Design, building and performance testing of GPS and computer vision combination for increasing landing precision of quad-copter drone

Heru Supriyono, Annaduny Akhara
Department of Electrical Engineering, Universitas Muhammadiyah Surakarta (UMS), Surakarta, Indonesia
E-mail: Heru.Supriyono@ums.ac.id

Abstract. This paper discusses the effort proposed for increasing accuracy of autonomous landing of quadcopter on available landing pad by combining GPS and computer vision. The main objective of this work is to recognize and to capture image from landing pad and then used it for controlling quad-copter during landing mode along with global positioning systems (GPS). The work were including design and building hardware and software of quad-copter using commercially available instruments and then tested its autonomous landing performance on predetermined landing pad. The landing pad is in the form of 3 circles black and white with similar center point on it, i.e. the inner circle (the smallest one) has radius of 25 cm, The second inner circle (the middle circle) has radius of 60 cm and the outermost circle has radius of 100 cm. The landing pad image was captured in real time and then was processed by using Canny Edge detection method involving steps, i.e. converting RGB image to grayscale, removing noise by adding blur, finding contour of image and then finally using centroid method to get center coordinate of landing pad. Testing was carried out in the outdoor environment in the sunny-day condition. Flight test was done by comparing the precision of autonomous taking off landing of quad copter using GPS only and GPS combined with computer vision algorithm on altitude 5 meters. The test results showed that landing error of quad-copter by using GPS only was 199.2 centimeters in average from the centre point compared to 42.2 centimeters when quad copter using combination of GPS and computer vision technique.

1. Introduction
In this modern era the development of Unmanned Aerial Vehicle (UAV) or commonly called as a drone is growing rapidly. It can be noted from the literature that UAV or drone was proposed to be applied for broad areas. In agriculture, drone was used broadly from monitoring condition to spraying pesticide [1]. Drone or UAV also was reported to be proposed for various remote sensing for urban area involving for example coastal environment [2], environmental management [3], and infrastructure and facilities monitoring [4]. Also, UAV/drones have potential to be applied for surveillance or monitoring in rescue mission such as sending data in the form of image or audio-video [5]. In the area of art and cinematography, drone was reported very potential to be applied for resulting multi-view videography in level of experts [6].

There are various types of UAVs or drones, i.e. could be in the form of aeroplane (fixed wing), helicopter (single rotor), multicopter and hybrid of fixed wing and with rotor (Vertical Take Off and Landing (VTOL)) [7]. Every UAV type has its own positive side and negative side. So
that what type of UAV should be developed would be mainly determined by the task and area it should be finished. Multi-rotor UAV has many advantages such as its maneuverability, its hovering ability in the air, could be used for both indoor and outdoor environment, able to VTOL and it has relatively simple design. Based on these positive aspects, multi-rotor UAV has been proposed for many application. However it has main concern in some aspects including its relatively small area coverage, its relatively limited payload and its durability especially in its flight time. For multi-rotor UAV, the most commonly developed and employed by researchers is quad-copter type. This type has 4 drive motors and propellers that have the form X or H in general. Quad-copter technology has the advantage of its movement system, which can move VTOL. This means that the quad-copter only requires a take-off place and landing that is not large and even very small.

Global positioning systems (GPS) is a navigation system which was initially used for military-related activities only however now it has been extensively used in the wide area of civilian activities in the last ten years such as transportation, building or people position, farming, and fisheries [8]. GPS was also used in aviation including aerial vehicle as well as UAV, i.e. for informing its current position, recognizing the destination. Especially for UAV/drone GPS has been used to recognise landing coordinate of landing pad in autonomous landing process. However, for particularly environment such as narrow place, it is very difficult for GPS to get precision coordinate while landing [9]. Quad-copter operation has an important landing phase and if not incorrectly done it would be produce a fatal moment. Relying on GPS for autonomous landing system of UAV/drone only is lack of the accuracy of the landing. So that, researchers looking at the potential of combination between GPS and image processing [10].

The potential utilisation of computer vision based method for guiding landing of aerial vehicle has attracted interest of researchers. Gautam [11] has applied a vision based control systems for landing of helicopter in a moving object., i.e. ship. The simulation results showed that the proposed system provide promising performance. Another positive results on the application of vision based landing systems also was reported for airliner [12]. Similar work but implemented for autonomous landing systems of multirotor micro-air vehicle with high speed moving ground vehicle was reported by [13]. Besides for moving landing target, computer vision-based also was employed for static, i.e. the landing target is not moving landing page. Autonomous landing system of a drone for static landing pad was reported by Gonzales [16] where two-dimensional image was used for the landing target. This image was captured by using PiCamera and then was processed by Rasberry Pi 3B+ processor. The goal for the process was to detect edge of the target or landing pad which was in black and white. The reason of using black and white color of the landing pad marker was because the more significant change of pixel in local intensity values the more easily to compute the magnitude gradient [14]. The usage of computer vision based landing method for quad-rotor aerial vehicle also was reported able to increase the landing accuracy, closer to the centre coordinate of the landing pad [15]. The computer vision method able to compensate the landing accuracy caused by the external disturbance and inaccuracy of GPS.

Based on the previous works above, here, in this publication, the autonomous landing for quad-copter based on combination of GPS and computer vision was proposed and tested. The image target of landing pad has form of circle. The computer vision was employed to get satisfying goal, i.e. quad-copter able to land in the centre of the landing pad. The computation steps are including percept the landing pad image, changing pixel from image to grayscale, removing noise with blur, applying Canny Edge detection algorithm, finding contour from edge detection and finally locating the center coordinate of circle which will be detected by using centroid method. Image has 330 x 240 pixels and will be divided by 9 grids equal.
2. Method
This published work is experimental research or research and development type research. The framework of research method is described in figure 1.

The system architecture phase describes the design process and the results of general system of quad-copter. After system architecture has been identified, then, the next is target layout design. In this phase, the landing pad construction as the target with its size for quad-copter landing was designed. Next, hardware design phase, is a construction and configuration of quad-copter including electrical side and mechanical side. Object detection algorithms phase explains the algorithm how to find target from landing pad and for quad-copter movement is the algorithm to control quad-copter movement according to detected object.

2.1. System Architecture
The systems architecture of the quad-copter and its targetted landing pad could be seen in figure 2.

In the developed quad-copter, Pixhawk was used as the flight controller to control the movement of four brushless direct current (BLDC) motors in order to control the position of quad-copter. Pixhawk get input from sensors such as gyroscope, accelerometer, compass and GPS. Then, Raspberry Pi was employed to process the image or target that has been captured by PiCamera according to the determined algorithm. Raspberry Pi and Pixhawk able to communicate each other via MAVLink protocol over a serial connection. The ground station, here is Android phone, was used to remote Raspberry Pi desktop via Wi-Fi with frequency 2.4 GHz.

2.2. Target Layout Design
For the target, landing pad has white and black ring circles with dimension 120 cm to be seen by camera with high altitude. The landing pad was made of very low reflective materials basically
in white and then there is a square line as boundary. Inside the square line given, there are three circles with different dimension. The biggest circle or the outermost circle (the color is black) has diameter of 100 centimeters in while the middle circle has diameter of 60 centimeters and for the inner circle or the smallest circle (the color is black) has 25 cm. The design of the landing pad is shown in figure 3.

![Landing Pad Design](image)

**Figure 3:** Landing Pad Design

The target as shown in figure 3 for precision landing made of plywood material. The purpose is to get better quality of image captured for precision landing. This material is tested outside at field with altitude 5 meters. Ring circle was used because when the altitude of quadcopter is low the camera still be able to capture the image of the inner (smallest) circle. The experiments were carried out in outdoor environment, i.e. at the open field with sunny weather.

### 2.3. Hardware Design

The designed quad-copter (can be seen in Figure 4) has frame i.e. factory made frame with size diagonally 450 millimeters. The components used for developing quad-copter are including brushless motor, battery, Pixhawk, Raspberry Pi 3B+, PiCamera, Electronic Speed Controller (ESC) and Propeller.

### 2.4. Object Detection Algorithm

The algorithms of object detection are used to detect the center coordinate of circle. Both of algorithms will be compared to find center coordinate with Canny Edge detection and Hough Transform in both of target landing pad. The making of program script according to algorithms using Python language with library OpenCV. The Canny Edge detection and Hough Transform algorithms are shown in figure 5.

As shown in figure 5, data from camera which was installed on quad-copter is processed by image processing method such as conversion of Red Green Blue (RGB) color to grayscale in order by using formula (1).

\[
\text{RGB to Gray} = 0.299R + 0.587G + 0.114B \tag{1}
\]
The next process is applying Canny Edge detection algorithm by removing the noise using Gaussian filter in equation (2).

\[ G(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2+y^2}{2\sigma^2}} \]  

(2)

where \( x \) is the distance from the origin in the horizontal axis, \( y \) is the distance from the origin in the vertical axis, and \( \sigma \) is the standard deviation of the Gaussian distribution. The next process is finding intensity gradient of the image written in equation (3).

\[ \text{Edge Gradient } (G) = \sqrt{G_x^2 + G_y^2} \quad \text{angle } (\theta) = \tan^{-1} \left( \frac{G_y}{G_x} \right) \]  

(3)

where \( G \) is edge gradient, \( G_x \) is horizontal direction and \( G_y \) is vertical direction value for the first derivative. Canny Edge detection method able to minimize the contamination on edge of detected image so that it is easier to do image processing for landing place. After that is finding the contour related with the intensity of each pixel from edge detected of object. Last step is to find the center of coordinate by using centroid \((x, y)\) in pixel from the circle which has detected to get the location to control the movement of quadcopter. Next is converting the center coordinate \((x, y)\) by centroid method written in equation (4) and (5).

\[ x = \frac{m_{10}}{m_{00}} \]  

(4)

\[ y = \frac{m_{01}}{m_{00}} \]  

(5)

where \( x \) is the x coordinate and \( y \) is the y coordinate of the centroid.
2.5. Quad-copter Movement for Landing

The landing movement process of the quad-copter is started from the detection of the target in landing pad which can be seen in the flowchart Figure 6.

![Flowchart of quad-copter landing movement](image)

Figure 6: Flowchart of quad-copter landing movement

As referred from figure 6, the quad-copter movement detects the object of landing pad as an input for the system. In certain altitude, the object target will be processed and determine center coordinate then it is decide where the coordinate is placed in the grids. If the center coordinate is not in one of the grids so that the quad-copter will be hovering until the target is detected. When it is detected in one of the grids, the quad-copter will move based on the command in the grid and when the center of coordinate is in the middle of the grid then quad-copter will move downward or landing. Figure 7 shows the grid for controlling quad-copter autonomously landing. Similar technique was previously used by [9].

As shown in figure 7, camera that is set in 640x480 pixel then resize into frame 330x240 pixel due to the processing speed limitation of Raspberry Pi so that the smaller size of pixel for image the better. The frame was divided into 9 parts, i.e. AD, BD, CD, AE, BE, CE, AF, BF and CF. There are 2 equations for dividing the frame in x and y pixel shown in formula (6) and (7).

\[
\text{x axis} = \frac{x \text{ axis}}{3} \quad (6)
\]

\[
\text{y axis} = \frac{y \text{ axis}}{3} \quad (7)
\]

According to the x axis equation, there are three dividing part with each of them consists of 110 pixel. For y axis equation, it has 80 pixel each of part. So that, x axis has range 0 – 110 (for part A), range 111 – 220 for part (B) and range 221 – 330 (for part C). For y axis, the range is 0 – 80 (for part D), range 81 – 160 (for part E) and range 161 – 240 (for part F).
The possible movements of quad-cpter based on the detected object in the grid are including moving forward, backward, left, downward, and right as can be seen in Figure 8. However if the quad-cpter could not detect the full object in any part of the grid then it will hovering.

The possibilities of movement of quad-cpter can be summarised as follow: (1) quad-cpter move forward if center coordinate in AD, BD and CD with x axis range 0 – 330 pixel and y axis 0 – 80 pixel which can be seen in Figure 8a, (2) quad-cpter will move backward if center coordinate in AF, BF and CF with x axis range 0 – 330 pixel and y axis 161 – 240 pixel which can be seen in figure 8(b), (3) quad-cpter will move left if center coordinate in AE with x axis range 0 – 110 pixel and y axis 81 – 160 pixel as can be seen in figure 8(c), (4) quad-cpter will move downward or landing if center of coordiante is in BE with x axis 111 – 220 pixel and y axis 81 – 160 pixel as can be seen in figure 8(d), (5) quad-cpter will move right if the center of coordinate in CE with x axis range 221 – 330 pixel and y axis 81 – 160 pixel as can be seen in figure 8(e) and (9) if the center coordinate is not in one of all grids or not detected then the quad-cpter will be hovering until the quad-cpter detect the target as shown in figure 8(f).

This possibilities of landing movement is similar to that used by [9], however, Lilian [9] used Douglas Peucker algorithm while this publication used Canny Edge detection algorithm.

3. Result and Discussion

To know the its performance, the built autonomous landing systems of the quad-cpter was tested. The experiment carried out by testing quad-cpter precision landing in GPS only and GPS with camera which employ Canny Edge detection algorithm [17–21]. All tests were conducted at peed of quad-cpter movement at level of 0.2 m/s and altitude 5 m.

3.1. Configuration of Quad-cpter and Landing Pad

The drone has been built as desired design. It has microcontroller that has been embedded with other sensor needed for quad-cpter called Pixhawk controller board as the flight controller. Then, Pixhawk is connected to Raspberry Pi to command quad-cpter movement. Communication between ground station and quad-cpter through Wi-Fi from phone to Raspberry Pi, so Android phone remote the desktop of Raspberry Pi to run the script. Figure 9 contains a picture of the real configuration of quad-cpter and table 1 provide detailed the specification of resulted quad-cpter. The landing pad material was made of plywood which can be checked in Figure 10.
3.2. Quad-copter Movement toward Landing Pad Detection

For the movement, quad-copter was tested in drone Simulation In The Loop (SITL) before apply it in real condition using software mission planner and linux ubuntu terminal on windows 10. In drone SITL and real has similar condition for running program and result. The image captured from real application of drone in field by recording the landing pad from altitude 5 m. Script of processing and controlling quadcopter with Canny Edge detection algorithm that have finished then run into drone SITL as shown in figure 11.

Figure 8: Quad-copter possible movement according to the position of detected object at the grids
Figure 9: Actual shape of built quad-copter

Table 1: Quad-copter Specification

| Specification         | Value                  |
|-----------------------|------------------------|
| Height (cm)           | 27                     |
| Total Weight (gram)   | 1330                   |
| Frame Size (mm)       | 450                    |
| Motors                | 2312 920 kV            |
| ESC                   | Simonk 30A             |
| Battery               | POWER 2800 mAh 25 C    |
| Propeller (inch)      | 9 x 4.5                |
| Safe Flight Time (min)| ± 5 - 7                |

Information: (1) GUI (Graphic User Interface) of GCS (Ground Control Station) mission planner; (2) Ubuntu terminal for opening connection between Ubuntu and mission planner; (3) Ubuntu terminal for creating firmware of SITL in mission planner; (4) Ubuntu terminal for executing the program and output of executed program; (5) Frame output from running program in Object Detected; (6) Parameters of condition quad-copter such as altitude, compass, speed and distance to quad-copter.

Library that was used to create the image processing, control drone and support the program was as following: (1) print_function from future module it means print_function used to bring the print function from Python 3 into Python 2.6+; (2) import time used for creating time sleep or delay in the program; (3) cv2 is from openCV library used for image processing; (4) mavutil from pymavlink module it means mavutil is a python mavlink utility function to implement the MAVLink protocol; (5) numpy is fundamental package for scientific computing with Python; (6) deque from collection is a list optimized for inserting and removing items; (7) DroneKit allows developers to create apps that run on an onboard companion computer and communicate with the ArduPilot flight controller.
Figure 10: Plywood Landing Pad

Figure 11: Drone SITL with Canny Edge detection algorithm
In accordance with figure 11, number 4 shows coordinate position of center from circle detected in Object Detected windows and followed with the command of controlling quadcopter. This simulation has been tested by Canny Edge detection and Hough Transform algorithm. Experiment of quad-copter movement toward grid position has similar as desired design from the frame of image in simulation. Figure 12 shows quad-copter movement toward grid position.

![Quadcopter Movement](image)

**Figure 12:** Quadcopter Movement toward Coordinate of Grid Position

### 3.3. Result of Autonomous Landing Performance

The autonomous control of quad-copter is from GPS in the system. Even for flight mission or landing moment. GPS device has error value to determine the latitude and longitude then it is not recommended to use GPS only as a landing mission. For reducing the error of autonomous landing, GPS was combined with camera to detect the landing pad.

Image processing has important role for getting precision coordinate from the target. Different algorithms will give different results. Canny Edge detection algorithm here does not use threshold as conversion of binary image, so the intensity of light is not influence at all. Another algorithm is Hough Transform using threshold, so the intensity of light has influence. These experiments have been with altitude 5 m measured by barometer sensor of quad-copter as shown in the table below. The distance error was measured by measuring the landing point of quad-copter to the center of coordinate of landing page.
Table 2: GPS Device Landing Performance

| No | Time      | Distance of Error (cm) |
|----|-----------|------------------------|
| 1  | 07.58 AM  | 236                    |
| 2  | 08.03 AM  | 222                    |
| 3  | 08.53 AM  | 145                    |
| 4  | 08.55 AM  | 111                    |
| 5  | 08.57 AM  | 282                    |

Error Average 199.2

In testing of autonomous take off and landing using GPS device only has error average landing 199.2 cm. From 5 times attempt the highest error is 282 cm and lowest is 111 cm.

Table 3: GPS Device and Canny Edge detection with Plywood Target

| No | Time      | Distance of Error (cm) |
|----|-----------|------------------------|
| 1  | 08.24 AM  | 31                     |
| 2  | 08.26 AM  | 55                     |
| 3  | 08.28 AM  | 48                     |
| 4  | 08.29 AM  | 52                     |
| 5  | 08.31 AM  | 41                     |

Error Average 45.4

In testing of autonomous take off and landing using GPS device combined with Canny Edge detection algorithm. For the landing used is plywood material which is not reflecting light. From 5 times attempt has error average landing 45.4 cm with the highest error is 55 cm and lowest is 31 cm.

4. Conclusion

Quad-copter was succesfully implemented by using comercially available instrument including the implementation of GPS plus Canny Edge detection method. The performance of built quad-copter was tested in outdoor environment on sunny day with altitude 5 meter. The test results showed that by using GPS only, the landing position of quad-copter was 199.2 centi meters in average distance from the center of landing pad coordinate compared to only 45.4 by using GPS plus canny edge detection. Based on these test results, it can be noted that the usage of GPS plus Canny Edge detection able to increase accuracy of quad-copter landing.

References

[1] Y. Z. H. Pei, "An Improved Vision-Based Algorithm for Unmanned Aerial Vehicles Autonomous Landing," Physics Procedia, pp. 935-941, 2012.
[2] R. Y. S. T. P. B and K. B. "Segmentation And Object Recognition Using Edge Detection Techniques," International Journal of Computer Science and Information Technology, pp. 153-161, 2010.
[3] V. F. Vidal, L. H. M, M. F. Santos, M. F. Silva, A. S. Cerqueira and E. J. Oliveira, "UAV Vision Aided Positioning System for Location and Landing," 2017 18th International Carpathian Control Conference, ICCC 2017, pp. 228-233, 2017.
[4] V. Tzivaras, Building a Quadcopter with Arduino, Birmingham: Packt Publishing Ltd, 2016.
[5] V. Sudevan, A. Shukla and H. Karki, "Vision based Autonomous Landing of an Unmanned Aerial Vehicle on a Stationary Target," International Conference on Control, Automation and Systems, pp. 362-367, 2017.
[6] C. Lilian, G. E. Setyawan and W. Kurniawan, "Sistem Pendaratan Otomatis Quadcopter dengan Pengolahan Citra menggunakan Metode Douglas Peucker," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIK) Universitas Brawijaya, pp. 4923-4931, 2018.

[7] S. Lange, N. Sunderhauf and P. Protzel, "Autonomous Landing for a Multirotor UAV Using Vision," Workshop Proceedings of the International Conference on Simulation, Modeling and Programming for Autonomous Robots, pp. 482-491, 2008.

[8] A. Hinas, J. M. Robert and F. Gonzalez, "Vision-Based Target Finding and Inspection of a Ground Target Using a Multirotor UAV System," Sensors, pp. 1-17, 2017.

[9] L. Goeller, "Drone Precision Landing using Computer Vision," Zurich, 2018.

[10] A. El-Rabbany, Introduction to GPS, Norwood: Seimitsu Kogaku Kaishi/Journal of the Japan Society for Precision Engineering, 2006.

[11] A. Gautam, P. Sujit and S. Saripali, "Autonomous Quadroto Landing Using Vision and Pursuit Guidance," IFAC, pp. 10501-10506, 2017.

[12] T. Nakamura, D. Magree and E. N. Johnson, "Estimation Techniques in Robust Vision-Based Landing of Aerial Vehicles." IFAC, pp. 11664-11669, 2017.

[13] J. Garcia-Pulido, G. Pajares, S. Dormido and J. d. l. Cruz, "Recognition of landing platform for unmanned aerial vehicles by using computer vision-based techniques," Elsevier, pp. 152-165, 2017.

[14] Y. Bi and H. Duan, "Implementation of autonomous visual tracking and landing for low-cost quadrotor," Elsevier, pp. 1-5, 2013.

[15] M. Bhargavapuri, A. K. Shastry, H. Sinha, S. R. Sahoo and M. Kothari. "Vision-based autonomous tracking and landing of a fully-actuated rotorcraft." Elsevier, pp. 113-129, 2019.

[16] R. C. Gonzalez, Digital Image Processing 2nd Edition, New Jersey: Prentice Hall, 2002.

[17] Muhammad Kusban, Aris Budiman, and P. Bambang Hari. Optimized palmprint recognition with gabor parameters adjustment and matching method selection. Advanced Science Letters, 24(12), 2018.

[18] Shiyi Liu, Zhenning Su, Ming Li, and Longtan Shao. Slope stability analysis using elastic finite element stress fields. Engineering Geology, 273:105673, 2020.

[19] Yufeng Tian and Zhanshan Wang. A new multiple integral inequality and its application to stability analysis of time-delay systems. Applied Mathematics Letters, 105:106325, 2020.

[20] Mustafa Turkyilmazoglu. Single phase nanofluids in fluid mechanics and their hydrodynamic linear stability analysis. Computer Methods and Programs in Biomedicine, 187:105171, 2020.

[21] Chen Feng, Jiancang Xie, Bo Sun, and Ni Wang. Numerical simulation of soil water and salt transport based on hydrus model in the storage condition. In 2011 International Symposium on Water Resource and Environmental Protection, volume 2, pages 830-833, 2011.