International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2454-9150 Vol-04, Issue-04, July 2018

Comparative Study on Positioning of Robot on The Basis of IMU and Camera Module

Sreelekha S, M-Tech ECE Dept., Mar Athanasius College of Engg., Kothamangalam, India, sreelekha1994@gmail.com

Lija Thomas, Assistant Professor ECE Dept., Mar Athanasius College of Engg., Kothamangalam, India,lijasoman@gmail.com

Abstract: An autonomous mobile robot needs to have the ability to position itself in an unknown environment in order to accomplish a set of objectives and improve the navigational performance. The robot positioning denotes the potential to establish current orientation and position of the robot within the reference frame. In this study, we are developing an autonomous robotic vehicle using LPC1769 ARM board with software implemented on LPCXpresso and OpenCV. Two different type localization methods are implemented here. First one using the integration of a low cost Inertial Measurement Unit (IMU) and the second one using camera module with image processing techniques for positioning. In both cases, the position of robot is calculated and the results are compared.

Keywords: Autonomous mobile robot, Positioning, LPC1769 ARM Board, Inertial Measurement Unit, Camera module

I. Introduction

In past few years, the technique of localization of mobile robot has been made considerable attention of researchers. Robot localization, that is the process of estimation of the current position of the robot within the frame of reference, has become an important step in the field of mobile robotics. There are different types of localization methods are available for positioning of mobile robots. Most of the robots positioning techniques are based on the Global Positioning System (GPS), odometers, Inertial Measurement Units (IMU), laser range finders, vision based systems…etc.[1]. These methods involve determining positions and paths of mobile robot in real-time i.e. the robot is its working condition, with the help of off-board and on-board sensors its position is determined.

Generally, common robotic navigation algorithms are depends on encoder based odometry and GPS. In case of outdoor localization, inaccurate odometric readings and GPS limitations will cause large errors. Unpredictable external influences and interruption of signal may cause difficulties in outdoor localization and navigation with GPS. An independent navigation algorithm is developed with the help of IMU, optical navigation sensor, GPS [2]. Mohammad Emal explains the possibilities of inertial navigation system in the field of mobile robotics and their analysis. Such type of control system let us to find the real position of the robot.

This study focusing on IMU and Camera based localization techniques. Inertial Measurement Unit (IMU) typically contain a 3-axis accelerometer and a 3- axis gyroscope in order to measure linear acceleration and angular velocity respectively. It is possible to find the orientation and current position of a device, by processing the signals obtained from these sensors [4]. Inertial sensor based navigation techniques are widely used in aircrafts, UAVs, missiles, spacecraft and submarines.

In camera based navigation technique, image information from a camera module is used to find out the real world position of mobile robot. Here, a completely autonomous mobile robotic platform is developed and tracked its position with the help of these two techniques and compared there performance.

II. Methodology

The development of a robotic vehicle with necessary sensors was the very first step in this work. Position of the robotic vehicle is determined by using IMU sensor and image retrieved from a camera module.

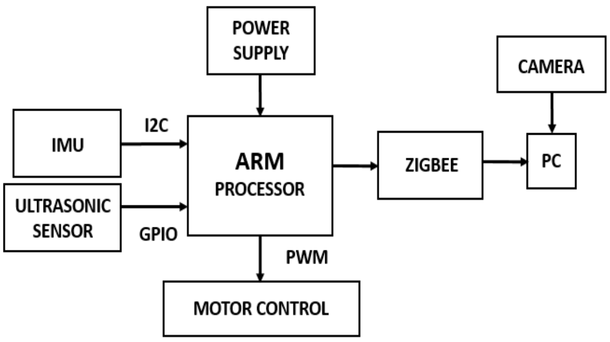


Fig.1. Block diagram of proposed system

Fig.1 shows the block diagram of the proposed system. The set up consists of a robotic vehicle, central PC and camera

 DOI : 10.18231/2454-9150.2018.0500

module. The entire structure of the robot is equipped with DC motors, wheels, castor wheels, LPC1769 board, MPU6050 Inertial Measurement Unit, Ultasonic sensor and CC2500 ZigBee module.

MPU6050 IMU sensor housing consists of two different sensors. A tri-axis accelerometer that generates 3 digital signals corresponding to the linear acceleration about X, Y, Z axes of the device. Similarly, a tri-axis gyroscope that generate angular rate of the vehicle along X, Y, Z axes. In this study, we are considering accelerometer data only. After sensor calibration, double integration of acceleration over time will give the displacement. I^2C interface for the MPU6050 was used to acquire the accelerometric and gyroscopic values from the IMU. In the I^2C interface LPC1769 microcontroller act as the master and MPU6050 as the slave. Raw values are read from accelerometer and gyroscope data registers directly and 16 bit values are obtained by combining the lower 8bit and higher 8bit data register values. Our sampling rate is 200Hz. The accelerometer sensitivity is 16384 LSB/g and gyroscope sensitivity is 131 LSB/dps. Obtained information from IMU is send over ZigBee to the processing computer for plotting the graph. Graphs are plotted by using MatPlot in Python.

The camera module is used to take the images of the field of inspection continuously. We need to obtain real world coordinate positions of robotic vehicle from the 2D image captured by the camera. For this purpose, OpenCV tools can be used. By using the colour matching technique keep track of the current position of mobile robot. By this technique, pixel coordinate values are obtained. The real world X, Y coordinate values are generated by multiplying the pixel coordinate values with a scale factor K, which is obtained from camera calibration.

III. Software And Coding

Mainly two softwares are used to complete the entire project work.

- Python
- LPCXpresso

3.1 Python 2.7.14

Python is completely a general purpose, open-source programming language that can be used for variety of applications. It is developed with an open source license, making it easily usable and portable. Python's own IDLE is used as the Graphical User Interface.

For real-time computer vision applications OpenCV (Open Source Computer Vision) library can be used. OpenCV-Python makes use of Numpy, which is a highly optimized fundamental library for numerical operations.

3.2 LPCXpresso 7.3.0

LPCXpresso IDE is NXP's low-cost microcontroller (MCU) development platform ecosystem, which provides an end-to-end solution and enabling engineers to develop different embedded applications from initial stage to final production.

This is a complete embedded C/C++ integrated development platform. It includes, a software development platform for creating embedded applications for NXP's ARM-based "LPC" range of MCUs. In this project using LPC1769 platform.

There is a range of LPCXpresso boards that work without any break with the LPCXpresso IDE. These boards mainly used for point for your LPC Cortex-M MCU based projects, provide easy-to-use and practical development hardware. A Workspace is created which is simply a directory that is used to store the project. Multiple projects can save in a single workspace, and can have multiple Workspaces on your computer. We can only access single Workspace at a time through the LPCXpresso IDE, although it is possible to run multiple instances in parallel with each instance accessing a different Workspace.

IV. Position Estimation Algorithm

To determine the translation of the MPU6050 the change in place (displacement) must be known. The acceleration of the MPU6050 must be converted into actual displacement in meters. For any moving body, with an acceleration 'a', its velocity 'v' can be obtained by integrating acceleration with time't'.

$$v = \int a.\, dt \qquad (1)$$

Displacement 'x' can be calculated by integrating velocity with time.

$$x = \int v.\, dt = \int (\int a.\, dt)dt \qquad (2)$$

To understand this formula, we can define the integral as the area under the curve, where the integration is the sum of very small areas whose width nearly equal to zero. Since we requires real time displacement updates from discrete input values. Therefore trapezoidal integration is used here. Trapezoidal integration is a discrete method that uses the current and previous measurement to determine the integrand. Suppose we have n such samples of signal, a first order approximation about the signal could be obtained as follows. In order to avoid confusion, sampling time T is taken as unity.

$$Area_n = Sample_n + \left(\frac{Sample_n - Sample_{n-1}}{2}\right)*T \qquad (3)$$

Fig.2 shows the flow diagram of the position estimation algorithm.

 DOI : 10.18231/2454-9150.2018.0500

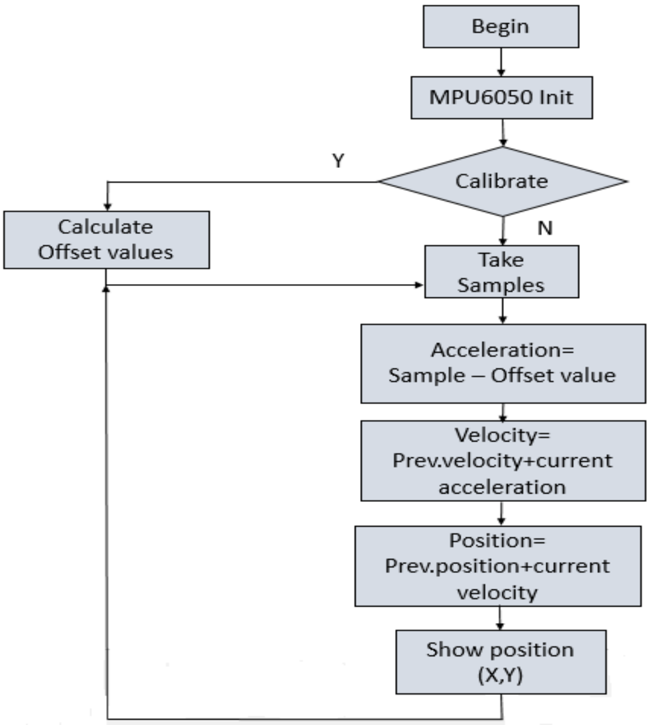


Fig.2. Position Estimation Method

By using this algorithm we can obtain displacement values in each axes form the acceleration. At first, we have to initialize the MPU6050 sensor by configuring its power management registers and other configuration registers. .For real world interpretation of data calibration should be done before taking the samples. Calibration is performed on the accelerometer by reading the data registers when there is a no movement condition. The output or offset obtained is considered the zero point reference, which is subtracted from the measured sample value and obtain the actual acceleration.

Accelerometer outputs the linear acceleration on each axes are in G's, where 1G = 9.8 m/s2. So we will convert the acceleration values in m/s2. Then integrating the acceleration measurement will yield a velocity in m/s, and a position in meters. First order integration is done by using equation (3). According to first equation of motion (v = u+at where v is the final velocity, u is initial velocity, a is acceleration and t is the time),

Current velocity = Previous velocity + (Current acceleration * Sampling time). (4)

Similarly, integration of velocity values will give the displacement.

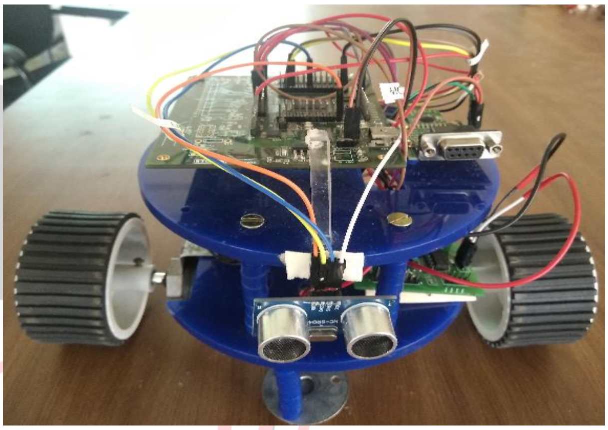
Current displacement = Previous displacement + (Current velocity * Sampling time). (5)

The obtained displacement values in X and Y directions stored and repeat above steps.

V. EXPERIMENT RESULTS

This project is aimed to estimate the position of a mobile robot by using IMU sensor and camera module**.** So the initial stage was to design and setup the chassis of the robot. The entire body of the robot was made up of acrylic material with 6mm thickness. Fig.3 shows the structure of the robotic vehicle used for this study. DC motors are controlled by PWM signals. CC2500 is inerfaced with UART1 in LPC1769 board. IMU sensor interfacing uses I2C0. Ultrasonic sensor is connected with GPIO pins of ARM board. For the objective of this study, the position of the mobile robot is estimated by using IMU and Camera sensor. The different sensor readings are verified by Docklight software.

For the case study, distance travelled by the robot within 15 seconds from the starting point is measured by using IMU sensor data and image information from camera sensor.



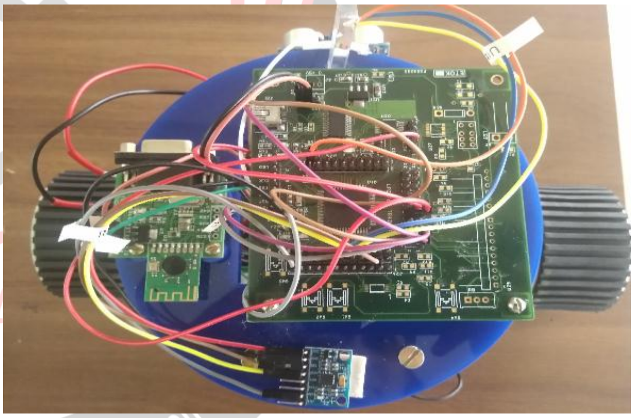
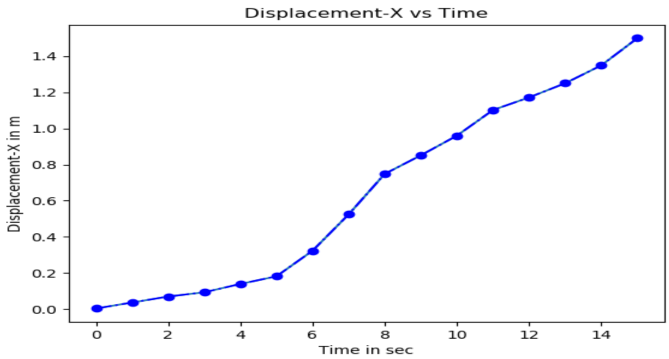


Fig.3. Proposed hardware set up

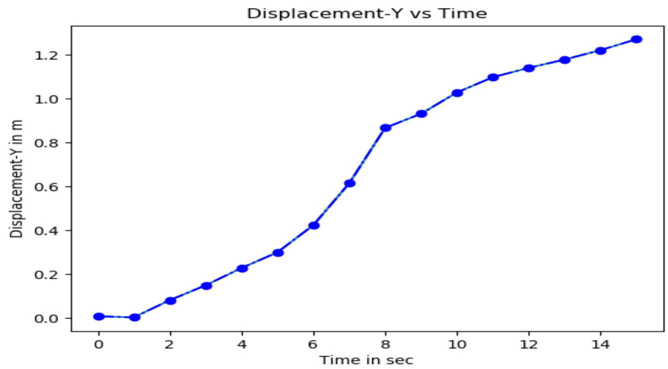
5.1 IMU Data Reading

I2C communication is used to interface MPU6050 with LPC1769. Raw values are read from accelerometer and gyroscope data registers directly and 16 bit values are obtained by combining the lower 8bit and higher 8bit data register values. After integration of accelerometric data, velocity values are obtained. Integration of velocity values gives the displacement information along X, Y axes of the vehicle. These values are send to the central PC through the wireless module CC2500. Graph is plotted by using Matplot. Fig.4 shows the displacement plot of the vehicle obtained from the IMU information. We can see that within 15secs, the vehicle displacement in X- direction is 1.5 meters and in Y-direction is about 1.3 meters from the starting point.

 DOI : 10.18231/2454-9150.2018.0500



(a)



(b)

Fig.4. Displacement plot obtained from IMU information (a) along X-axis (b) along Y-axis

5.2 Data Reading from Camera module

The camera module is directly connected to the PC for this study. First we set a pixel boundary in the image. The movement of the robot is continuously monitored by colour matching technique controlled its motions within the fixed boundary. Pixel coordinates corresponding to the current position of robot is retrieved first. Then the pixel coordinates are converted into real world coordinates by multiplying with the scale factor.

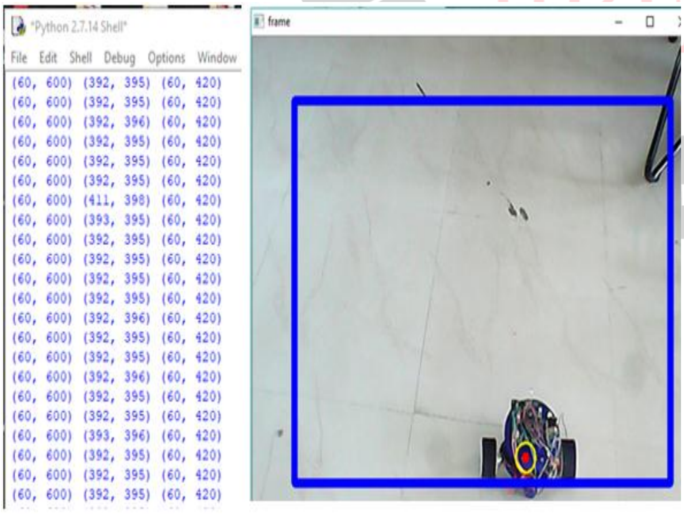
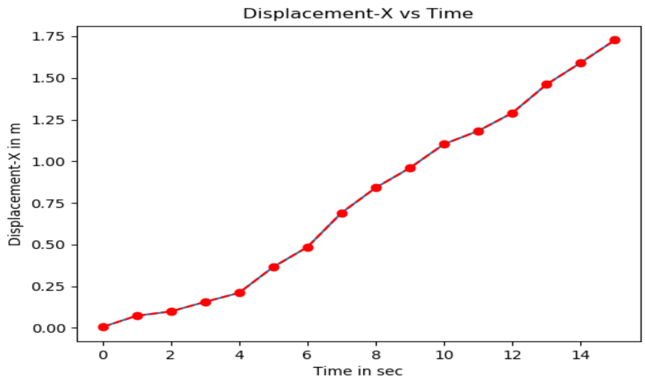
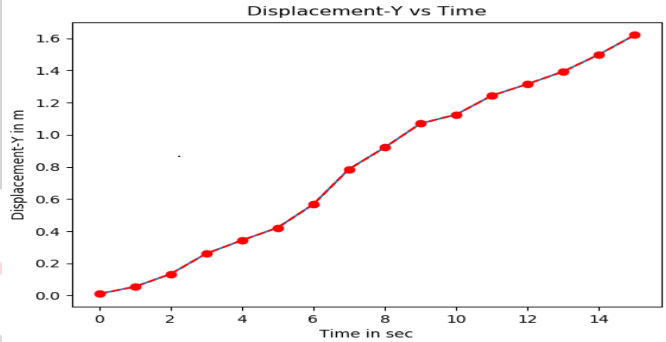


Fig.5. Pixel values of current position of robot

In the Fig.5 pixel coordinate values (X, Y) corresponding to the current position of robot are shown. First column values shows the X axis boundary and last column values shows the Y axis boundary. The column values shows the current (X, Y) position of the robot. Fig.6 shows the displacement plot of the vehicle obtained from the image information. In Fig.6(a), we can see that within 15secs, the vehicle displacement in X- direction is 1.75 meters. From Fig.6(b), the vehicle displacement in Y-direction is about 1.6 meters from the starting point within 15sec.



(a)

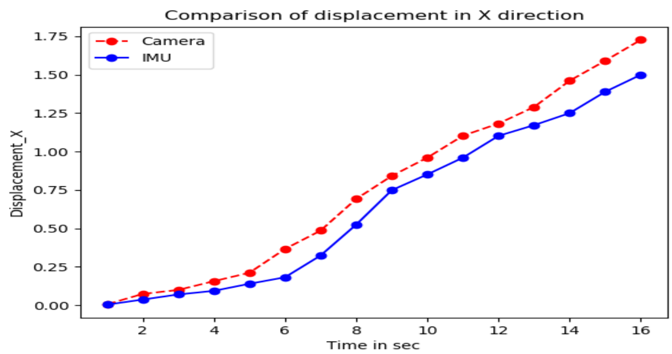


(b)

Fig.6. Displacement plot obtained from image information (a) along X-axis (b) along Y-axis

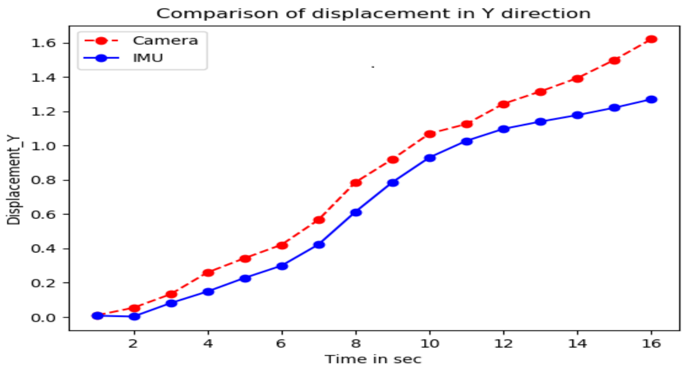
VI. Comparison Of Results

The positional values of the mobile robot in X, Y directions obtained from IMU and camera module are compared with the help of a graph. Fig.7 shows the comparison of displacement plot in X and Y-direction using IMU and camera sensor. The actual distance travelled by the mobile robot is 1.9 meter in X – direction and 1.7 meter in Y-direction within 15secs. From the Fig.7(a), the displacement in X- direction is about 1.5 meter by using IMU information and it is measured 1.75 meter by using image information. Similarly, in Fig.7(b), IMU measured displacement in Y- direction is about 1.3 meter and it is 1.6 meters measured by using image information from the camera module.



(a)

 DOI : 10.18231/2454-9150.2018.0500



(b)

Fig.7. Comparison of Displacement (a) along X-axis (b) along Y-axis

From these results we can understand that camera based method giving more accurate results. In the case of IMU, errors are accumulating over time. Misalignment of sensor axes or miscalibration causes measurement errors in acceleration.Fig.8 shows the calibrated data from IMU at no movement condition.

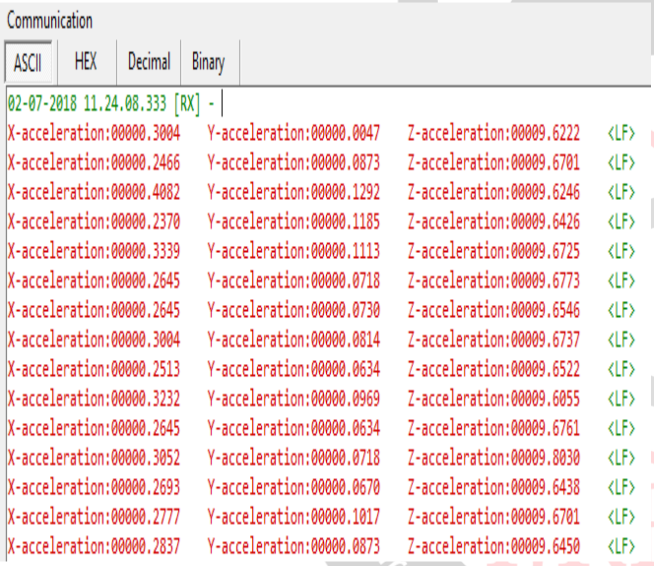


Fig.8. Calibrated data from Accelerometer (m/s2)

It can be observed that in the accelerometer calibrated data, X and Y axes are approximately zero and the Z-axis is nearly 9.8m/s2. There is no acceleration on X and Y directions, in fact, the only acceleration present is the earth's gravity, which is along the Z-axis pointed downward. The fluctuations which are seen in X, Y axes are white noise. This noise is inseparable from the data; it will cause the drift. Accumulation of errors in the double integration of acceleration can greatly affect the position estimation.

Table1 present the results of the trials under each test condition.

Table 1. Test results with the desired displacement of 1 meter

Trial	Measured displacement with IMU (m)	Measured displacement with camera (m)
1	1.23	0.92
2	0.81	1.07
3	1.34	1.18

VII. CONCLUSION

In this project, a comparative study of position estimation of autonomous mobile robot with the help of inertial sensors and camera module is presented. We have successfully developed an autonomous mobile robot with LPC1769 board and IMU sensor. Position of the robot is estimated by using IMU sensor and camera module. Experiments are also conducted to demonstrate the effectiveness of each method. IMU based position estimation is useful in situations where precision in is not extremely critical. With low cost consumer grade IMU, the accuracy if result is poor. In this study, camera based position estimation method give more accurate results. In future, the errors in IMU measurements can be reduced by the implementation of extended Kalman filter based algorithm.

REFERENCES

[1] Alexandru Barnea, Sid Ahmed Berrabah, Cezar Oprisan, Ioan Doroftei, "IMU (Inertial Measurement Unit) Integration for the Navigation and Positioning of Autonomous Robot Systems", CEAI, (2011) Vol.13, No.2, pp. 38-43.

[2] Youngmok Yun, Jingfu Jin, Namhoon Kim, Jeongyeon Yoon, Changhwan Kim, "Outdoor Localization with Optical Navigation Sensor, IMU and GPS", IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI) 2012.

[3] Mohammad Emal, Qazizada, Elena Pivarciova, "Mobile robot controlling possibilities of inertial navigation system", International Conference on Manufacturing Engineering and Materials, ICMEM (2016), 404 – 413.

[4] Christian B. Abellanosa, Ruth Pearl J. Lugpatan, Diogenes Armando D. Pascua, "Position Estimation using Inertial Measurement Unit (IMU) on a Quadcopter in an Enclosed Environment", International Journal of Computing, Communications & Instrumentation Engg. (IJCCIE) Vol. 3, Issue 2 (2016) ISSN 2349-1469 EISSN 2349-1477.

[5] Junghun Suh, Seungil You, Sungjoon Choi, Songhwai Oh, "Vision-Based Coordinated Localization for Mobile Sensor Networks", IEEE Transactions on Automation Science and Engineering (2014) 1545-5955.

[6] M. Saifizi, D. Hazry, Rudzuan M.Nor, "Vision Based Mobile Robot Navigation System", International Journal of Control Science and Engineering (2012), 83-87.

[7] Joko Siswantoro,Anton Satria Prabuwono,Azizi Abdullah, " Real World Coordinate from Image Coordinate Using Single Calibrated Camera Based on Analytic Geometry'', Communications in Computer and Information science book series (CCIS, volume 378).

[8] Joel A Hesch, Dimitrios G Kottas, Sean L Bowman, Stergios I Roumeliotis, "Camera-IMU-based localization: Observability analysis and consistency improvement", The International Journal of Robotics Research, (2014) Vol 33(1) 182–201.

 DOI : 10.18231/2454-9150.2018.0500