

Sistem Kendali Quadcopter Melalui Jaringan Internet Berbasis Lokasi dan Pengenalan *Marker* Menggunakan Smartphone

Hadi Supriyanto¹, Nur Afifah², Aris Budiarto³

¹Jurusan Teknik Otomasi Manufaktur dan Mekatronika, Politeknik Manufaktur Bandung, Bandung 40135
 hspriyant@yahoo.com, afifah4@gmail.com

ABSTRAK

Quadcopter adalah helicopter multirotor yang diangkat dan didorong oleh empat buah rotor dan memiliki kemampuan VTOL (*Vertical Take Off Landing*) dikendalikan menggunakan *flight controller* yang telah memiliki algoritma dan sensor-sensor tertentu dan dilengkapi komponen elektronika lainnya seperti gps, dan sensor kamera. Rancang bangun dan implementasi quadcopter kali ini memungkinkan mengikuti rute spesifik terencana yang telah ditentukan untuk menjalankan misi *waypoint* berbasis lokasi, kemudian muncul permasalahan pada saat menjalankan program *waypoint* menggunakan GPS pada penelitian sebelumnya yang ditandai dengan perubahan posisi lokasi *waypoint* selama melakukan percobaan 15 kali dengan perubahan jarak lokasi rata-rata 4.42 meter dan waktu pengiriman rata-rata data internet sebesar 2.27 sekon. Metodologi yang digunakan dalam penyelesaian permasalahan ini menggunakan sensor kamera sebagai pengenalan *marker* berupa bulat merah di atas tanah diproses oleh algoritma *image processing* di Raspberry Pi menggunakan *software* OpenCV. Tujuan dilakukan penelitian ini adalah quadcopter menjalankan misi *waypoint* yang telah ditentukan berupa *marker* untuk melakukan pendaratan. Dengan menggunakan sensor kamera pada quadcopter sebagai pengenalan *marker*, maka dapat menjalankan misinya menuju *waypoint* yang telah ditentukan berupa bulat merah, disamping itu didapat perubahan jarak rata-rata lokasi *waypoint* berupa *marker* sebesar 0.4 meter.

Kata Kunci

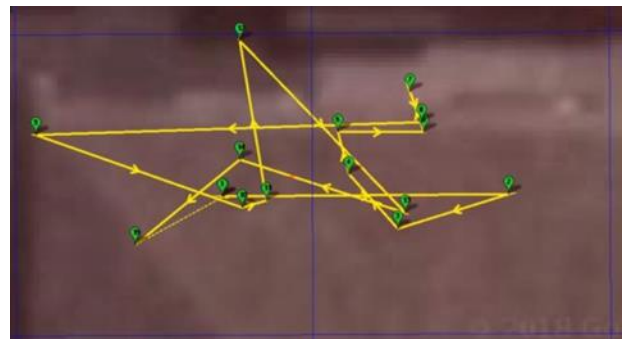
Quadcopter, VTOL, waypoint, marker

1. PENDAHULUAN

1.1 Latar Belakang

Unmanned Aerial Vehicle (UAV) berkembang lebih populer dan menjadi subjek pada banyak penelitian. Beberapa aplikasi nyatanya adalah untuk pengiriman barang, pemadam api, dan pembersih debu tanaman pada pertanian. Salah satu jenis *Unmanned Aerial Vehicle* (UAV) adalah multirotor contohnya yaitu quadcopter. Quadcopter menggunakan empat buah motor penggerak yang dikendalikan menggunakan *flight controller* dimana didalamnya telah memiliki algoritma dan sensor-sensor tertentu, dilengkapi dengan GPS, dan kamera sebagai sensor utamanya. Quadcopter memiliki dua mode yaitu manual dan *autonomous*, mode manual dapat dikendalikan oleh *remote control*, sedangkan mode *autonomous* dikendalikan dengan program yang di proses di *microcomputer* yang telah dibuat untuk menjalankan misi *autonomous* salah satunya

yaitu misi *waypoint*. Dibawah ini adalah gambar pada penelitian sebelumnya yaitu visualisasi dari sistem *waypoint* pada GPS Quadcopter berbasis lokasi



Gambar 1. Visualisasi Sistem Waypoint pada GPS Quadcopter[1]

1.2 Batasan Masalah

Berikut adalah batasan masalah dalam penelitian ini:

1. Quadcopter menggunakan mode terbang *autonomous* yaitu *guided*, *auto*, dan *land*.
2. Ketinggian maksimum terbang 30 meter karena keterbatasan komponen.
3. Faktor kecepatan angin, tekanan, suhu, dan tingkat gravitasi pada setiap tempat di asumsikan sama.
4. Resolusi kamera Pi 320, 240.
5. Metode pengenalan *marker* yang digunakan adalah pengenalan *contour*.
6. *Marker* berupa bulat/kotak.

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Untuk mengetahui kemampuan Quadcopter dalam menjalankan misi *waypoint* dari titik A ke titik B.
2. Untuk mengetahui karakteristik Quadcopter dalam melakukan pengenalan *marker* ketika berada di titik B.
3. Untuk mengetahui kemampuan Quadcopter dapat mendarat pada posisi *waypoint* yang telah ditentukan berupa *marker* bulat merah.

1.4 Metodologi Penelitian

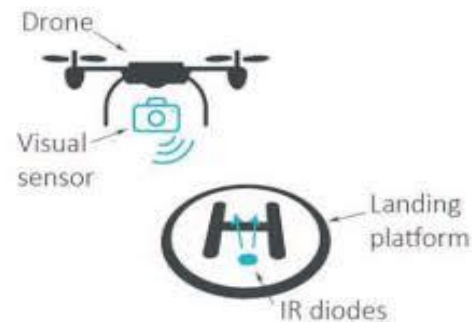
Tahapan penelitian yang dilakukan adalah sebagai berikut:

1. Mendefinisikan permasalahan pada quadcopter pada penelitian sebelumnya.
2. Melakukan *study literature* dan mencari referensi yang terkait dengan quadcopter.
3. Menganalisis permasalahan yang timbul pada saat quadcopter menjalankan misi *waypoint*.
4. Melakukan percobaan misi *waypoint* pada quadcopter selama 15 kali yang di tandai dengan lokasi *waypoint* yang berubah-ubah.
5. Merancang sistem kendali pada quadcopter menggunakan kamera sebagai sensor untuk pengenalan *marker*.
6. Mengimplementasikan algoritma *image processing* untuk pengenalan *marker* dan quadcopter melakukan misi pendaratan.
7. Melakukan uji coba quadcopter

8. Membuktikan perbandingan antara quadcopter menggunakan GPS berbasis lokasi dengan quadcopter berbasis lokasi dan pengenalan *marker*.

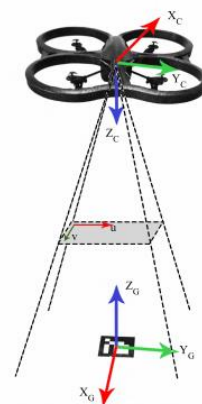
2. TINJAUAN PUSTAKA

Kemampuan quadcopter untuk menjalankan misi pendaratan pada objek telah banyak metode ataupun penggunaan sensor yang beraneka ragam. Berikut ini adalah quadcopter menjalankan misi pendaratan menggunakan sensor kamera dan IR diodes yang berada di atas marker agar pendaratan yang dilakukan akurat.



Gambar 2. Quadcopter menggunakan Sensor kamera dan IR diodes[2]

Penambahan sensor-sensor tertentu memungkinkan quadcopter untuk melakukan misi yang spesifik, salah satunya pengenalan objek atau marker untuk melakukan misi pendaratan, salah satunya adalah pengenalan objek berupa aruco marker seperti gambar dibawah ini.



Gambar 3. Visualisasi Frame Kamera Drone terhadap Frame Aruco Marker[3]

Pengenalan objek aruco marker menggunakan library aruco marker yang terdapat di OpenCv, pada pengenalan marker ini didapat sumbu x, y, z dari marker yang digunakan sehingga pengontrolan misi

pendaratan quadcopter dilakukan oleh kamera dan *microcomputer*.

Penelitian yang memanfaatkan sistem *waypoint* dan GPS sebagai salah satu cara untuk pergerakan sebuah quadcopter ialah penelitian yang dilakukan oleh Aji Kurnia Rahman yang berjudul “Rancang Bangun dan Implementasi Kendali Quadcopter Melalui Jaringan Internet Berbasis Lokasi Menggunakan Smartphone Android” yang dipublikasikan oleh Jurusan Teknik Mesin dan Manufaktur, Politeknik Manufaktur Bandung pada Desember 2018. Penelitian tersebut memanfaatkan sistem tracking *waypoint* pada sistem navigasi quadcopter dengan modul *gps* sebagai sensor utamanya untuk mengetahui posisi dari tujuan akhir pergerakan quadcopter dan smartphone berbasis Android untuk antar mukanya dengan menggunakan internet sebagai sistem komunikasi antara smartphone berbasis Android dan quadcopter. Hasil yang dicapai dari penelitian ini memiliki perbedaan jarak rata – rata antara posisi akhir quadcopter dengan posisi tujuan sebenarnya di lapangan sebesar 4,42 meter[1].

Penelitian yang memanfaatkan sistem *autonomous visual tracking* dan *landing* pada benda yang bergerak sebagai salah satu cara untuk pergerakan sebuah quadcopter ialah penelitian yang dilakukan oleh Yingcai Bi, dan Haibin Duan yang berjudul “*Implementation of Autonomous Visual Tracking and Landing for Low Cost Quadrotor*” yang dipublikasikan oleh Optik-International Jurnal Light Electron Optics, 2013. Penelitian tersebut memanfaatkan sistem *visual tracking* and *landing* pada sistem navigasi quadrotor dengan kamera sebagai sensor utamanya yang di proses pada algoritma *computer vision* menggunakan pendekatan *visual tracking* berupa warna merah, hijau, dan biru (RGB) untuk mengetahui *marker* dan melakukan misi pendaratan pada benda yang bergerak[4].

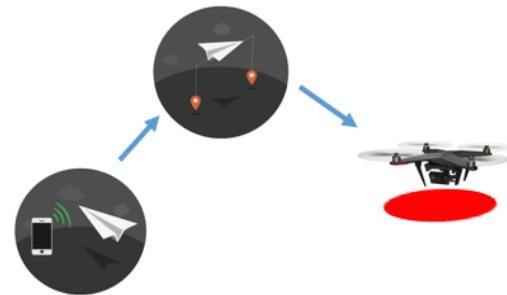
Penelitian yang memanfaatkan sistem algoritma *image processing* pada quadrotor dirancang untuk memerintahkan dengan mode *autopilot* untuk berpindah dari *marker* sekarang ke *marker* berikutnya sebagai salah satu cara untuk pergerakan sebuah quadcopter ialah penelitian yang dilakukan oleh Jonathan D. Blythe, Krzysztof A. Borowicz, dan Alyssa N. Hollander yang berjudul “Autonomous Quadrotor Navigation and Guidance” yang dipublikasikan oleh Jurusan Teknik Mesin Worcester Polytechnic Institute pada 2016. Penelitian tersebut memanfaatkan pendeteksian *marker* berupa kotak berwarna cerah menggunakan kamera Pi sebagai sensor utamanya yang diproses di

Raspberry Pi untuk menjalankan program *image processing* menggunakan *software* OpenCV[5].

Dari hasil – hasil tersebut maka dapat disimpulkan bahwa penelitian ini akan mengambil tema GPS dan kamera sebagai sensor utamanya untuk mengetahui posisi dari tujuan akhir pergerakan quadcopter sehingga quadcopter dapat menjalankan misi *waypoint* dan pengenalan *marker* berupa bulat merah dan melakukan misi pendaratan pada *marker* tersebut.

3. PERANCANGAN SISTEM

3.1 Perencanaan Sistem

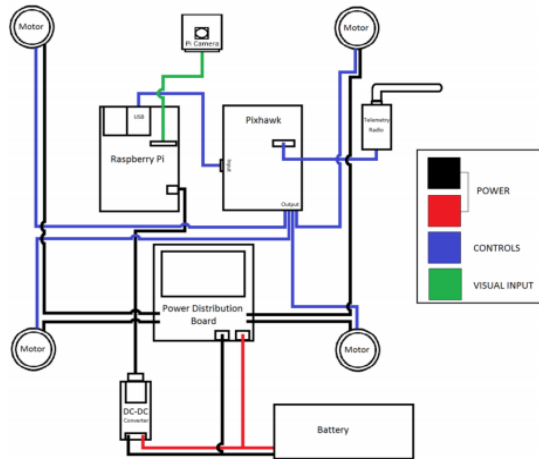


Gambar 4. Perencanaan Sistem

Pengguna menggunakan smartphone Android untuk mengendalikan quadcopter secara otomatis menuju lokasi dari smartphone Android tersebut dan terdapat *marker* sebagai misi pendaratan. Data posisi dari smartphone Android akan dikirimkan menuju *cloud storage* melalui jaringan internet, kemudian data tersebut akan diterima oleh *micro computer* pada quadcopter, lalu *microcomputer* akan mengirimkan program *autopilot* kepada *flight controller* sehingga quadcopter akan terbang menuju smartphone Android tersebut. Setelah quadcopter sampai pada lokasi smartphone Android dan terdapat *marker*, kemudian *microcomputer* menjalankan program *image processing* menggunakan *software* OpenCV untuk pendeteksian dan pengenalan *marker*. Kemudian quadcopter mencari koordinat X, Y dari *marker* bulat merah tersebut.

3.2 Perancangan Sistem

3.2.1 Pembuatan Perangkat Keras



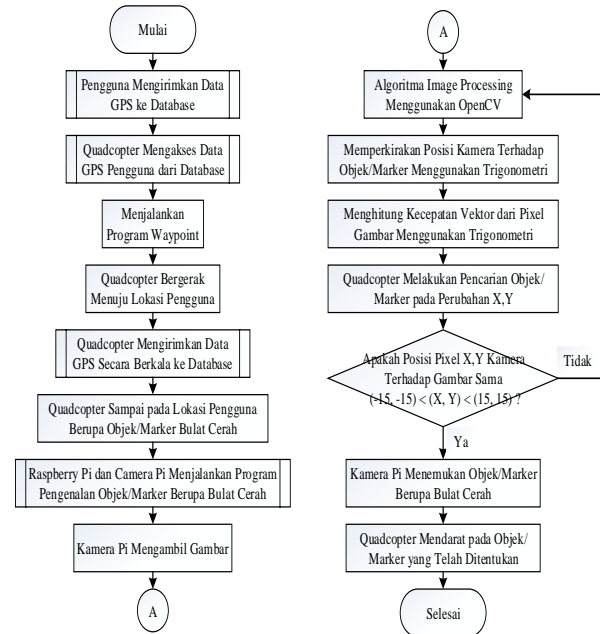
Gambar 5. Skematik Perangkat Keras Sistem[4]

Pixhawk digunakan sebagai *flight controller* yang berfungsi mengatur seluruh algoritma terbang dari quadcopter. Modul GPS digunakan untuk mengetahui posisi dari quadcopter. Kamera Pi sebagai sensor melakukan pengenalan *marker* berupa bulat merah. Raspberry Pi 3 yang terhubung dengan internet berfungsi sebagai penghubung antara *flight controller* dengan pengguna smartphone sebagai tujuan quadcopter ketika melakukan misi terbang, Raspberry Pi 3 juga berfungsi untuk memberi perintah kepada *flight controller* untuk melakukan misi terbang secara otomatis dan melakukan pengenalan *marker* menggunakan *algoritma image processing*. ESC yang terhubung dengan motor penggerak menerima sinyal dari *flight controller* untuk mengatur kecepatan dari motor tersebut. Semua perangkat diatas terhubung dengan *power module* sebagai pendistribusi sumber tegangan utama yang berasal dari baterai LiPo.

3.2.2 Pembuatan Perangkat Lunak

Pertama quadcopter diaktifkan menggunakan smartphone Android dan program pada micro computer sudah siap digunakan sekaligus

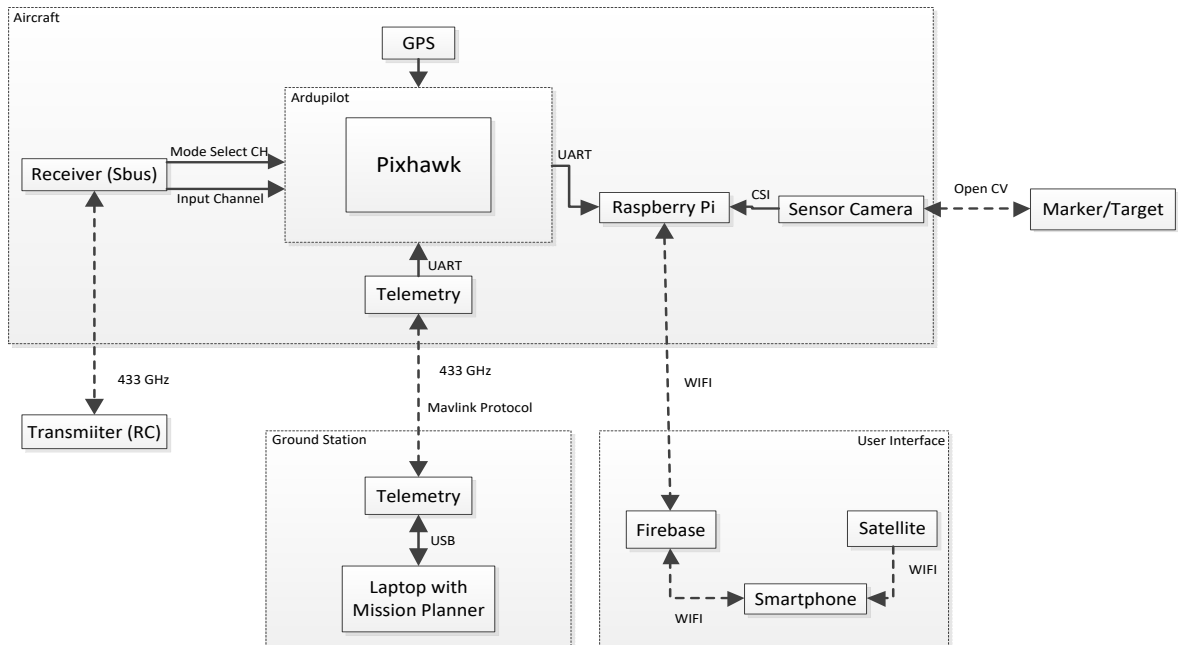
terhubung dengan internet untuk menjalankan misi waypoint, quadcopter menuju titik waypoint yang telah ditentukan berupa marker bulat merah. Kamera Pi dan microcomputer melakukan pengenalan marker yang di proses dengan software OpenCV. Ketika marker sudah ditemukan pada rentang pixel X,Y sebesar $(-15, -15) < (X, Y) < (15, 15)$. Setelah marker ditemukan maka quadcopter melakukan misi pendaratan.



Gambar 6. Diagram Alir Sistem

3.3 Blok Diagram Sistem

Tujuan blok diagram sistem adalah sebagai acuan dalam merancang sistem dan merancang kendali Quadcopter, gambar 5 menunjukkan blok diagram sistem. Blok Diagram ini terdapat komponen-komponen dan jalur komunikasi yang digunakan, komponen yang digunakan pada Quadcopter adalah Pixhawk sebagai flight controller, GPS, Telemetry, Receiver, Raspberry Pi, dan Kamera, Sedangkan untuk berkomunikasi antara Quadcopter dengan Ground station di permukaan menggunakan protokol mavlink.



Gambar 7. Blok Diagram Sistem

3.4 Design dan Spesifikasi Quadcopter



Gambar 8. Design Quadcopter

Tabel 1. Spesifikasi Quadcopter

No	Parameter	Nilai
1	Diagonal Wheelbase	450 mm
2	Baterai	5400 mAH
3	Durasi Terbang	20 menit
4	Berat	2,2 kg

3.5 Algoritma Image Processing

OpenCV adalah library dan *software open-source image processing* yang memungkinkan pemrosesan

dan pengolahan gambar menggunakan bahasa pemrograman C, C++, Python, MATLAB dan Java pada berbagai sistem operasi. Berikut adalah library OpenCV yang digunakan untuk pengenalan marker antara lain: `cv2.VideoCapture(0)`, `capWebcam.read()`, `cv2.cvtColor(...)`, `cv2.`

`GaussianBlur(...)`, dan `cv2.canny(...)`, serta `cv2.findContours(...)[5]`. Berikut ini adalah algoritma misi autonomous pada quadcopter.

1. From dronekit import connect, VehicleMode, LocationGlobalRelative, Command, LocationGlobal
2. From picamera import PiCamera
3. Import firebase
4. Import cv2
5. Camera setting
6. Connect to vehicle
7. Def initcam():
8. Def location_callback(self, attr_name, value)
9. Def arm_and_takeoff(aTargetAltitude)
 - a. Altitude takeoff = 5 m
10. Def send_land_message(x, y)
 - a. Sending pixel to find the marker

11. Set status for database
 - a. Status launch = 1/0, ON/OFF status from user
12. Wait until data launch set from user
 - a. Status launch = 1
13. Print "Ready to launch "
14. Print "Going towards destination."
15. Begin mission using image processing
 - a. Frame capture
 - b. Filter image noise
 - c. Find contours in the edge map
 - d. Loop over the contours
 - e. Ensure that the approximated contour is "roughly" rectangular
 - f. Compute the bounding box of the approximated contour and
 - g. Use the bounding box to compute the aspect ratio
 - h. Compute the solidity of the original contour
 - i. Compute whether or not the width and height, solidity, and
 - j. Aspect ratio of the contour falls within appropriate bounds
 - k. Ensure that the contour passes all our tests
 - l. Draw an outline around the target and update the status text
 - m. Compute the center of the contour region and draw the crosshairs
 - n. Check the position marker to camera
 - o. Is there camera find treshold pixel 240x240?
 - p. If finished : break
16. Drone landing at marker

3.6 Pengujian Sistem

Pengujian dilakukan dengan cara mengukur perbedaan jarak antara *marker* berupa bulat merah sebagai misi pendaratan dengan posisi *quadcopter* mendarat menggunakan GPS *smartphone* Android menggunakan meteran sehingga didapat perubahan lokasi pada GPS *quadcopter* ketika melakukan misi penerbangan menuju tujuan tersebut.

4. HASIL DAN PEMBAHASAN

Tabel 2. Perbedaan Jarak GPS Quadcopter dengan GPS Smartphone User

Pengujian ke-	Jumlah satelit GPS Quadcopter	Perbedaan Jarak dari tujuan akhir (m)	Waktu Tempuh (s)
1	16	5,1	75
2	13	4,6	66

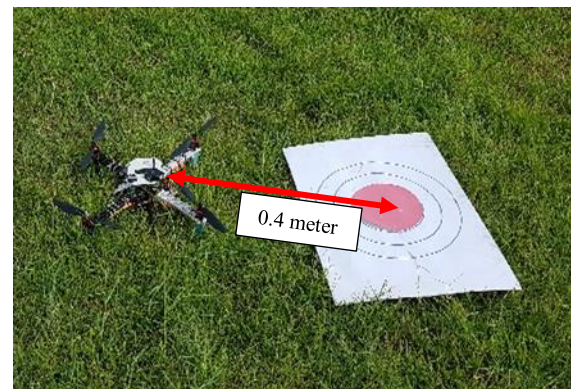
3	14	3,5	63
4	11	4,4	68
5	13	4,5	70
Rata - rata		4,42	68,4

Pengujian ini dilakukan untuk mengetahui hasil dari rata –rata jarak perubahan lokasi GPS pada *quadcopter* terhadap *smartphone* user berupa bulat merah. Data ke-1 dijadikan titik acuan untuk mencari perbedaan jarak setiap data sehingga didapatkan nilai rata – rata perubahan jarak GPS pada *quadcopter* dengan GPS *smartphone* user.

Tabel 3. Perbedaan Jarak Quadcopter dengan Marker

Data ke-	Perbedaan Jarak (mm)
1	30
2	47
3	50
4	52
5	54
Rata-rata	40

Pengujian ini dilakukan untuk mengetahui hasil dari rata –rata jarak perubahan lokasi GPS pada *quadcopter* terhadap *smartphone* user berupa bulat merah. Data ke-1 dijadikan titik acuan untuk mencari perbedaan jarak setiap data sehingga didapatkan nilai rata – rata perubahan jarak GPS pada *quadcopter* dengan GPS *smartphone* user. Berikut ini ditampilkan gambar perbedaan jarak antara *quadcopter* dengan *marker* menggunakan pengukur jarak yaitu meteran.



Gambar 9. Selisih 0.4 meter pada Quadcopter terhadap Marker

Dibawah ini adalah *print script* yang ditampilkan oleh Raspberry Pi 3B menggunakan bahasa pemrograman python yang dihubungkan dengan protokol mavlink untuk berkomunikasi dan mengambil parameter yang ada di *flight controller* pada quadcopter dengan *library* dronekit dan OpenCV.

```
File Edit Format Bun Options Windows Help
Python 2.7.9 Shell
File Edit Shell Debug Options Windows Help
WAITING LAUNCH FROM USER
WAITING LAUNCH FROM USER
WAITING LAUNCH FROM USER
READY TO LAUNCH
Basic pre-arm checks
Arming motors
Waiting for arming...
Taking off!
Altitude: -0.01
Altitude: -0.015
Altitude: -0.039
Altitude: -0.041
Altitude: 1.929
>>> EKFX IMU1 in-flight yaw alignment complete
>>> EKFX IMU0 in-flight yaw alignment complete
Altitude: 4.101
Altitude: 4.716
Altitude: 4.86
Reached target altitude
Set default/target airspeed to 1
Going towards destination.
Latitude: -6.9281023
Longitude: 107.4743465
Altitude: 4.936
finish
Finished
Landing
Latitude: -6.9280662
Longitude: 107.4742289
Altitude: 5.13
Close vehicle object
Latitude: -6.9280745
Longitude: 107.4742253
Altitude: 4.234
```

Gambar 10. *Print script* parameter yang ditampilkan di *microcomputer*



Gambar 11. Kamera Pi Melakukan Pengenalan *Marker*

Raspberry Pi sebagai *microcomputer* memproses dan mengolah gambar berupa *marker* bulat merah menggunakan *software* OpenCV. Kemudian Quadcopter melakukan misi pendaratan pada *marker* bulat merah menggunakan mode landing yaitu `vehicle.mode =`

`VehicleMode("LAND")` yang didapat dari *library* dronekit.

5. KESIMPULAN

Berdasarkan pengujian yang dilakukan terhadap sistem yang dibangun selama proses penyusunan penelitian ini dapat ditarik beberapa kesimpulan yaitu:

- Quadcopter dapat menjalankan misi waypoint dari titik A ke titik B dengan menggunakan pengenalan *marker* didapat perubahan jarak rata-rata lokasi *marker* sebesar 40 milimeter.
- Pengiriman data dengan menggunakan internet dengan rata – rata waktu 2,74 sekon untuk setiap pengiriman data.

UCAPAN TERIMA KASIH

Penulis mengucapkan banyak terimakasih kepada pembimbing dan pihak-pihak yang telah membantu dalam penyelesaian paper ini.

DAFTAR PUSTAKA

- [1] Rahman, Aji Kurnia. 2018. “Rancang Bangun dan Implementasi Kendali Quadcopter Melalui Jaringan Internet Berbasis Lokasi Menggunakan Smartphone Android”, Jurusan Teknik Mesin dan Manufaktur, Politeknik Manufaktur Bandung.
- [2] O. A. Yakimenko, I. I. Kaminer, W. J. Lentz, and P. A. Ghyzel, “Unmanned Aircraft Navigation for Shipboard Landing Using Infrared Vision”, IEEE Transactions on Aerospace Electronic Systems. Oct 2002, Volume 38, pp. 1181-1200.
- [3] Pavol Fedor, Daniela Perdukova, “Autonomous Flying with Quadcopter Using Fuzzy Control and Aruco Markers”, Intelligent Service Robotics. July 2017, Volume 10, Issue 3, pp 185–194.
- [4] Yingcai Bi, dan Haibin Duan “Implementation of Autonomous Visual Tracking and Landing for Low Cost Quadrotor”, *Optik-International Jurnal Light Electron Optics*, 2013.
- [5] Jonathan D. Blythe, Krzysztof A. Borowicz, dan Alyssa N. Hollander, “Autonomous Quadrotor Navigation and Guidance”, Jurusan Teknik Mesin Worcester Polytechnic Institute, 2016.
- [6] Andrew Hansen, “Using Waypoints To Do More With Your Drone”, 21 December 2016. [Online]. Available: <https://www.autelrobotics.com/blog/usingwaypoints-to-do-more-with-your-drone/>.
- [7] Doxygen, “Contours : Getting Started”, 18 Desember 2015. [Online]. Available: https://docs.opencv.org/3.1.0/d4/d73/tutorial_py_contours_begin.html.

