

Visual Control of an Autonomous Indoor Robotic Blimp

L. M. Alkurdi, R. B. Fisher
University of Edinburgh

ABSTRACT

The problem of visual control of an indoor blimp autonomously is investigated in this chapter. Autonomous aerial vehicles have been an attractive platform for a wide range of applications, especially since they don't have the terrain limitations the autonomous ground vehicles face. They have been used for advertisements, terrain mapping, surveillance and environmental research. Blimps are a special kind of autonomous aerial vehicles; they are wingless and have the ability to hover. This makes them overcome the maneuverability constraints winged aerial vehicles and helicopters suffer from. Our blimp platform also provides an exciting platform for the application and testing of control algorithms. This is because blimps are notorious for the uncertainties within their mathematical model and their susceptibility for environmental disturbances such as wind gusts. We have successfully applied visual control by using a fuzzy logic controller on the robotic blimp to achieve autonomous waypoint tracking.

INTRODUCTION

A blimp is a special kind of lighter-than-air airship; it does not have a rigid skeleton supporting its balloon. Blimp and airship automation has recently emerged as an attractive field of research due to their properties.

Unmanned aerial vehicles in general have advantages over unmanned ground vehicles. They are able to reach locations where it is hard for ground vehicles to reach due to hazards or terrain limitations. They also have the advantage of a larger field of view making them able to survey and collect data of a larger area of terrain at a given instance. Unmanned aerial vehicles are also faster and have better maneuverability.

Blimps also have advantages over winged unmanned aerial vehicles and helicopters. Blimps have much safer failure degradation. They are able to hover over one area for a long time, achieve low altitude flights and do not suffer from maneuverability constraints. They also have minimal vibration and do not influence the environment they are in. The properties previously mentioned make them ideal for data collection, exploration, monitoring and research applications. They take off and land vertically. This means that they can be easily deployed with no need for a runway, which makes them attractive as platforms for rescue operations or as communication beacons when communication is cut-off from a certain area. Other attractive properties include long flight durations and low energy consumption as they depend on buoyancy to achieve vertical position. The blimp's relatively slow speed makes it also an attractive platform for computationally expensive algorithms that need many state updates such as simultaneous localization and mapping (SLAM).

Blimps have been studied as a viable platform for rapidly deployable communication beacons (Flahpour et al., 2009), advertisements and atmospheric data collection and analysis. They are also attractive for military operations such as surveillance and rapid equipment deployment. Blimps serve as an option for providing images and information about regions which have suffered natural catastrophes.

Map building and localization of targets have also been studied through the work of LAAS/CNRS (Hygounenc, Soueres, & Lacroix, 2004). Astro-explorations are also an application studied by the Jet Propulsion laboratory at NASA (Kampke & Elfes, 2003).

Blimp Used

The Surveyor blimp “YARB” (Yet Another Robotic Blimp), which is a 66” helium blimp, was employed in this project. This robotic blimp is driven by three motors, two propellers and a third vectoring motor. The onboard electronics include a Blackfin processor, color camera and a Matchport wireless LAN interface.

A network camera fixed to the ceiling provides the images for the image processing algorithms. The processing is done on a laptop, and the motor commands are sent to the blimp via wireless LAN interface.

The blimp is 1.68m long and has a maximum diameter of 0.76m giving it a fineness ratio (length/diameter) of 2.2. It has a volume of 0.26m^3 and a total lift capacity of 0.3kg given that the lighter than air gas used is helium. While hydrogen is a cheaper alternative that provides more lift capacity for the same volume, helium remains the safer choice.

The blimp platform under study has a few drawbacks making its control rather challenging. The most challenging aspects of the control problem are modeling the dynamics of the blimp and accounting for uncertainties. Examples of uncertainties include disturbances in the form of temperature and pressure variations that could vary the size of the blimp’s envelope and vary the buoyancy, or disturbances such as wind gusts. Another problem faced in this project is that the blimp’s envelope leaks helium varying its buoyancy from one test run to the other. Airship dynamics are also notoriously hard to control due to large moment of inertia (Khoury & Gillett, 2002). Furthermore, the blimp’s lack of an internal rigid frame structure makes its envelope susceptible to expansion and contraction due to acceleration, pressure and temperature variations, adding uncertainty to the blimp’s dynamic model. Signal latency has also been observed in our platform as well as delay in control signals.

Therefore, the blimp is indeed a hard platform to control; and just like any controller, for successful operation, an input of positional state is essential. The states we are interested in regarding blimp control are: position in three dimensional space, vertical and horizontal velocities, angular position and finally angular velocity. The space the blimp flies in is shown in figure 1. In practice this represents the space inside the Informatics Forum at the Edinburgh University.

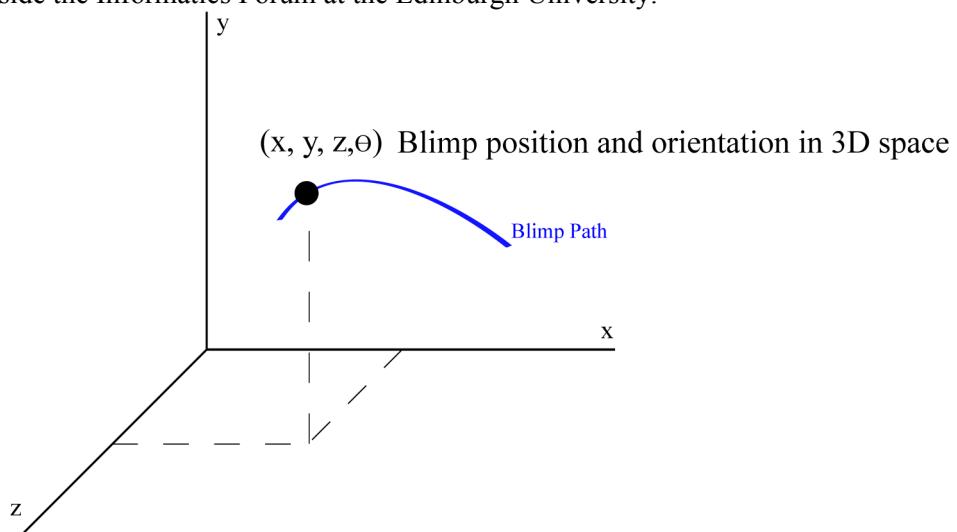


Figure 1. The space blimp flies in.

Ideally, a global positioning system (GPS) would be the most suitable choice for the inputs. However; because the blimp operates indoors, the GPS signal might not be accurate enough or not available. Sensors in the form of accelerometers and inertial navigation systems are also not available to this platform as blimps have the disadvantage of limited payload. Payload, when speaking about blimps, is a function of envelope size, and with small blimps limited payload means limited amount of sensors the blimp can be equipped with. This means that the variety of information that the blimp's controller can be fed is limited.

As such, we employ a ceiling camera looking down on the blimp's operation area to guide the blimp and provide positional state. Figure 2.a shows the image as seen from the ceiling camera and the virtual waypoints and the line that blimp should ideally follow. Figure 2.a also shows sources of wind gust that influence the behavior of the blimp. They are labeled as doors, stairs and elevators. The setup of the camera system is shown in figure 2.b.

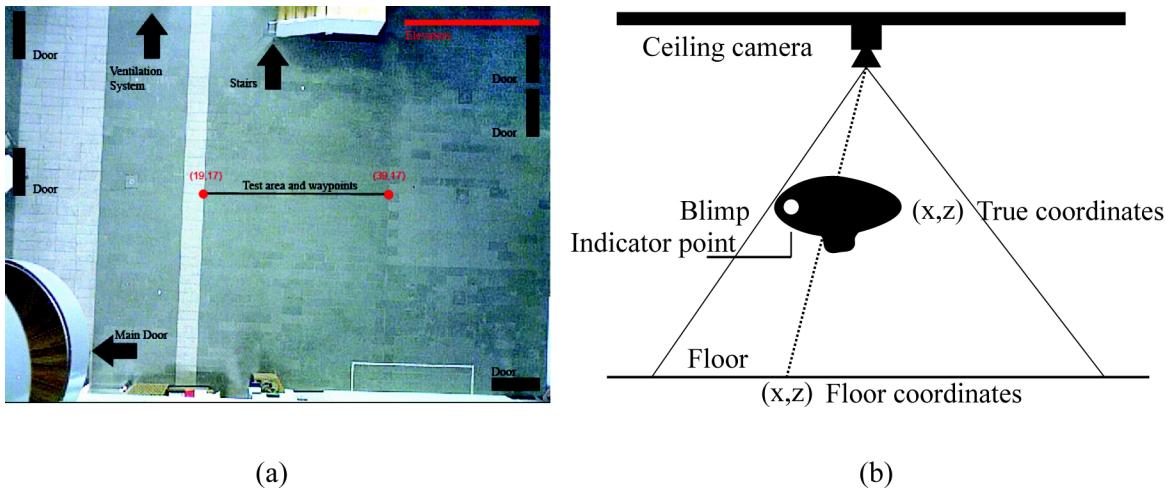


Figure 2. a) An image as seen from the ceiling camera and the virtual waypoints as well as the sources of error. b) The setup of the camera system.

Following the analysis of each image frame captured from the ceiling camera, the absolute parameters of position in 3D space, linear velocity, orientation and angular velocity are computed. They are transformed into relative parameters by calculating the deviation from preset values. The relative positions represent error signals and can be then fed into the fuzzy control algorithm to obtain motor output.

In this chapter we discuss how we have successfully applied a vision system to extract and locate the robotic blimp. Visual servoing was applied to the blimp by way of a fuzzy logic controller to achieve autonomous waypoint tracking. The vision system as well as the fuzzy logic controller were successful at achieving the objective of indoor waypoint controller and proved to be robust against environmental disturbances.

BACKGROUND

This section introduces previous research done on airships and blimps. The first subsection mentions the major airship platforms and the control algorithms used. The second subsection introduces vision algorithms applied for blimp and airship control. The third subsection focuses on fuzzy logic control schemes applied on airships and blimps.

Airships in the literature

This subsection aims to provide a summary of major airship platforms and discusses the control aspects used in each of these projects. Reviews on airship platforms as well as other UAV can be found in (Avenant, 2010), (Liu, Pan, Stirling, & Naghdy, 2009) and (Ollero & Merino, 2004).

The University of Stuttgart's project "Lotte" is a 15m airship, with a volume of 107m³ and has a maximum payload of 12kgs. It has been a platform for many research projects such as aerodynamic research (Lutz, Funk, Jakobi, & Wagner, 2002). The dynamical characteristics of the ship have been modeled using system identification techniques (Kornienko, 2006). The control relies on a number of sensor inputs such as GPS (global positioning system) information for position tracking as well as electronic compasses. The accelerations are calculated using inertial measuring units and the helium temperature and pressure is calculated and compensated against. A full description of the sensors used in this project can be found in (Kungl, Schlenker, Wimmer, & Kröplin, 2004) and a discussion on the controllers used is given in (Wimmer et al., 2002).

The LAAS-CNRS airship "Karma" is 8m, has a volume of 15m³ and a maximum payload of 3.5kg (Hygounenc et al., 2004). This platform had been developed for high resolution terrain mapping and the controllers are built to execute planned trajectories by using the blimp sensor input and detecting special ground elements (Lacroix, Soueres, Hygounenc, & Berry, 2003). Positioning is done through vision, where two successive frames are analyzed to get a position update. The controller assumes decoupling between longitudinal and lateral planes. Once the airship achieves the desired longitudinal position the lateral controller starts to achieve path following. The airship's control algorithm involves geometric and dynamic models whose constraints are taken into account by employing backstepping techniques (Hygounenc & Soueres, 2003). This airship platform has been used to apply SLAM (simultaneous localization and mapping) techniques successfully as discussed in (Hygounenc et al., 2004).

The Titan Aerobot project developed at the NASA and the Jet Propulsion Laboratory at the University of California was proposed to be used for planetary exploration on Titan, one of Saturn's moons (Elfes et al., 2005). The airship is an Airspeed AS-800B, it utilizes a nonlinear airship model for control purposes discussed in (Payne & Joshi, 2004). The controllers were built to accomplish tasks such as loiter, hover and cruise. A special controller is built for subtasks of ascent, descent, turning and altitude control. A full list of controllers that include sequential-loop-closure and linear-quadratic-regulator control algorithms is discussed in (Kulczycki, Joshi, & Hess, 2006).

The AURORA Airship project at the Autonomous Institute of CTI Campinas, Brazil, is another important airship platform that has been used for environmental monitoring missions, investigations of airship dynamic models and visual servo guidance (Azinheira et al., 2002). The control of this airship is discussed in (Ramos et al., 2001) where the airship makes use of a proportional integral (PI) controller and a proportional derivative (PD) controller to follow a path trajectory by outputting a heading angle. The problem of hover control has been also been investigated using this platform and is discussed in (Azinheira, Ramosb, & Buenob, 2000) where image processing is used to provide an offset from the desired position which then is input into the controller to account for the positional deviation.

Blimp Vision Systems

This subsection introduces work done on vision systems used for blimp feedback and control. The vision systems introduced cover both ceiling cameras as well as on board cameras.

Visual feedback control using receding horizon control, also known also model predictive control, applied to unmanned planar blimp system is presented in (Kawai, Hirano, Azuma, & Fujita, 2004). The camera is fixed on a ceiling looking down on the test area and blimp. The camera identifies two characteristic points drawn (black dots) on top of the blimp envelope. These characteristics points have different sizes and can always be seen from the ceiling camera. The position and orientation of the blimp can be deduced by image processing techniques applied to the characteristics points. Visual

feedback control using two different controllers were applied on this platform. The first controller was a receding horizon controller using a control Lyapunov function as terminal cost to stabilize the system. The second controller was a linear parameter using carrying system with a self-scheduling parameter. Results showed that the receding controller provided the better results. The same platform was used in (Kawai, Kitagawa, Izoe, & Fujita, 2003) for PD (proportional derivative) control. A dynamical model is derived. Feedback linearization techniques are applied to achieve a visual linearizing feedback PD controller.

In (van der Zwaan, Bernardino, & Santos-Victor, 2000) an onboard vision system is applied on a small-sized, indoor blimp. Visual control is applied to this platform to achieve station keeping and docking; the main objective here is to maintain the blimp at a certain 3D location and orientation in reference to a specific landmark. Information about the blimp's pose and location is extracted through the vision system; this information is then used to attain visual servoing. The visual algorithm applied makes use of an initial image patch set by the user. Subsequent blimp movements would then distort this reference image. Image registration is then applied through minimization of the error function (sum-of-squared-differences) to calculate optical flow. The measurements obtained by the image processing are then passed to PD controllers so as to achieve the objective of station keeping and docking. The PD parameters were set experimentally. The height of the blimp was maintained by comparing the initial's patch area and the current tracked patch area. Results showed success in tracking the object and achieving station keeping and docking. The system had difficulties with lateral movements of the blimp due to air currents, as this blimp does not have lateral degrees of freedom to control. This required using rotation of the blimp which introduced oscillatory behavior because the rotation of the blimp is not performed around the camera optical axis.

Visual navigation of robotic airships has been investigated in (Xie, Luo, Rao, & Gong, 2007). An onboard camera extracts and tracks natural landmarks (buildings in a city) and uses them as visual beacons for localization and control. The system makes use of geographical information systems (GIS) to extract the geometric information of these extracted visual beacons. An algorithm is then applied to obtain orientation and position of the airship by comparing the given geometric data of the beacons (by the GIS) with those extracted by the vision system. Visual feedback is then passed to an optimal fuzzy flight control system to keep the airship on a predefined track. Genetic algorithms were applied for the optimization of the controller.

In (Azinheira et al., 2002) visual servo control of a hovering outdoor robotic airship is discussed. An onboard camera is used to identify a circle on the ground and a ball floating above it. This represents the hovering location. The circle-ball configuration was used as it has an interesting property of decoupling longitudinal and lateral dynamics which simplifies the design of the controller. Once these landmarks are extracted, their properties are transformed into visual signals. An image Jacobian is built using these visual signals and used for visual servoing. The algorithm includes airship dynamics and uses optimal control design to obtain motor command signals. Experiments have been set up to test environments of no wind disturbances, slight wind gusts and finally strong winds and gusts. The results show that stabilization of the airship was maintained by the optimal controller even with the existence of harsh wind conditions.

Blimp Control

This subsection introduces major work done on proportional integral derivative (PID) and fuzzy controllers applied to blimps. While the main focus will be on projects relating to fuzzy controllers, we will mention ongoing work in the form of intelligent controllers such as model predictive controllers and reinforcement learning controllers that have shown very good results when applied to blimps.

Acquiring the blimp dynamical model is a first step of studying controller design. A general dynamical model for blimps is presented in (Gomes & Ramos, 2008). In this work, a platform for

controller design and simulation research is presented through a complete physical and dynamical model of the blimp.

Classical control methods in the form of PID control have the advantage of simple implementation and reliability, however it can be computationally expensive to model the system and tune its parameters. Work on a PD controller can be seen in (Azinheira et al., 2002). This project employed a dynamical model controlled by a PD error controller that gets feedback from an onboard camera that sends feedback signals to the controller. PID control has been also been applied to landing of a blimp by Toshihiko Takaya in (Tayaka, Minagawa, Yamamoto, & Ohuchi, 2006), using orbital control. Another platform is presented in (Hygounenc et al., 2003) where a PID controller is used for altitude and horizontal positions of the blimp.

In the work of Falahpour et al. in (Falahpour et al., 2009), a fuzzy logic controller was compared with a PID controller. System model and dynamics were derived in order to apply PID control. The model accounted for air friction and random wind gusts. The PID controller was able to achieve the desired position and could cope with gusts of wind of varying direction and force. The fuzzy controller used three membership functions for the inputs and five membership functions for the outputs. There are four error inputs (plane position, orientation and angular speed) giving eighty one control rules. The defuzzification method used was the weighted average method. Results from the comparison showed better performance with the fuzzy controller in terms of less oscillation and faster convergence speed. This result was obtained under MATLAB simulation of the second order balloon dynamic model.

Gonzalez et al. in (Gonzalez, Burgard, Sanz, & Fernandez, 2009) applied a fuzzy altitude controller on a low-cost autonomous indoor blimp and compared it to PID control. Vertical and horizontal controllers were decoupled much like our system. The system also employs a fuzzy collision avoidance controller where a PID collision avoidance controller failed to provide satisfactory results. PID altitude control parameters were experimentally calculated using the Ziegler-Nichols method and showed good results in stable, undisturbed environments but showed large oscillations in environments with disturbances. The fuzzy logic controller has two inputs of velocity and vertical position error and employed five membership functions for each input, actuation output had nine membership functions. The Fuzzy controller out-performed the PID controller in practical tests especially in environments with wind disturbances. The same platform had a fuzzy collision controller that uses five membership functions for velocity and three membership functions for positional error. The fuzzy controller showed the desired behavior while the PID had oscillatory behavior and was judged to be inadequate.

In (Jian-guo & Jun, 2008) altitude control of an autonomous airship is investigated with the use of fuzzy logic. Seven triangular membership functions were used for the positional error as well as for the speed of the blimp. The controller has two fuzzy logic subsystems, one that calculates the current error and the other calculates the predicted error. Each one is activated depending on the blimp's altitude. This compound controller was designed in this way to be robust to disturbances. The results showed that the blimp was able to achieve and maintain the required altitude as well as being robust to parametric perturbations and disturbances.

Backstepping control, model-predictive control and reinforcement learning control of autonomous blimp navigation are prominent control methodologies currently receiving much attention. Important reviews and introductory material to the field for control of autonomous airships is given in (Liu et al., 2009) and (Ollero & Merino, 2004).

BLIMP EXTRACTION

In this section we discuss the methods investigated to detect the pixels that belong to the blimp object. These pixels are an important region of interest (ROI) for the extraction of positional states. Our setup has the advantage of a fixed camera with a fixed viewpoint. Ideally, we would like to subtract the current frame from the fixed viewpoint to detect changes in the blimp's position. However; several problems

have to be dealt with. These problems come in the form of dynamically changing ambient light, varying sun positions, varying sunlight intensity, the cloudy nature of the city of Edinburgh, shadows and finally dynamically changing reflections and glares that are produced on the background. Other problems the detector has to deal with are external objects being introduced into the background such as people, furniture and stationary objects. More serious problems come in the form of white tables that can be easily be mistaken for the blimp.

Two detection methods will be discussed in this section. First we will investigate detection with background subtraction using principle component analysis (PCA) techniques as discussed in (Majecka, 2009). The second method involves using colored thresholding as discussed in (Ntelidakis, 2010).

Blimp detection: background modeling using PCA.

In (Majecka, 2009), different methods were used to model the background image seen from the ceiling camera; these methods are discussed in this subsection.

The first method of modeling the background image was achieved by calculating the mean of fifty images that were obtained in different lighting conditions.

The second method applied the idea of constantly updating the mean background model with the most recently captured image. A weighted sum is used to combine these two images. This idea proved to be robust to changes in overall lighting conditions. However; as this method relies on weights, it can be difficult to optimally fine tune, and achieve the correct results.

The third technique used chromaticity coordinates to represent the mean image. This image would then be subtracted from a current image also represented by its chromaticity coordinates. A dynamically chosen threshold was then applied to detect the blimp. This technique proved robust against reflections, however; it failed in the case of shadows.

Finally background modeling using PCA techniques was applied. This method has been most successful in extracting objects from the background. It is discussed with more detail in the following paragraphs.

PCA was used to model the background image as seen from the ceiling camera in the Informatics Forum. Fifty images were obtained in different lighting conditions. This image-set contained images with glares, light reflections, shadows and all major disturbances that can occur in the background image, examples are shown in figure 3.a and 3.b. A background model was produced using PCA from these fifty images. The background model is composed of the mean image as well as the eigenvectors that represent the significant principal components (directions of the most variation in the image), these were retrieved by way of eigendecomposition. To extract newly introduced objects to the background in any new image, this new image is projected onto the space characterized by the most significant eigenvectors and then projected into the original image space. This projected image is then compared to the original image. Foreground objects resulting from the previous operation would represent objects alien to the background. They can be extracted using binary thresholding.

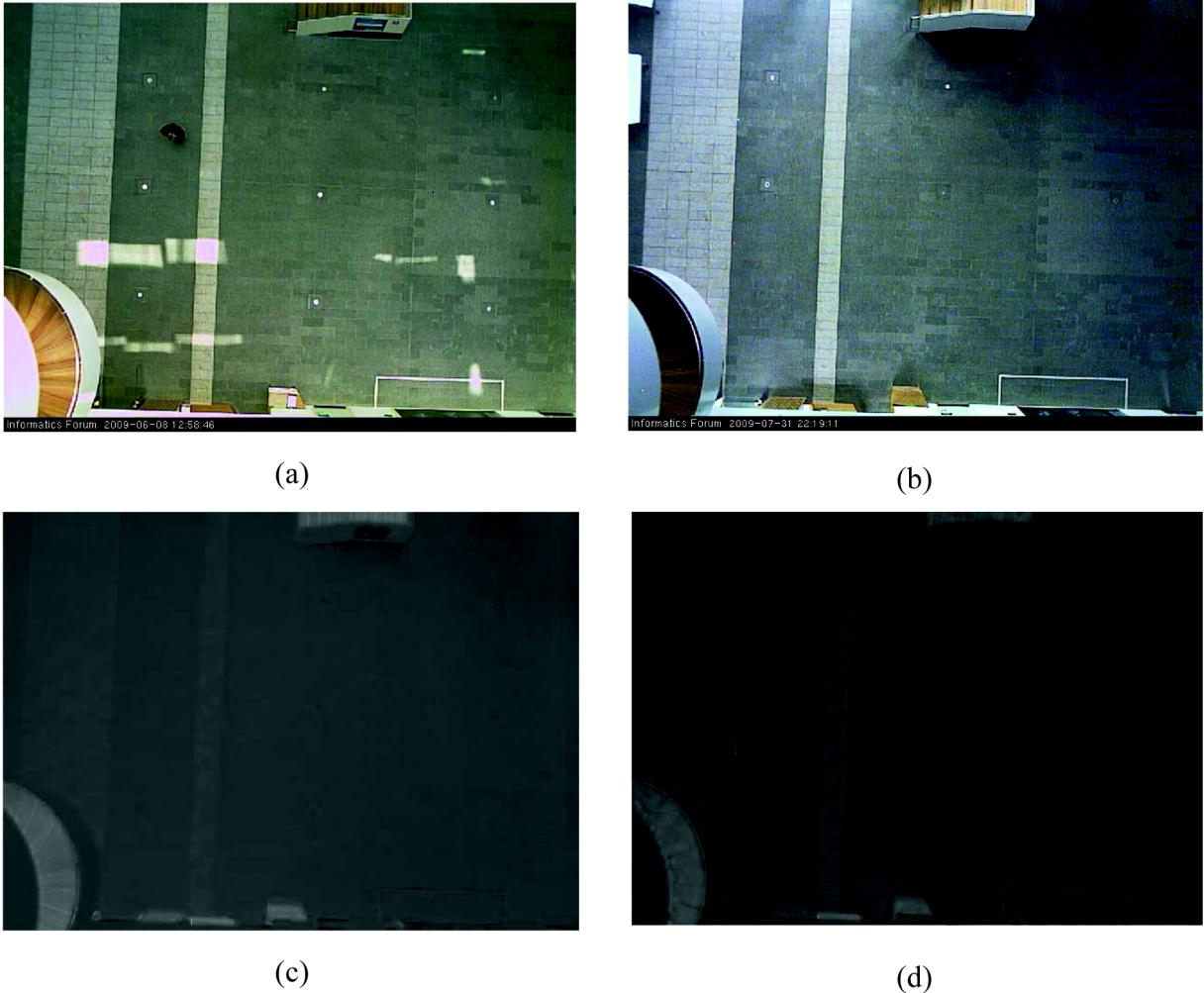


Figure 3. a) Reflections on the background image. b) Shadows and lighting variations on the background image. c) Mean background image. d) First principal component. Figure copied from (Majecka, 2009) by permission of author.

To extract the blimp, we only used the first eigenimage \mathbf{F} of the Principle Component Model and the mean image seen by the ceiling camera \mathbf{M} . The projection of the current image \mathbf{C} onto the first eigenimage would then be $\mathbf{P} = (\mathbf{C} - \mathbf{M}) * \mathbf{F}$. This projection is then used to estimate the background by the following equation $\mathbf{B} = \mathbf{P} * \mathbf{F} + \mathbf{M}$. Detection can then be done by applying $\mathbf{O} = \text{abs}(\mathbf{C} - \mathbf{B})$. Finally, a binary threshold is applied on \mathbf{O} to extract the blimp blob.

The blob (blimp) is next used as a mask on the current frame to obtain a ROI image of the blimp itself. Next, a new histogram is built of this ROI to find a threshold to extract the indicator point (black marker) on the white blimp. That marker is then used to obtain the orientation of the blimp by applying the *arctan* function to the center of gravity of the blimp and the indicator point's center of gravity. The steps of the algorithm are presented next:

- Obtain grayscale image of the current frame.
- Apply the PCA method on the current to obtain the difference image (DI).
- Build the DI's histogram.

- Apply smoothing to the histogram and find the optimum threshold value (valley between two peaks).
- Threshold the DI image to obtain a binary image.
- Apply morphological open operation to remove noise and small artifacts.
- Label remaining blobs.
- Calculate the area of the blobs and assume the largest to be the blimp.
- Create a mask from the identified blimp blob.
- Fill all gaps in the mask.
- Create an inverted image of the original current frame, and apply the mask to it to obtain an ROI consisting of the blimp.
- Build the ROI's histogram.
- Apply smoothing to the histogram and find the optimum threshold (valley between two peaks) value for the ROI. This value will separate the blimp from its marker point.
- Apply thresholding to obtain binary image, foreground will represent the marker.
- Apply morphological open operation to remove noise.
- Label remaining blobs.
- Calculate the area of the blobs and assume the largest to be the blimp's marker.

This algorithm proved very fast, however; it still suffered from failures under certain lighting conditions, and whenever the blimp flew over white patches of the floor. Thus, there was a need to devise a different algorithm for the detection of the blimp.

Blimp detection: using color threshold.

This algorithm that was adapted from (Ntelidakis, 2010) makes use of the fact that the blimp is the whitest object in the image frame. As such, this approach becomes robust to reflections from the sun or sudden changes in ambient light brightness. This is because these changes are never bright enough or intense enough to become as intense as the white blimp. However; other white objects in the background are extracted, therefore a mask is applied to mask out the stairs and the visible desk area.

The steps of this algorithm are presented in the following points:

- Obtain image of the current frame. This is shown in figure 4.a.
- Blur the current frame to remove false positives.
- Convert the image to grayscale. This is shown in figure 4.b.
- Use a high threshold (200) to transform the grayscale image into a binary image. This makes sure that only the whitest objects are seen as white pixels while others become black pixels in the binary image. This step is shown in figure 4.c.
- Apply morphological operation of erosion to remove salt noise.
- Apply morphological operation of dilation; this is to repair any damage done by the erosion in the previous step. This step is important because we want to maintain the pixels that represent the blimp marker, as its detection is important to calculate the orientation of the blimp. The result is shown in figure 4.d.
- Apply the mask to remove the stairs area in the background as it is too white, as shown in figure 4.e.
- Label the blobs and calculate their sizes.
- Pick the largest blob and assume it is the blimp.
- Give elliptical properties to the blimp and find its center of mass, minor and major axes and size.
- Use the blimp as an ROI, as shown in figure 4.f.
- Find the biggest blobs inside that ROI.
- Pick the blob with the smallest compactness.

- Find its center of mass.
- Estimate the blimp orientation based on this center of mass, and the center of mass of the blimp.

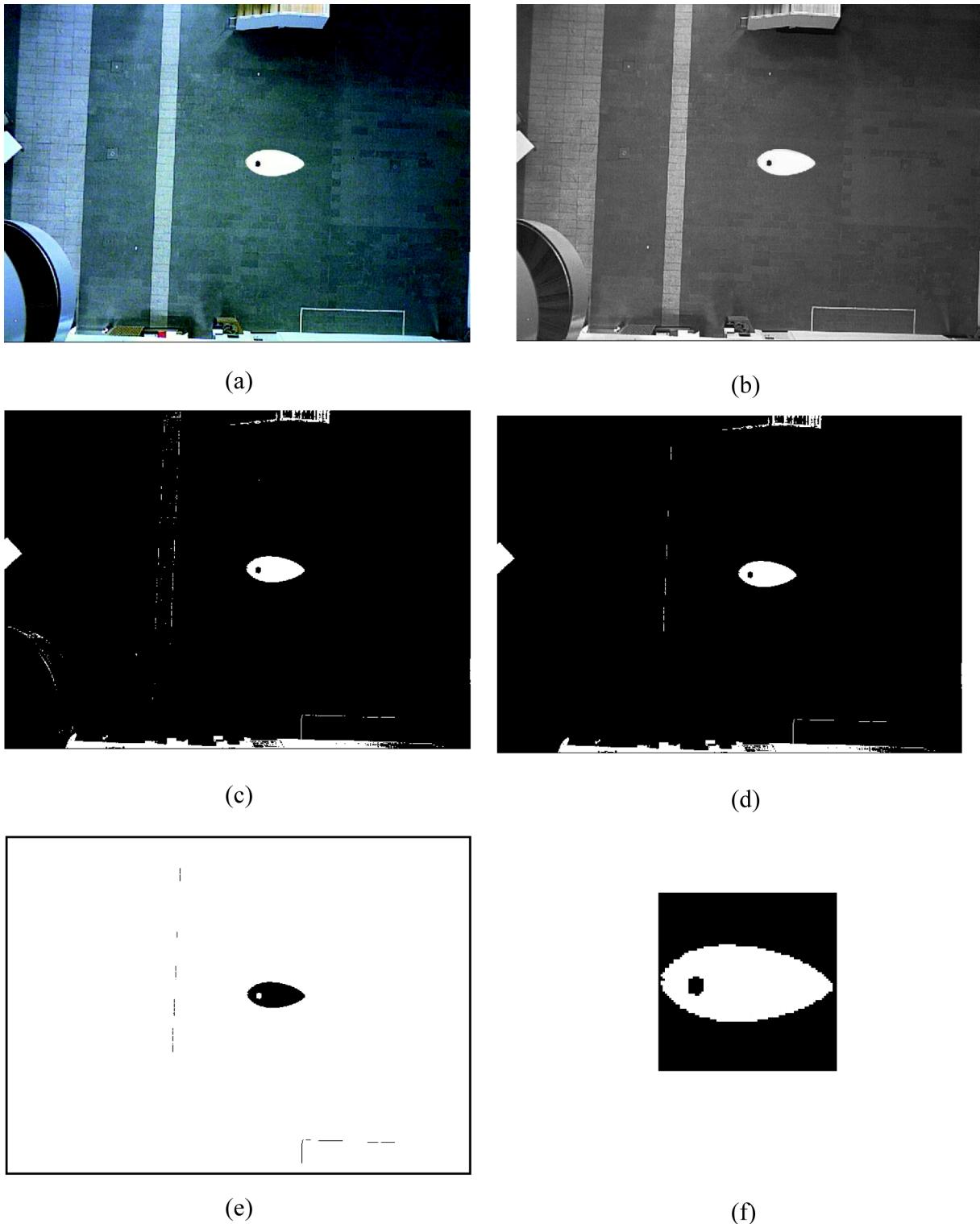


Figure 4. a) Current image. b) Image blurred and converted to greyscale. c) Binary image (threshold =200). d) Binary image after erosion and dilation. e) Masked image. f) ROI for black marker identification. Figure adapted from (Ntelidakis, 2010).

This method then returns the important values of blimp's center of mass coordinates, the blimp's orientation, major and minor axes of the elliptical shape fitted to the blimp and finally the area of the blimp as seen from the ceiling camera.

This method proved to be robust to shadows and the sun's reflection and other failures experienced in the first algorithm, however, it performs very poorly in low ambient light conditions. A smaller threshold can be applied to solve that problem. The second method also proved faster than the first taking only 0.3 seconds compared to 0.5 seconds.

Now that we have this information, we can calculate the blimp's position in 3D space. The next section discusses the methods used to calculate these parameters.

BLIMP'S VISUAL ODOMETRY

In this section we discuss the calculations involved in transforming the pixel information obtained from the image frame obtained to real word information; vital to the controller performance (Ntelidakis, 2010). In the last section we have obtained:

- \mathbf{x}_t : Planar position (x_{im}, z_{im}) in the current image frame t in pixels.
- θ : Angular orientation of the blimp (yaw) in radians.
- R_{im} : The length of the major axis of the blimp (from the fitted ellipsoid).
- A : Area of the blimp in pixels.

The information we aim to obtain is the blimp's position in three dimensional space, vertical and horizontal velocities, angular orientation (yaw) and finally angular velocity. We are only interested in the 4 degrees of freedom (3D position and angular orientation (yaw)). We assume that the roll and pitch parameters of the blimp are always maintained due to the blimp's nature. In all the calculations discussed in this section the distortion of the ceiling camera is neglected.

The following subsections discuss the calculation of altitude, planar position, planar velocity, and vertical velocity.

Calculating altitude, position and velocities

To calculate the location of the blimp, we make use of consecutive frames of the extracted blimp as seen in figure 5.a. We also make use of the fact that the blimp is flying inside an irregular pyramid with a height of 77.58 feet; this is the camera's height from the floor. The irregular pyramid's base is a rectangle 52x39 feet-squared. This is shown in figure 5.b. Estimating the blimp's planar position and height is very important in this case; this is because the blimp changes planes inside this pyramid as it flies inside it. Each of these planes has different widths and lengths. After we calculate the blimp's position inside the irregular pyramid, we apply calculations to project the blimp's position onto the ground coordinates. These positions are used to calculate planar and vertical velocities.

Figure 5. a) Binary image of the extracted blimp. b) The irregular pyramid that the blimp is flying inside.

VISUAL SERVOING CONTROLLER

Now that we have the inputs required for the controller function we can input them into a controller to issue the appropriate motor commands for waypoint tracking. As discussed earlier, the blimp is a challenging platform to control. Moreover; it is being constantly attacked by several environmental disturbances. In this section we discuss the controllers used in our project and introduce the methodology used to compare between them (Alkurdi, 2011). First we discuss the experimental setup used in our project.

Experimental Setup

We used waypoint tracking as an experiment for measuring the performance of the controllers applied to the blimp. The setup can be seen in figure 2.a. The blimp is expected to travel between the two red dots in a straight line. A way to measure the actual performance is shown in figure 6 below. As the blimp flies in 3D space, we calculate the (x,y,z) coordinates through the vision system and store them at each time step. We calculate the perpendicular distance of each data point from the line it should be following. This is shown in figure 6.a. Over all the time steps we would obtain a surface in 3D space. This surface is shown in figure 6.b. Ideally, for perfect control, we would like to minimize this surface area to zero. However, this is not possible because of a set of inaccuracies within the blimp model and external disturbances. The following subsection discusses these errors.

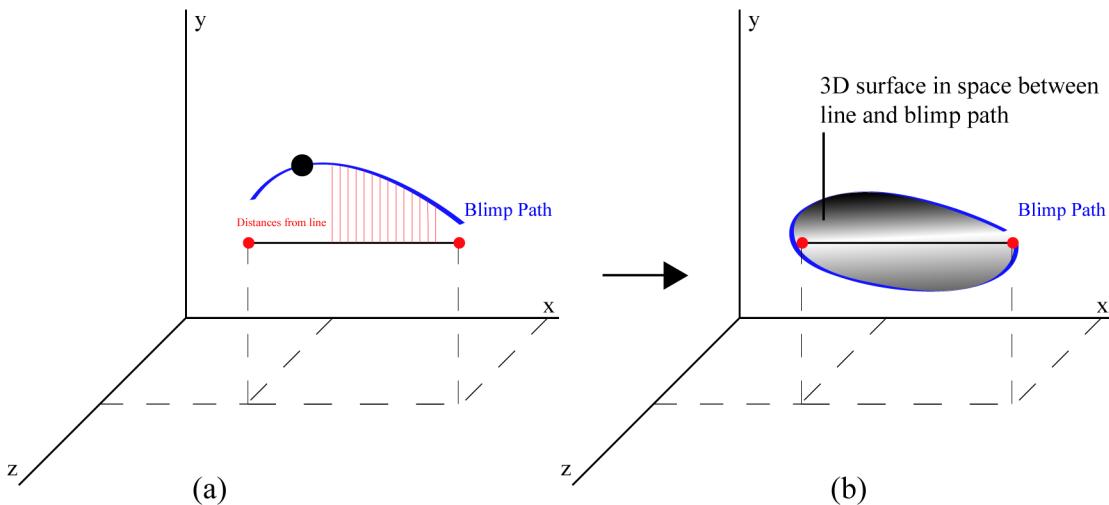


Figure 6. a) The perpendicular distance of each data point from the line it should be following. b) The surface obtained in 3D space.

Sources of Error

The blimp does not have a pressure sensor to calculate the contained helium pressure, so each day of testing the blimp will contain a different amount of helium. The gondola's position also differs at the beginning of each testing day. This has the effect of changing the center of mass for the blimp and could alter the performance slightly.

During the test run, factors such as people passing, elevators running, and doors opening in the Informatics Forum ground level will induce gusts that drive the blimp in a certain direction, pushing it away off its course.

The blimp has a lot of inertia when moving in a certain direction, thus changing the direction requires some time. It then becomes important that no series of faulty motor commands are sent to the blimp, as this will cause the blimp to stray from its path and it requires more time to set it back onto its course.

The software for the blimp calculates the area of the blimp as seen from a network camera fixed at the ceiling of the Informatics Forum looking down on the testing area. The area is calculated to get a height estimation, and since this is done every frame, a lot of noise is introduced into the recorded height of the blimp, as lighting conditions might change from one captured frame to another.

The blimp is held by two ropes at each end of its envelope. The amount of rope length given to the blimp affects the weight the blimp is carrying and thus would affect its final height. The amount of rope length can vary throughout the test run, and can affect the results.

Other issues that affect the blimp during its run is change of temperature in the testing area. This will lead to a change in pressure and change of the height of the blimp.

These errors are expected to be dealt with via the controller function such that a correct motor command is sent to the blimp to overcome these disturbances and set the blimp back to its correct path. The controllers applied to our robotic blimp are discussed in the next subsection.

Blimp Control

Three different linguistic controllers were applied to our blimp platform. The aim was to avoid computationally expensive mathematical algorithms for the realization of real time control. The first of these controllers is a 3-parameter state based controller. The second controller studied was a 4-parameter state based controller. Finally a fuzzy logic controller was studied.

The 3-parameter state based controller determines the current, predicted and the rate of change of the blimp's position for position control and it also calculates the current, predicted and the rate of change of the blimp's orientation for heading control. If the three parameters of say, position, are in a certain state, then a certain hardcoded motor impulse command is applied in the current time step. For more information please refer to (Alkurdi, 2011).

The 4-parameter state based controller is basically the same. However; an additional parameter of history is added. The idea was to study the effect of past state to further enhance the blimp's understanding of its position.

The fuzzy logic controller was introduced to study the effect of having a continuous range of motor commands issued from the controller; rather than having hardcoded impulse commands as in the previous two controllers. The fuzzy logic control system employed two sub controllers, one for heading and one for position. The heading controller has an input parameter of angle that uses 7 triangular membership functions, and a parameter of angular velocity that uses 3 triangular membership functions. The output used 5 triangular membership functions. The position controller has an input parameter of distance from goal that uses 5 triangular membership functions, and an input parameter of planar velocity that uses 3 triangular membership functions. The output used 3 triangular membership functions. The rule base is determined through our experience of the blimp's performance. Full design parameters are discussed in (Alkurdi, 2011).

Results

The fuzzy logic controller outperformed the other two controllers. Many test runs were conducted to test the performance of each controller and on average the flying area in 3D was smaller for fuzzy logic (area was 1923 feet²) than the other two controllers (the area was 2714 feet² and 2300 feet² for the 3 and 4 parameter controller respectively). In terms of time, the fuzzy logic controller (3.35 minutes) performed its task the fastest (*versus* 3.55 and 4.27 minutes for the 3 and 4 parameter controller respectively).

FUTURE RESEARCH DIRECTIONS

In this section future work on the vision, control and mathematical modeling is discussed. In terms of vision, the blimp platform carries an on-board camera that could be used for SLAM applications. Work on mono-SLAM techniques can be found in (Davison, Reid, Molton, & Stasse, 2007). The application of such algorithms can complement or replace the current localization algorithms.

As for the control algorithms, further tuning to the fuzzy controller would enhance the performance of the robotic blimp. This could be done manually by changing the number and/or shape of the membership functions. The ranges the membership functions cover can also be manipulated to tune

the fuzzy logic controller. The rule base can always be changed to alter the behavior of the controller and is an important factor in tuning the controller. Finally, the defuzzification method can be changed, where the center of area method that is used throughout this project provides a weighted average output to the motors, a more/less aggressive method could be used. Some other defuzzification methods that can be tested are: Mean of maxima, leftmost maximum and rightmost maximum. Automatic tuning; however, can be done in the form of ANFIS (adaptive neural fuzzy inference system) (Kitt, Chekima, & Dhargam, 2011).

Further work can also be done through obtaining a mathematical model of the robotic blimp and applying computer simulations to understand the best way to apply control algorithms on this platform.

CONCLUSIONS

The research showed that the visual feedback was effective for blimp control even under unpredictable indoor conditions that included varying lighting conditions and gusts arising from door openings, people walking and elevators. Additional research showed that the fuzzy logic controller was more effective than a 3-parameter state table approach.

A typical path by the blimp is a run between the two waypoints; they are shown as circles in figure 7.a. A typical recorded path in 3D space is shown in figure 7.a, the red dots represent the outbound path, the blue dots represent the inbound path. Figure 7.b shows the average path of 20 runs of the fuzzy logic controlled blimp. These figures show the successes of the image processing and visual control techniques and the range of their coverage.

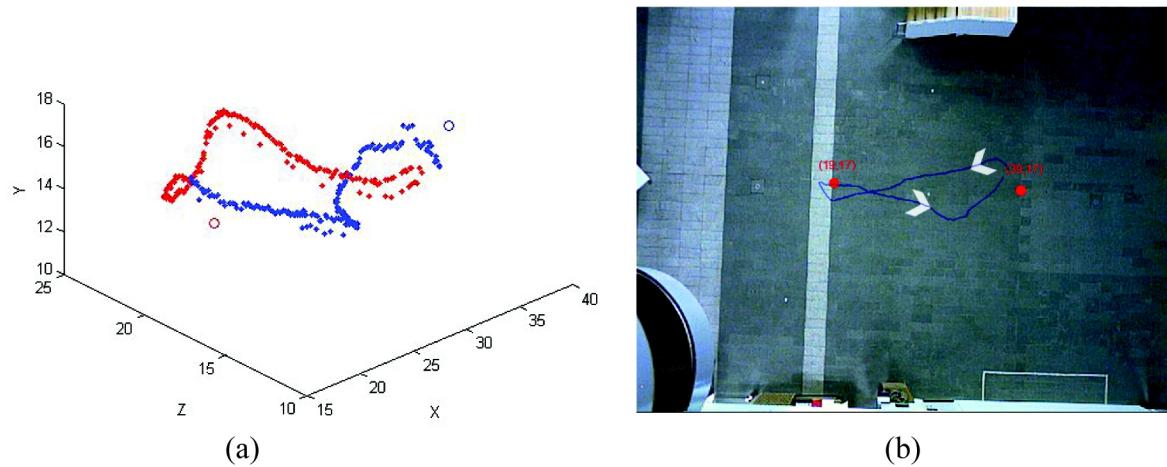


Figure 7. a) A typical recorded path in 3D. b) Average path of the blimp's path under fuzzy control.

REFERENCES

- Alkurdi, L. (2011). *Fuzzy logic control of a robotic blimp*. Master's thesis, University of Edinburgh, U.K.
- Avenant, G. C. (2010). *Autonomous flight control system for an airship*. Master's thesis, Stellenbosch University, South Africa.
- Azinheira, J.R., de Paivab, E.C., Ramosb, J.J.G., & Buenob, S.S. (2000). Hovering control of an autonomous unmanned airship. In *proceedings of the 4th Portuguese Conference on Automatic Control*, (pp.430–435).

- Azinheira, J. R., Rives, P., Carvalho, J. R. H., Silveira, G. F., de Paiva, E. C., & Bueno, S. S. (2002). Visual servo control for the hovering of an outdoor robotic airship. In *proceedings of the IEEE International Conference on Robotics and Automation ICRA '02, Vol. 3*, (pp. 2787–2792).
- Davison, A.J., Reid, I.D., Molton, N.D., & Stasse, O. (2007). MonoSLAM: Real-Time Single Camera SLAM. *Pattern Analysis and Machine Intelligence, 29* (6), 1052 – 1067.
- Elfes, A., Montgomery, J.F., Hall, J.L., Joshi, S.S., Payne, J., & Bergh, C.F. (2005, September). *Autonomous flight control for a titan exploration aerobot*. Paper presented at the 8th International Symposium on Artificial Intelligence, Robotics and Automation in Space. Munich, Germany.
- Falahpour, M., Moradi, H., Refai, H., Atiquzzaman, M., Rubin, R., & Lo Presti, P. G. (2009, October). *Performance comparison of classic and fuzzy logic controllers for communication airships*. Paper presented at the IEEE/AIAA 28th Digital Avionics Systems Conference DASC '09, Orlando, FL, U.S.A.
- Gomes, S.B.V., & Ramos, J. G., Jr. (1998). Airship dynamic modeling for autonomous operation. In *proceedings of the IEEE International Robotics and Automation Conference, Vol. 4*, (pp. 3462–3467). Leuven, Belgium.
- Gonzalez, P., Burgard, W., Sanz, R., & Fernandez, J.L. (2009). Developing a low-cost autonomous indoor blimp. *Journal of Physical Agents, 3*(1), 43– 52.
- Hygounenc, E., & Soueres, P. (2003). Lateral path following GPS-based control of a small-size unmanned blimp. In *Proceedings of the IEEE International Conference of Robotics and Automation ICRA '03, Vol. 1*, (pp. 540–545).
- Hygounenc, E., Jung, I., Soueres, P., & Lacroix, S. (2004). The autonomous blimp project of LAAS-CNRS: Achievements in flight control and terrain mapping. In *The International Journal of Robotics Research, 23*, 473–511.
- Jian-guo, G., & Jun, Z. (2008, December). *Altitude control system of autonomous airship based on fuzzy logic*. Paper presented at the 2nd International Symposium on Systems and Control in Aerospace and Astronautics, Shenzhen, China.
- Kampke, T., & Elfes, A. (2003). Optimal aerobot trajectory planning for wind-based opportunistic flight control. *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS 2003): Vol. 1*, (pp. 67–74).
- Khoury, G. A., & Gillett, J. D. (2002). *Airship Technology*. Cambridge, England : Cambridge University.
- Kawai, Y., Hirano, H., Azuma, T., & Fujita, M. (2004). Visual feedback control of an unmanned planar blimp system with self-scheduling parameter via receding horizon approach. In *proceedings of the IEEE international conference on Control Applications, Vol. 2*, (pp. 1063-1069).
- Kawai, Y., Kitagawa, S., Izoe, S., & Fujita, M. (2003, August). *An unmanned planar blimp on visual feedback control: experimental results*. Paper presented at the SICE 2003 Annual Conference, Fukui, Japan.

- Kitt, W.W., Chekima, A., & Dhargam, J.A. (2011). Design and verification of an ANFIS based control system for an autonomous airship on a simulation environment. *GJCAT*, 1, 119–128.
- Kornienko. A. (2006). *System Identification Approach for Determining Flight Dynamical Characteristics of an Airship from Flight Data*. PhD thesis, University of Stuttgart, Germany.
- Kulczycki, E.A., Joshi, S.S., & Hess, R.A. (2006). Towards controller design for autonomous airships using slc and lqr methods. In AIAA Guidance Navigation and Control Conference and Exhibit.
- Kungl, P., Schlenker, M., Wimmer, D.A., & Kröplin, B.H. (2004). Instrumentation of remote controlled airship "LOTTE" for in-flight measurements. *Aerospace Science and Technology*, 8(7), 599–610.
- Lacroix, S., Jung, I-K., Soueres, P., Hygounenc, E., & Berry, J-P. (2003). The autonomous blimp project of LASS/CNRS: Current status. *Experimental Robotics VIII*, 5(2003), 487–496.
- Liu, Y., Pan, Z., Stirling, D., & Naghdy, F. (2009). Control of autonomous airship. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics* (pp. 2457–2462).
- Lutz, T., Funk, P., Jakobi, A., & Wagner, S. (2002, July). *Summary of Aerodynamic Studies on the LOTTE Airship*. Paper presented at the 4th International Airship Convention and Exhibition. Cambridge, England, U.K.
- Majecka, B. (2009). *Statistical models of pedestrian behavior in the Forum*. Master's thesis, University of Edinburgh, U.K.
- Ntelidakis, A. (2010). *Using a blimp to build an interior model of the forum*. Master's thesis, University of Edinburgh, U.K.
- Ollero, A., & Merino, L. (2004). Control and perception techniques for aerial robotics. *Annual Reviews in Control*, 28(2), 167–178.
- Payne, J., & Joshi, S.S. (2004). 6 degree-of-freedom non-linear robotic airship model for autonomous control. Unpublished technical report, University of California. Oakland, California, U.S.
- Ramos, J.J.G., de Paiva, E.C., Azinheira, J.R., Bueno, S.S., Bergermana, M., Ferreira, P.A.V., & Carvalho, J.R.H. (2001). Lateral/directional control for an autonomous unmanned airship. *Aircraft Engineering and Aerospace Technology*, 73(5), 453–458.
- Takaya, T., Kawamura, H., Minagawa, Y., Yamamoto, M., & Ohuchi, A. (2006). PID landing orbit motion controller for an indoor blimp robot. *Artificial Life and Robotics*, 10(2), 177–184.
- Van der Zwaan, S., Bernardino, A., & Santos-Victor, J. (2000). Vision based station keeping and docking for an aerial blimp. In *proceedings of the International Conference on Intelligent Robots and Systems*, Vol. 1, (pp. 614-619). Takamatsu, Japan.
- Wimmer, D., Bildstein, M., Well, K.H., Schlenker, M., Kungl, P., & Kröplin, B-H. (2002, October) *Research airship "lotte" development and operation of controllers for autonomous flight phases*. Paper presented at the International Conference on Intelligent Robots and Systems. Lausanne, Switzerland.

Xie, S-R., Luo, J., Rao, J-J., & Gong, Z-B. (2007). Computer Vision-based Navigation and Predefined Track Following Control of a Small Robotic Airship. *ACTA AUTOMATICA SINICA*, 33(3), 286-291.

ADDITIONAL READING SECTION

Alkurdi, L. (2011). *Fuzzy logic control of a robotic blimp*. Master's thesis, University of Edinburgh, U.K.

Avenant, G. C. (2010). *Autonomous flight control system for an airship*. Master's thesis, Stellenbosch University, South Africa.

Bekiroglu, K. (2010). Vision based control of an autonomous blimp with actuator saturation using pulse width modulation. Master's thesis, Northeastern University, U.S.A.

Metelo, F. M., & Campos, L. R. G. *Vision Based control of an Autonomous Blimp (VIDEOBLIMP)*. (2003). Undergraduate final year project, Universidade Técnica de Lisboa, Portugal.

Fukao, T., Fujitani, K., & Kanade, T. (2003). Image-based Tracking Control of a Blimp . In *proceedings of the 42nd IEEE Conference on Decision and Control*, Vol. 5, (pp. 5414-4519). Maui, Hawaii, U.S.A.

Liu,Y., Pan,Z., Stirling,D., & Naghdy.F. (2009). Control of autonomous airship. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics* (pp. 2457–2462).

Moutinho, A. (2007). *Modeling and nonlinear control for airship autonomous flight*. PhD thesis, Universidade Técnica de Lisboa, Portugal.

Ollero,A., & Merino, L. (2004). Control and perception techniques for aerial robotics. *Annual Reviews in Control*, 28(2), 167–178.

KEY TERMS & DEFINITIONS

Blimp: A lighter-than-air airship that does not have an exterior frame to hold its envelope.

Visual servoing: Applying vision sensors and algorithms to extract feedback parameters necessary for robotic motion control.

Visual odometry: Extracting the robot's pose and location by the use of vision sensors.

Fuzzy logic control: Applying fuzzy logic mathematics to control a given system. This ultimately leads to a control system that uses a linguistic framework that an experienced user can understand, rather than a mathematical framework of transfer functions.

Waypoint tracking: A robot is said to have an objective of waypoint tracking if its goal is to achieve sequential tracking of specific points in 3D space.

Object detection: In image processing, object detection is the operation of deciding whether a desired object is in a given image (and often also includes location of the object or extracting the pixels that belong to the desired object).

Thresholding: In image processing, thresholding is applying a pixel intensity value(s) to discriminate foreground objects from background objects (e.g. pixels brighter than the threshold belong to the foreground object).