A simple approach on implementing IMU sensor fusion in PID controller for stabilizing quadrotor flight control

Article · March 2011

DOI: 10.1109/CSPA.2011.5759837

2 authors:

A. Zul Azfar
Universiti Malaysia Perlis

Hazry Desa
Universiti Malaysia Perlis

A Simple Approach on Implementing IMU Sensor Fusion in PID Controller for Stabilizing Quadrotor Flight Control

A. Zul Azfar[1], D. Hazry[2]

Autonomous System and Machine Vision (AutoMAV) Research Cluster,
Universiti Malaysia Perlis, Taman Pertiwi Indah,
Jalan Kangar-Alor Setar, Seriab,
01000 Kangar, Perlis, Malaysia.
[1]zulazfar.automav@gmail.com
[2]hazry@unimap.edu.my

***Abstract*—This paper discussed about a simple approach for implementing well-known PID controller in microcontroller for quadrotor project. This controller will be used for stabilizing the quadrotor while in flight. Towards to stabilize this type of air craft, sensors must be used as the feedback for PID controller. A new method in sensor fusion and inclination angle estimation is discussed in this paper. A simple system architecture for controlling four motors also included as a reference for the flow of the programming sequence. Single axis test has been done with result to prove this method and discussed in this paper.**

***Keywords*—quadrotor, PID controller, sensor fusion**

I. Introduction

Recently, quadrotor is a popular project being developed by researchers as in [1], [2], [3], [4] and also remote control air craft hobbyist. It is already developed by researchers for any area of application with different approach of control system and design. Quadrotor capabilities already known by its maneuverability, simple design and can be used for various application. However, this air craft is not very stable in flight. To achieve a stable flight, a control system must be developed to control the speed of the propeller. Some sensors must be used for this approach. To get a better value for feedback or estimation, some researchers used to combine the sensors in sensor fusion method. Then, this sensor value will be inserted into equation which is embedded in the microcontroller to control the stability of this air craft in flight.

II. Quadrotor

Quadrotor is an aircraft with vertical or short take-off and landing (V/STOL) capabilities compare to fixed wing air craft. Because of this advantage, it can be deployed almost anywhere on land. There is more advantage of quadrotor compared to traditional helicopter which is reducing the complexity of the mechanism which called ‘swash plate’. Quadrotor just using two sets of counter-rotating fixed pitch propellers as stated in [2] which can be replaced easily and with low cost. Because of its maneuverability and speed, it can be said that, quadrotor is a hybrid of fixed-wing and helicopter. Quadrotor can move fast as fixed-wing with the movement of helicopter which can take-off, hover, roll, pitch and yaw. The movement for this special air craft is achieved by manipulating the lift and thrust from each propeller to produce some movement such as take-off and landing, roll, pitch and yaw.

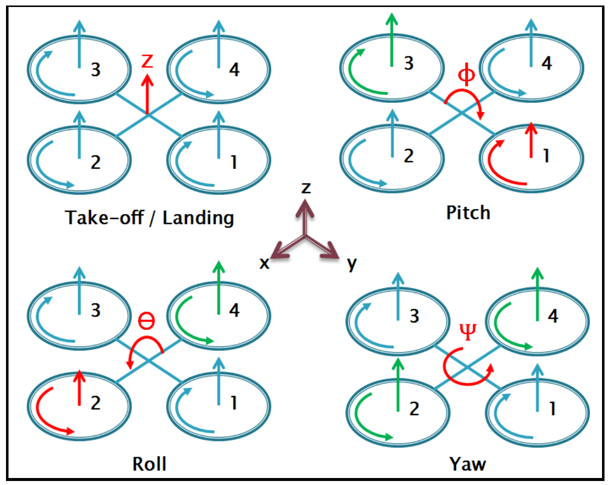

Fig. 1: Quadrotor Flight Control Movements

Fig. 1 shows the illustration for quadrotor flight control movement. Take-off or landing movement is controlled by increasing or decreasing all four motors, Assuming motor 1 is front of the air craft, pitch movement which is in y-axis is controlled by varying speed of motor 1 and 3. For roll movement which is in x-axis is controlled by varying speed of motor 2 and 4. For Yaw movement which is in z-axis is controlled by varying the speed of pair propeller in opposite axis. This control also explained in [5], [6], and [7].

III. System Architecture

System architecture is the heart of any design development. A simple architecture is enough to control the quadrotor flight operation stably. This system architecture can be changed or added to meet various applications. The system architecture can be more complex. There are many type of system architecture for quadrotor, for examples as in [8].

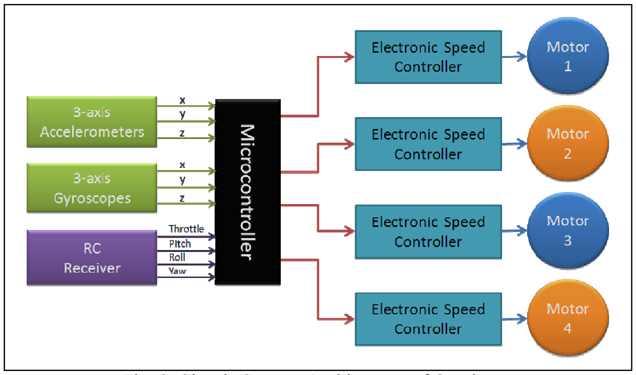

Fig. 2: Simple System Architecture of Quadrotor

Fig. 2 shows the simple system architecture of quadrotor. The structure of the architecture is simple for radio controlled quadrotor. The RC receiver is the radio signal receiver from standard RC transmitter. Microcontroller will capture the width of the signal received as the input control from the operator. The minimum number of channels must be captured to control the quadrotor is four channel, which are throttle, roll, pitch and yaw. Some transmitter have up to eight channel which can be used to control other movement or operation on quadrotor such as move on-board camera angle, capture image from on-board camera, to activate autonomous control or others.

The microcontroller also will read analog signal from accelerometers and gyroscopes. This signal will be converted into digital to be processed by microcontroller. The digital data then will be combined to measure the inclination of the quadrotor for pitch and roll and also measure the angular rate for yaw. All this information from sensor must be accurate to prevent miscalculation that will affect the stability in quadrotor flight. So, a suitable filter or estimation algorithm is needed to reduce the noise from sensor to get a near- real value. Then, this value will be used as feedback in PID control system that will be discussed later.

The microcontroller also has to produce four pulse width modulations (PWM) to control the brushless motor speed. The electronic speed controller (ESC) will receive the PWM from microcontroller and will control the speed of the brushless motor. Each brushless motor will be controlled by one ESC. The ESC used in this architecture is the ordinary ESC that being used by hobbyist to control their brushless motor speed.

All of this task will be performed by a microcontroller. A suitable microcontroller must be chosen so it will process those entire tasks in specific time. Which mean it must have high speed processing to do all of the tasks. Usually the system runs at 50 Hz rate because the ESC being used only receive servo PWM signal which have 20 ms time interval. So, update rate of the brushless motor only can be done after 20 ms. A project in [9] stated their quadrotor ran in 1 kHz update rate. The system is using i2c interface to the brushless speed controller. Instead of high processing speed, the update rate is very high. So the quadrotor will have a better response and very stable in flight.

IV. Sensors Fusion and Filtering

Sensor fusion is a common method to increase the accuracy and remove noise in sensory system. Just using single type of sensor, the output maybe just accurate but it not noise proof. Whenever noise occurred, the sensor output will be not accurate or changing rapidly. Nowadays, sensors manufacturer embed low pass filter to their sensor. So, it will reduce the noise due to environment such as vibration, electric spark or others. However, just using a low pass filter will not gain accuracy from the sensor. So, to get better output, sensor fusion is the easy way to do. Besides, sensor filtering also can get a better output.

Actually, these kinds of methods are using the estimation algorithm to approach an approximate value. The sensor fusion is already used widely in project such as in [10]-[13]. In this paper, a simple estimation algorithm using sensor fusion will be elaborate in step by step. This algorithm also known as simplified Kalman Filter. This algorithm is introduced by [14]. Thanks to Starlino because it will make this project become simpler and easy. The algorithm is very simple and can be easily coded in C language to be embedded in the microcontroller. The sensors are being used is inertial measurement unit (IMU) which are combination of accelerometers and gyroscopes.

A. *Accelerometer*

Accelerometer is a sensor to measure acceleration and also can be used to measure tilting by measuring the inertial force due to gravity applied in x, y and z axis. Although this sensor can be used to measure inclination of a body, but it is sensitive to vibration. The value will changed rapidly and the exact value of inclination cannot be determined properly.

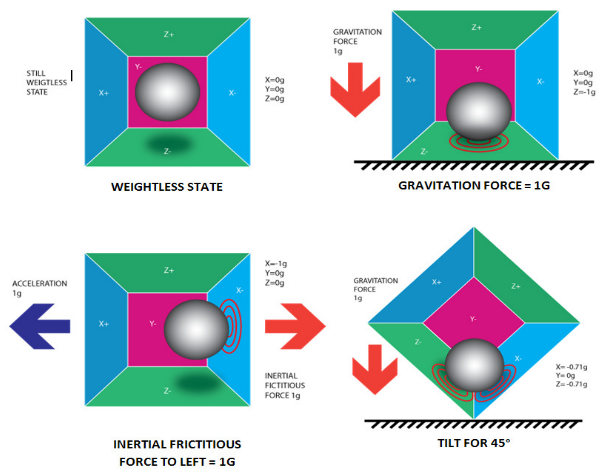

Fig. 3: Illustration of Measuring Inclination Due to Inertial Force.

Fig. 3 shows the illustration of measuring inclination due to inertial force. A ball placed in a box representing the effect of inertial force. The walls are representing axis where the inertial force will be applied. In weightless or floating condition, no force applied on any walls. So, the values of inertial force for all axes are zero. When a body placed on earth, actually gravity force applied on the bottom wall (negative z axis) in scale of 1G which is in acceleration value of 9.81 ms^{-2}. So the value for z-axis is equal to -1G and the rest is zero. If a force of 1G is applied to the left (positive x axis), the force actually effect on the opposite site of the wall where the force is applied. So the value for x axis is -1G and the rest is zero. If the body tilts for 45°, the ball will touch the negative x and negative z axis on the bottom due to gravity. So the value for both axes are equal to SQRT (1/2) or -0.71G and y axis equal to zero. How this value is calculated, it will be described in the next model representation for accelerometer.

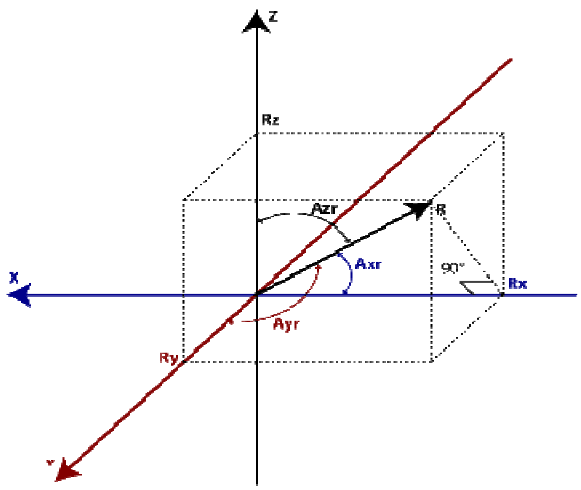

Fig. 4: Model Reference for Accelerometer.

Fig. 4 shows the model reference for accelerometer which useful in understanding how the accelerometer interacts with outside forces; it is more practical to perform calculations if the coordinate system is fixed to the axes. The vector R is the force vector that accelerometer is measuring while Rx, Ry, Rz are projection of the R vector on the X, Y and Z axes. The relationship between R, Rx, Ry, and Rz are:

$$R^2 = Rx^2 + Ry^2 + Rz^2 \tag{1}$$

The equation is equivalent to Theorem Pythagoras in 3D. Rx, Ry and Rz actually represent the value of measurement accelerometer in G for x, y and z axis respectively. The angle of tilting of each axis can be calculated by using equation:

$$Axr = \arccos\left(\frac{Rx}{R}\right) \tag{2}$$

$$Ayr = \arccos\left(\frac{Ry}{R}\right) \tag{3}$$

$$Azr = \arccos\left(\frac{Rz}{R}\right) \tag{4}$$

B. Gyroscope

Gyroscope is a sensor to measure angular rate in an axis. The unit sense by the sensor is in deg/s. This sensor can be used to measure the rotation rate of the body. The angle is measured in degree which can be used to measure inclination rate in 1 second. However, this sensor is less sensitive to linear movement such as vibration but it has a problem which is drift. Drift is the value not becoming to zero when the rotation stops.

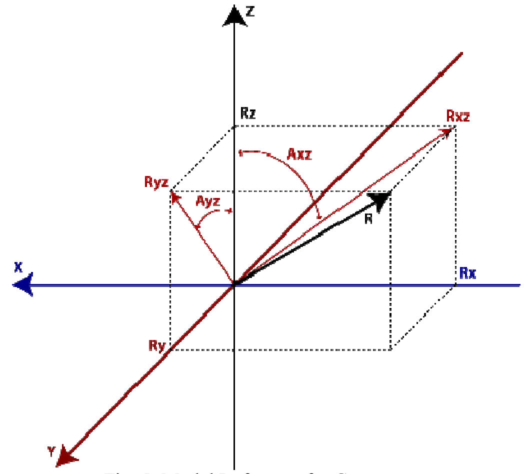

Fig. 5: Model Reference for Gyroscope.

Fig. 5 shows the model reference for gyroscope. This model is taken from accelerometer model to show what gyroscope measure from this model. Rxz is the projection of the inertial force vector R on the XZ plane while Ryz is the projection of the inertial force vector R on the YZ plane. Axz is the angle between the Rxz (projection of R on XZ plane) and Z axis while Ayz is the angle between the Rxz (projection of R on YZ plane) and Z axis. This vector will formed triangle shape and the angle between them can be calculated using trigonometry. From this model, the angle of rotation rate can be calculated by using:

$$Axz = \arctan2(Rx, Rz) \tag{5}$$

$$Ayz = \arctan2(Ry, Rz) \tag{6}$$

Arctan2 is similar to arctan except it returns values in range of (-PI, PI) as opposed to (-PI/2, PI/2) as returned by atan, and it takes 2 arguments instead of one. It allows the conversion of two values of Rx,Rz to angles in the full range of 360 degrees (-PI to PI). Remember that gyroscopes measure the rate of change of angle, so the value has to be times by the time taken for measuring the degree of rotation. The equation as follows where n is the present measurement and T is time taken in measuring the rotation angle:

$$Axz(n) = Axz(n-1) + Rate\ Axz(n) * T \tag{7}$$

$$Ayz(n) = Ayz(n-1) + Rate\ Ayz(n) * T \tag{8}$$

From the [14], the equation can be simplified so it is easy to code in C programming and embeds to the microcontroller. The simplified equations become:

$$RxGyro = \frac{\sin(Axz(n))}{\sqrt{\left(1 + \cos(Axz(n))^2 * 2 * \tan(Ayz(n))^2\right)}} \tag{9}$$

$$RyGyro = \frac{\sin(Ayz(n))}{\sqrt{\left(1 + \cos(Ayz(n))^2 * 2 * \tan(Axz(n))^2\right)}} \tag{10}$$

$$RzGyro = RzGyro * \sqrt{(1 - RxGyro - RyGyro)} \quad (11)$$

C. Accelerometer and Gyroscope (IMU) Sensor Fusion

There are two types of IMU sensor that being used, which are digital and analog. In this paper, the discussion more to analog typed IMU sensor. The analog sensor will give voltage level to microcontroller to represent the value being measured. So microcontroller will convert the value to digital using ADC module so it is can be computable by microcontroller. The value from accelerometers will be converted into G while the value from gyroscopes will be converted into deg/s. Then, the G value for accelerometers will be computed using equation (1)-(4) to get value in degree. Value from gyroscopes will be also converted to degree by using equation (9)-(11).

Using value from accelerometers and gyroscopes, this value will be combined to get estimation for inclination of the quadrotor body. The combination will be using weighted average value. This algorithm is similar to Kalman Filter which is using weight to estimate the result. The main difference of this algorithm from Kalman Filter is that this weight is relatively fixed, whereas in Kalman Filter the weights are permanently updated based on the measured noise of the accelerometer readings. Kalman Filter is focused at giving "the best" theoretical results, whereas this algorithm can give a "good enough" results for this project. The combination equations with weight added are:

$$RxEst(n) = \frac{(RxAcc + RxGyro * wGyro)}{(1 + wGyro)} \quad (12)$$

$$RyEst(n) = \frac{(RyAcc + RyGyro * wGyro)}{(1 + wGyro)} \quad (13)$$

$$RzEst(n) = \frac{(RzAcc + RzGyro * wGyro)}{(1 + wGyro)} \quad (14)$$

From equation (12)-(14), the weighted is given to gyroscope. In this algorithm, gyroscope is being trusted than the accelerometer. The RxAcc, RyAcc and RzAcc are the tilting angle in each axis. This value can be calculated using equation (2)-(4). This algorithm can be implemented in microcontroller by adjusting wGyro depending on measured noise factors, but fixed values will work well for most applications.

V. Control System

PID controller is a well known controller that being used in various application. In this paper, a simple PID controller block diagram is introduced. The estimation inclination angle of quadrotor will be used as feedback to the PID controller. PID controller will be used to stabilize the quadrotor in flight. Whenever the quadrotor is not stable, the PID will determine how much to increase or decrease the motor speed so it can be stable again.

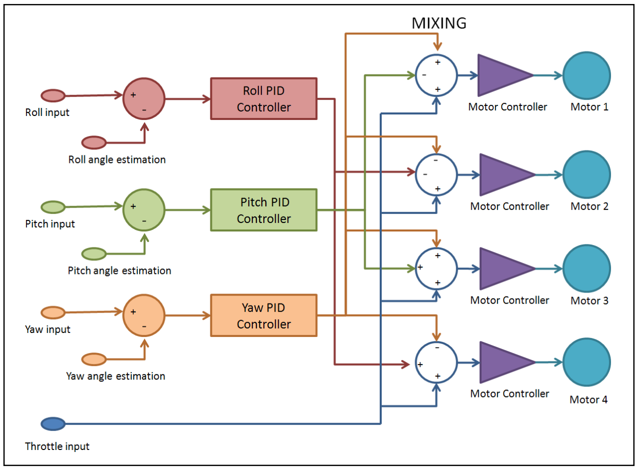

Fig. 6: PID Controller Block Diagram

Fig. 6 shows the PID controller block diagram. The input of the controller is the desired angle to achieve and the feedback is from estimation angle. The output of the PID controller is the PWM to control the brushless motor. The mixing is to control the movement of the quadrotor. This mixing concept is taken from control input from various papers which are doing quadrotor project such as [1], [5], [7], [15] and [16]. The control inputs are similarly wrote as:

$$U_1 = (F_1 + F_2 + F_3 + F_4) \quad (15)$$

$$U_2 = (F_4 - F_2) \quad (16)$$

$$U_3 = (F_3 - F_1) \quad (17)$$

$$U_4 = (F_1 + F_3 - F_2 - F_4) \quad (18)$$

Where U_1 is take-off, U_2 is pitch, U_3 is roll and U_4 is yaw. F is representing motor thrust. All equation in this paper is can be coded in C programming.

VI. Testing and Result

Using the information from part IV and V, the coding in C language is done and programmed into a microcontroller. Test was done to prove by using simple approach is enough to get a stable flight for quadrotor project.

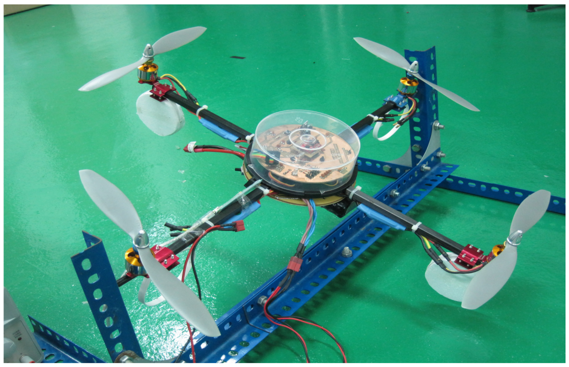
Fig. 7:Quadrotor Testing Setup for Single Axis

Fig. 7 shows the quadrotor testing setup for single axis. The quadrotor is mounted on free friction support in single axis and the test only applied on another axis. Which mean the testing was done on single axis in a time whether pitch or roll.

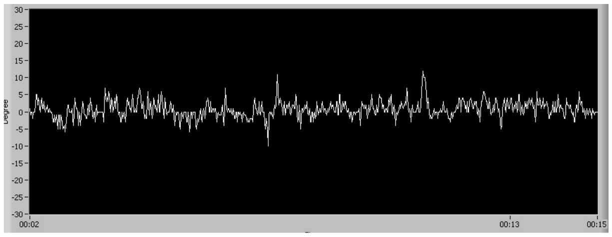

Fig. 8: Test Result for Roll Angle

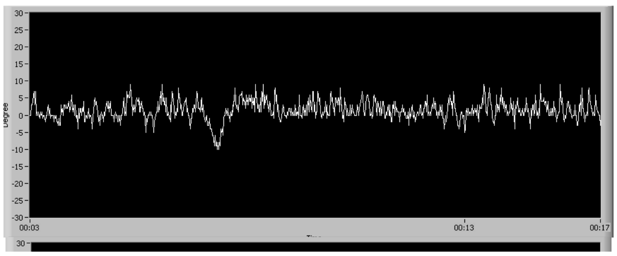

Fig. 9: Test Result for Pitch Angle

Fig. 8 shows the test result for roll angle while Fig. 9 shows the result for pitch angle. It seems as a good result for stabilizing system. But, the PID controller still can be tuned to get a better result. The yaw test not be conducted yet because of the test jig is for pitch and roll.

VII. Future Work

This approach is a good idea and method as a beginner to a project using IMU sensor. This algorithm can be altered to meet the system requirement. For future work, the controller maybe can be combined with other controller such as Fuzzy Logic. A work about hybrid controller is in progress to get a better result. The controller also can be change using other controller such as Neural Network and others. Using various types of controllers can be a good way to compare which one is the best.

Acknowledgment

A special thanks to Associate Professor Dr. Hazry Desa for his advices and guidance in finishing this paper. Thanks also to AutoMAV team for supports and teamwork. A great appreciation to AutoMAV research cluster for the equipment and facilities provided in completing this project.

References

[1] Samir Bouabdallah, Pierpaolo Murrieri and Roland Siegwart, “Design and Control of an Indoor Micro Quadrotor”, Robotics and Automation, 2004.

[2] Osamah A. Rawashdeh, Hong Chul Yang, Rami D. AbouSleiman, and Belal H. Sababha, “Microraptor: A Low-Cost Autonomous Quadrotor System”, ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 2009.

[3] Min, B. C., Cho, C. H., Choi, K. M. and Kim, D. H., “Development of a Micro Quad-Rotor UAV for Monitoring an Indoor Environment”, Proceedings of the FIRA RoboWorld Congress 2009 on Advances in Robotics, 2009.

[4] Spencer G. Fowers, Dah-Jye Lee, Beau J. Tippetts, Kirt D. Lillywhite, Aaron W. Dennis, and James K. Archibald, “Vision Aided Stabilization and the Development of a Quad-Rotor Micro UAV”, Proceedings of the 2007 IEEE International Symposium on Computational Intelligence in Robotics and Automation, 2007.

[5] Samir Bouabdallah, Andr´e Noth and Roland Siegwart, “PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor”, Intelligent Robots and Systems, 2004.

[6] Kong Wai Weng, Mohamad Shukri b. Zainal Abidin, “Design and Control of a Quad-Rotor Flying Robot For Aerial Surveillance”, 4th Student Conference on Research and Development, 2006.

[7] Jack F. Shepherd III and Kagan Tumer, “Robust Neuro-Control for A Micro Quadrotor” , Proceedings of the 12th annual conference on Genetic and evolutionary computation, 2010.

[8] Alexandros Soumelidis, P´eter G´asp´ar, Gergely Regula, B´ela Lantos, “Control of an experimental mini quad-rotor UAV”, 16th Mediterranean Conference on Control and Automation Congress Centre, Ajaccio, France,2008.

[9] Daniel Gurdan, Jan Stumpf, Michael Achtelik, Klaus-Michael Doth, Gerd Hirzinger and Daniela Rus, “Energy-efficient Autonomous Four-rotor Flying Robot Controlled at 1 kHz”, IEEE International Conference on Robotics and Automation, 2007.

[10] Antal Turóczi, “Flight Control System of an Experimental Unmanned Quad-Rotor Helicopter”, 10th International Symposium of Hungarian Researchers on Computational Intelligence and Informatics, 2009.

[11] Philippe Martin and Erwan Sala¨un, “The True Role of Accelerometer Feedback in Quadrotor Control”, 2010 IEEE International Conference on Robotics and Automation, 2010.

[12] Tianguang Zhang, Wei Li, Markus Achtelik, Kolja K¨uhnlenz and Martin Buss, “Multi-Sensory Motion Estimation and Control of a Mini-Quadrotor in an Air-Ground Multi-Robot System”, 2009 IEEE International Conference on Robotics and Biomimetics, 2009.

[13] Sven Lange, Niko S¨underhauf and Peter Protzel, “A Vision Based Onboard Approach for Landing and Position Control of an Autonomous Multirotor UAV in GPS-Denied Environments”, International Conference on Advanced Robotics, 2009.

[14] Starlino (2010) A Guide To using IMU (Accelerometer and Gyroscope Devices) in Embedded Applications.[online]. Available:, http://www.starlino.com/imu_guide.html.

[15] [11] Bora Erginer and Erdinç Altuğ, “Modeling and PD Control of a Quadrotor VTOL Vehicle”, Proceedings of the 2007 IEEE Intelligent Vehicles Symposium, 2007.

[16] Erdinc Altu, James P. Ostrowski and Robert Mahony, “Control of a Quadrotor Helicopter Using Visual Feedback”, Proceedings of the IEEE International Conference on Robotics and Automation, 2002.