: PAL/CCIR NTSC/EIA
- Effective Pixel: PAL: 628X582 NTSC: 510X492
- Image Area: PAL: 5.78X4.19mm NTSC: 4.69X3.45mm
- Horizontal Definition: 380 TV Lines
- Scanning Frequency: PAL/CCIR: 50HZ NTSC/EIA: 60HZ
- Minimum Illumination: 3LUX
- Sensitivity: +18DB-AGL ON-OFF
- Output Electrical Level: 50MW
- Output Frequency: 1.2G/2.4G
- Transmission Signal: Video, Audio
- Linear Transmission Distance: 50-100M
- Voltage: DC+9V
- Current: 300mA
- Power Dissipation: <=640MW

Technical Parameters of Receiving Unit

- Wireless Audio Receiver
- Receiving Method: Electronic Frequency Modulation
- Reception Sensitivity: +18DB
- Receiving Frequency: 1.2G/2.4G
- Receiving Signal: Video, Audio
- Voltage: DC 12V
- Current: 500mA



Figure 10 Wireless Camera Used in This Thesis.

APPENDIX A.2

- Microcontroller : ATmega2560
- Operating Voltage : 5V
- Input Voltage (recommended) : 7-12V
- Input Voltage (limits) : 6-20V
- Digital I/O Pins : 54 (of which 15 provide PWM output)
- Analog Input Pins : 16
- DC Current per I/O Pin : 40 mA
- DC Current for 3.3V Pin : 50 mA
- Flash Memory : 256 KB of which 8 KB used by bootloader
- SRAM : 8 KB
- EEPROM : 4 KB
- Clock Speed : 16 MHz



Figure 11 Arduino Mega 2560 Board Front View.

APPENDIX A.3

- 3.3V @ 40mA
- 250kbps Max data rate
- 2mW output (+3dBm)
- 400ft (120m) range
- Built-in antenna
- Fully FCC certified
- 6 10-bit ADC input pins
- 8 digital IO pins
- 128-bit encryption
- Local or over-air configuration
- AT or API command set



Figure 12 XBee 2mW Chip Antenna - Series 2

APPENDIX A.4

- 9 Degrees of Freedom on a single, flat board:
 - LY530ALH - 300°/s single-axis gyro
 - LPR530ALH - 300°/s dual-axis gyro
 - ADXL345 - 13-bit resolution, $\pm 16g$, triple-axis accelerometer
 - HMC5843 - triple-axis, digital magnetometer
- Outputs of all sensors processed by on-board ATmega328 and sent out via a serial stream
- Autorun feature (hit 'Ctrl-z') and help menu integrated into the example firmware
- Output pins match up with FTDI Basic Breakout, Bluetooth Mate, XBee Explorer
- 3.5-16 VDC input
- ON-OFF control switch and reset switch
- Dimensions: 1.95 x 1.10 " (49.53 x 27.94 mm)



Figure 13 9 Degrees of Freedom - Razor IMU.

APPENDIX B.1

```
% Matlab Code to Grab Images to Create Dataset
clear all
clear all hidden
clc
%% Starting Video Camera
vid = videoinput('winvideo', 1, 'I420_640x480');
set(vid, 'ReturnedColorspace', 'rgb');
src = getselectedsource(vid);
% Configure the object for manual trigger mode.
triggerconfig(vid, 'manual');
vid.FramesPerTrigger = 1;
vid.ROIPosition = [0 85 555 480];
vid.FrameGrabInterval = 1;
start(vid)
pause(0.5)

tic
for n=0:15:360
    frame1=getsnapshot(vid);
    imshow(frame1)
    if n < 10
        imwrite(frame1, ['p1h19500' num2str(n) '.tif']);
        save(['p1h19500' num2str(n) '.mat'], 'frame1')
    end
    if n > 10 && n < 100
        imwrite(frame1, ['p1h1950' num2str(n) '.tif']);
        save(['p1h1950' num2str(n) '.mat'], 'frame1')
    end
    if n > 100
        imwrite(frame1, ['p1h195' num2str(n) '.tif']);
        save(['p1h195' num2str(n) '.mat'], 'frame1')
    end
    pause(0.2)
end
stop(vid)
delete(vid)
```

APPENDIX C.1

Longitudinal Transfer Functions for Unstable System:

Transfer function from input to output...

$$4.443 s^3 + 95.31 s^2 - 75.82 s + 4591$$

$$\#1: \frac{\quad}{s^4 + 31.15 s^3 + 21.57 s^2 + 1720 s - 874.8}$$

$$-1.365 s^3 - 12.89 s^2 - 83.49 s + 453.2$$

$$\#2: \frac{\quad}{s^4 + 31.15 s^3 + 21.57 s^2 + 1720 s - 874.8}$$

$$-10.25 s^3 - 5.507 s^2 + 493.8 s$$

$$\#3: \frac{\quad}{s^4 + 31.15 s^3 + 21.57 s^2 + 1720 s - 874.8}$$

$$-10.25 s^2 - 5.507 s + 493.8$$

$$\#4: \frac{\quad}{s^4 + 31.15 s^3 + 21.57 s^2 + 1720 s - 874.8}$$

Longitudinal Transfer Functions for LQR Controller Applied Situation:

Transfer function from input to output...

$$4.443 s^3 + 95.31 s^2 - 75.82 s + 4591$$

$$\#1: \frac{\quad}{s^4 + 357 s^3 + 8644 s^2 + 3.596e004 s + 3.247e005}$$

$$-1.365 s^3 - 12.89 s^2 - 83.49 s + 453.2$$

$$\#2: \frac{\quad}{s^4 + 357 s^3 + 8644 s^2 + 3.596e004 s + 3.247e005}$$

$$-10.25 s^3 - 5.507 s^2 + 493.8 s$$

$$\#3: \frac{\quad}{s^4 + 357 s^3 + 8644 s^2 + 3.596e004 s + 3.247e005}$$

$$-10.25 s^2 - 5.507 s + 493.8$$

$$\#4: \frac{\quad}{s^4 + 357 s^3 + 8644 s^2 + 3.596e004 s + 3.247e005}$$

Lateral Transfer Functions for Unstable System:

Transfer function from input to output...

$$-0.008083 s^3 - 31.35 s^2 - 85.05 s + 1710$$

#1: -----

$$s^4 + 33.17 s^3 + 281 s^2 + 283.3 s - 2757$$

$$161.5 s^2 + 664.7 s - 3324$$

#2: -----

$$s^4 + 33.17 s^3 + 281 s^2 + 283.3 s - 2757$$

$$35.93 s^3 + 349.1 s^2 + 258.4 s$$

#3: -----

$$s^4 + 33.17 s^3 + 281 s^2 + 283.3 s - 2757$$

$$35.93 s^2 + 349.1 s + 258.4$$

#4: -----

$$s^4 + 33.17 s^3 + 281 s^2 + 283.3 s - 2757$$

Lateral Transfer Functions for LQR Controller Applied Situation:

Transfer function from input to output...

$$-0.008083 s^3 - 31.35 s^2 - 85.05 s + 1710$$

#1: -----

$$s^4 + 57.51 s^3 + 739 s^2 + 3210 s + 4652$$

$$161.5 s^2 + 664.7 s - 3324$$

#2: -----

$$s^4 + 57.51 s^3 + 739 s^2 + 3210 s + 4652$$

$$35.93 s^3 + 349.1 s^2 + 258.4 s$$

#3: -----

$$s^4 + 57.51 s^3 + 739 s^2 + 3210 s + 4652$$

$$35.93 s^2 + 349.1 s + 258.4$$

#4: -----

$$s^4 + 57.51 s^3 + 739 s^2 + 3210 s + 4652$$

CURRICULUM VITAE



Name Surname : Ertuğrul BAYRAKTAR

Place and Date of Birth : ADAPAZARI/1985

Address : Fatih Mh. Saka Sk. No:5 Arifiye/SAKARYA

E-Mail : ertugrulbayraktar@gmail.com

B.Sc. : Yıldız Technical University/Mechanical Engineering

M.Sc. : Kadir Has University/Financial Engineering

Professional Experience and Rewards:

Canovate Electronics R&D Department Engineer (2008-2009)