

## PERANCANGAN DAN IMPLEMENTASI *AUTONOMOUS QUADCOPTER* DENGAN KEMAMPUAN *FOLLOW ME* YANG TERINTEGRASI PADA ANDROID

### DESIGN AND IMPLEMENTATION OF ANDROID - BASED "FOLLOW ME" APPLICATION ON AN *AUTONOMOUS QUADCOPTER*

Tedy Zulkarnain<sup>1</sup>, Erwin Susanto, S.T., M.T., PhD<sup>2</sup>, Agung Surya Wibowo, S.T., M.T.<sup>3</sup>

<sup>123</sup>Prodi S1 Teknik Elektro, Fakultas Teknik Elektro, Universitas Telkom

<sup>1</sup>[tedyzulkarnain@gmail.com](mailto:tedyzulkarnain@gmail.com), <sup>2</sup>[erwinelektro@telkomuniversity.ac.id](mailto:erwinelektro@telkomuniversity.ac.id), <sup>3</sup>[agungsw@telkomuniversity.ac.id](mailto:agungsw@telkomuniversity.ac.id)

#### Abstrak

Pada saat ini perkembangan sistem *autonomous* digunakan pada sistem wahana bertipe *rotary-wing*. Salah satu jenisnya yang berkembang saat ini adalah *quadcopter*. *Quadcopter* adalah sebuah wahana terbang dengan empat buah motor yang tergolong dalam UAV (*Unmanned Aerial Vehicle*) atau *drone*. *Drone* dapat diimplementasikan untuk *aerial photography*, *aerial mapping*, membawa muatan (*payload*) dan pantauan udara. Tetapi, terdapat kekurangan dalam hal *aerial photography* yaitu pada umumnya *quadcopter* masih dikendalikan oleh seorang pilot dengan menggunakan sebuah *remote control* dan juga mengalami kesulitan dalam hal mengambil foto atau video untuk objek yang sedang bergerak.

Oleh karena itu, dibuatlah sebuah wahana yaitu *autonomous quadcopter* dengan kemampuan *follow me* yang terintegrasi pada Android. Jadi, *quadcopter* dapat dikendalikan secara otomatis menggunakan aplikasi Android sebagai antar muka pengguna dimana komunikasinya menggunakan modul *wireless 3DR* dengan frekuensi 433 MHz, tanpa menggunakan *remote control* dan bisa mengikuti objek yang bergerak dengan cara melacak keberadaan GPS (*Global Positioning System*) pada Android yang diletakkan atau dibawa oleh objek tersebut.

Hasil tugas akhir ini merupakan suatu sistem kontrol gerak *quadcopter* yang terintegrasi pada sistem *mobile monitoring* untuk memudahkan para pengguna dalam melakukan proses pengambilan gambar pada suatu objek yang bergerak melalui sebuah perangkat *smartphone* berbasis android.

Kata kunci : *autonomous, quadcopter, drone, follow me, GPS, Android*

#### Abstract

At this time the development of *autonomous systems* is used in system type *rotary-wing vehicle*. One of it's kind developed recently is *quadcopter*. *Quadcopter* is a flying vehicle with four motorcycles that is one of member of UAV (*Unmanned Aerial Vehicle*) or *drones*. *Drones* can be implemented for *aerial photography, aerial mapping, carrying a payload and monitoring*. However, there are disadvantages for *aerial photography*. They are a *quadcopter* still controlled manually by a pilot that using *remote controll* and also have difficulty to take a picture or video for moving objects.

Therefore, we invented *autonomous quadcopter* has abilities to *follow me* that integrated to *android*. So, the *quadcopter* can be controlled automatically using *android application* as a user interface that using a *wireless module 3DR* with frequency 433 MHz for it's communication, without using the *remote control* and can following a moving object by tracking the *GPS (Global Positioning System)* od *Android device*.

The results of this final assignment is *motion control system* of a *quadcopter* that integrated in the *mobile monitoring system* to facilitate the users for take a picture or video on a moving object use a *smartphone* automatically based on *Android*.

Keywords: *autonomous, quadcopter, drones, follow me, GPS, Android*

**1. Pendahuluan**

Pada saat ini perkembangan sistem *autonomous* digunakan pada sistem wahana bertipe *rotary-wing*. Salah satu jenis *rotary-wing* yang berkembang saat ini adalah *quadrotor* atau *quadcopter*.<sup>[1]</sup> *Quadcopter* adalah sebuah wahana terbang dengan empat buah motor yang tergolong dalam UAV (*Unmanned Aerial Vehicle*) dimana sistem kendali dan penggeraknya menggunakan *Flight Controller* yang sudah terdapat algoritma dan sensor – sensor di dalamnya. Salah satu teknologi yang ada di *quadcopter* yaitu GPS (*Global Positioning System*) yang berfungsi untuk melacak

koordinat dan lokasi sehingga wahana dapat mempertahankan posisinya pada saat terbang. Dengan kemampuannya tersebut, maka *quadcopter* dapat diimplementasikan untuk *aerial photography*, *aerial mapping*, membawa muatan (*payload*) dan pantauan udara.

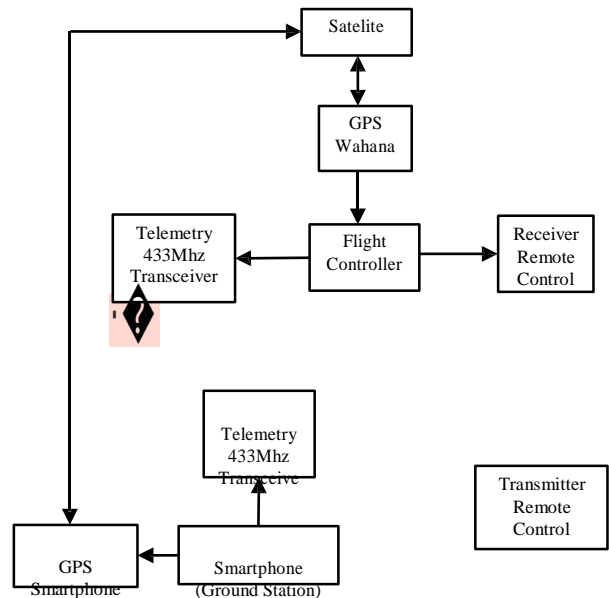
Dari keunggulan yang telah disebutkan diatas, terdapat kekurangan atau masalah dalam hal *Aerial*

*Photography* yaitu pada umumnya *quadcopter* masih dikendalikan oleh seorang pilot dengan menggunakan sebuah *remote control* dan juga mengalami kesulitan dalam hal mengambil foto atau video untuk objek yang sedang bergerak. Berdasarkan masalah tersebut, maka penulis berkeinginan untuk merancang dan mengimplementasikan sebuah wahana yaitu *Autonomous Quadcopter* dengan kemampuan *Follow Me* yang terintegrasi pada Android.

Jadi, *quadcopter* dapat dikendalikan secara otomatis menggunakan Android tanpa menggunakan *remote control* dan bisa mengikuti objek yang bergerak dengan cara melacak keberadaan Android yang ditaruh atau dipegang pada objek tersebut. Adapun sistem kontrol yang dirancang yaitu *quadcopter* sebagai wahana, *Ardu Flyer* sebagai *Flight Controller* dan *interface*, sinyal radio 433 MHz yang terhubung ke protocol Mavlink sebagai media komunikasi, serta menggunakan aplikasi Android sebagai *platform* untuk si pengguna.

**2. Dasar Teori dan Perancangan**

Adapun diagram blok sistem secara umum bisa dilihat pada gambar berikut :



**Gambar 2. 1** Diagram Sistem *Follow Me*

Gambaran umum alat secara garis besar bisa dilihat pada gambar 2.1. *flight controller* yang terdapat pada wahana (*quadcopter*) mengontrol keseluruhan dari mode terbang sehingga bisa terbang secara *autonomous* dan *Follow Me*. Komponen – komponen yang terhubung pada *flight controller* yaitu GPS (*Global Positioning System*), *telemetry* 433 MHz *transmitter*, dan *receiver remote control*.

Cara kerja dari sistem tersebut adalah GPS pada wahana mengirimkan data lokasi berupa koordinat ke satelit. Begitu juga satelit memberikan balasan ke GPS pada wahana. Data tersebut diolah di dalam *Flight Controller*, lalu *Telemetry* 433 MHz *Transmitter* mengirimkan informasi data yang terkoneksi pada *Flight Controller* ke receivernya yang sudah terhubung ke *smartphone*. Untuk mengendalikan wahana menggunakan *smartphone* digunakan aplikasi Android. Aplikasi ini dapat menampilkan gambar lokasi dari wahana dan si pengguna yang sudah dikirimkan oleh satelit. Hal itu bisa terjadi karena GPS pada *smartphone* mengirimkan data berupa lokasinya kepada satelit, begitu juga satelit merespon dan mengirimkan data ke GPS pada *smartphone*. Jadi peran GPS dan satelit pada mode *Follow Me* ini sangat besar, karena selama mode ini dijalankan, koneksi antara GPS dan satelit tidak boleh terputus, karena yang dapat membuat si wahana mengikuti si penggunanya adalah hubungan antara GPS wahana dan GPS *smartphone* ke satelit. Jika semua komponen dari sistem diatas sudah terkoneksi maka pengendalian wahana dengan *smartphone* sudah bisa digunakan.

Adapun peran dari receiver remote control yang terkoneksi pada *Flight Controller* sebagai sistem *Fail Safe* (pengaman) jika terjadi hilangnya kendali si wahana dan *remote control* bisa langsung mengambil alih kontrol wahana yang tadinya dikontrol oleh *smartphone*. Jadi dalam mode *Follow Me* ini masih tidak lepas dari peranan *remote control*.

### 2.1 UAV (*Unmanned Aerial Vehicle*)

Kendaraan udara tak berawak atau *Unmanned Aerial Vehicle* (UAV) adalah sebuah mesin terbang yang dikendalikan jarak jauh oleh pilot atau secara otomatis dengan prinsip terbang memanfaatkan hukum aerodinamika. UAV dapat digunakan untuk membawa muatan seperti senjata, kamera, dan muatan lainnya. UAV dibagi atas tiga kategori berdasarkan pada struktur sayap dan badan yaitu *fixed wing*, *flapping wing* dan *rotary wing*.

Jenis *fixed wing* memiliki bentuk sayap tetap seperti pada pesawat terbang komersial. Jenis *flapping wing* menggunakan sayap membrane untuk dapat terbang. Sedangkan jenis *rotary wing* memiliki sayap yang berputar berupa baling – baling yang digunakan sebagai gaya angkat, seperti *helicopter* dan jenis *multirotor* yang lain. Meskipun jenis *rotary wing* memiliki kekurangan pada kecepatannya disbanding dengan UAV jenis *fixed wing*, akan tetapi UAV jenis *rotary wing* mempunyai banyak keunggulan diantaranya kemampuan lepas landas dan mendarat secara vertikal, atau biasan disebut dengan *Vertical Take Off Landing* (VTOL). Hal ini memungkinkan UAV jenis ini dapat terbang di tempat sulit sekalipun, melakukan *hover* (melayang) secara diam pada titik tertentu sehingga dapat digunakan sebagai alat untuk pengawasan (*surveillance*), berbeda dengan UAV jenis *fixed wing* yang membutuhkan landasan pacu yang luas untuk proses lepas landas dan mendarat. [2]

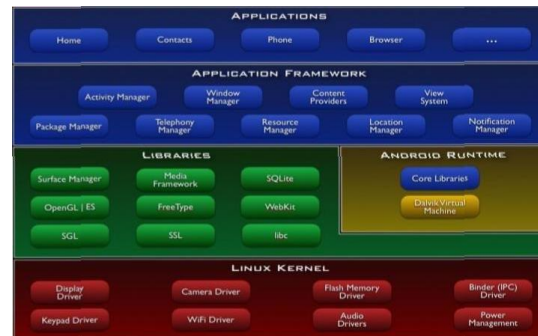
### 2.2 Mode Terbang *Follow Me*

Mode *Follow Me* adalah fitur yang sangat keren. Kemampuan ini sangat berguna untuk orang - orang yang suka melakukan aktivitas seperti olahraga *action*, *aerial photography*, dan lain – lain. Gambar yang dihasilkan yaitu penglihatan orang ketiga (*third-person view*). [3] *Follow Me* mengikuti si pengguna dengan membawa kamera seperti *paparazzi* terbang. Wahana (*drone*) diintegrasikan pada *smartphone*, lalu wahana mendeteksi lokasi *smartphone* dan mengikuti kemanapun kita pergi. Adapun kekurangan dari mode ini yaitu, GPS (*Global Positioning System*) pada *smartphone* yang mengendalikan wahana tidak terlalu akurat dalam mendeteksi lokasi. Tetapi kekurangan ini tidak terlalu berpengaruh terhadap terbang wahana

dalam mode *Follow Me* [4]

### 2.3 Android

Android merupakan sistem operasi untuk telepon seluler yang berbasis *Linux*. Android memiliki berbagai keunggulan sebagai *software* yang memakai basis kode komputer yang bisa didistribusikan secara terbuka (*open source*) sehingga pengguna bisa membuat aplikasi baru di dalamnya. Android memiliki aplikasi native Google yang terintegrasi seperti *pushmail Gmail*, *Google Maps*, dan *Google Calendar*. [5] Sistem operasi Android memiliki 4 lapisan (*layer*) yang merupakan komponen sistem Android. Gambar berikut merupakan lapisan arsitektur sistem operasi Android:



Gambar 2. 2 Arsitektur Android<sup>[5]</sup>

a. *Applications and Widgets*  
*Application and Widgets* ini adalah layer di mana *user* berhubungan dengan aplikasi saja, di mana biasanya *download* aplikasi dilakukan kemudian dilakukan instalasi dan menjalankan aplikasi tersebut. Di layer ini juga terdapat seperti aplikasi inti dari Android termasuk klien email, program SMS, kalender, *map*, *browser*, kontak, dan yang lainnya. Semua aplikasi tersebut ditulis menggunakan program Java. [5]

b. *Application Frameworks*  
 Android adalah “*Open Development Platform*” yaitu Android menawarkan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi *resources*, menjalankan *service background*, mengatur alaram, dan menambah status *notifications*, dan sebagainya. Sehingga dapat disimpulkan bahwa *Application Frameworks* merupakan layer dimana para pengguna android dapat melakukan pembuatan serta pengembangan aplikasi yang akan dijalankan pada sistem [5]

android.

c. *Libraries*

Libraries adalah layer di mana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya. Berjalan di atas kernel, Layer ini meliputi berbagai library C/C++ inti seperti Libc dan SLL, dan lain-lain. Beberapa *libraries* di antaranya:

- *Media Library* untuk memutar dan merekam berbagai macam format audio dan video.
- *Surface Manager* untuk mengatur hak akses layer dari berbagai aplikasi.
- *Graphic Library* termasuk didalamnya *SGL* dan *OpenGL*, untuk tampilan 2D dan 3D.
- *SQLite* untuk mengatur relasi database yang digunakan pada aplikasi.
- *SSL* dan *WebKit* untuk *browser* dan keamanan internet.<sup>[5]</sup>

d. *Android Run Time*

Pada *android* tertanam paket *libraries* inti yang menyediakan sebagian besar fungsi *android*. Inilah yang membedakan *Android* dibandingkan dengan sistem operasi lain yang juga mengimplementasikan *Linux*. *Android Runtime* merupakan mesin virtual yang membuat aplikasi *android* menjadi lebih tangguh dengan paket pustaka yang telah ada. Dalam *Android Runtime* terdapat 2 bagian utama, diantaranya:

- **Pustaka Inti**, *Android* dikembangkan melalui bahasa pemrograman Java, tapi *Android Runtime* bukanlah mesin virtual

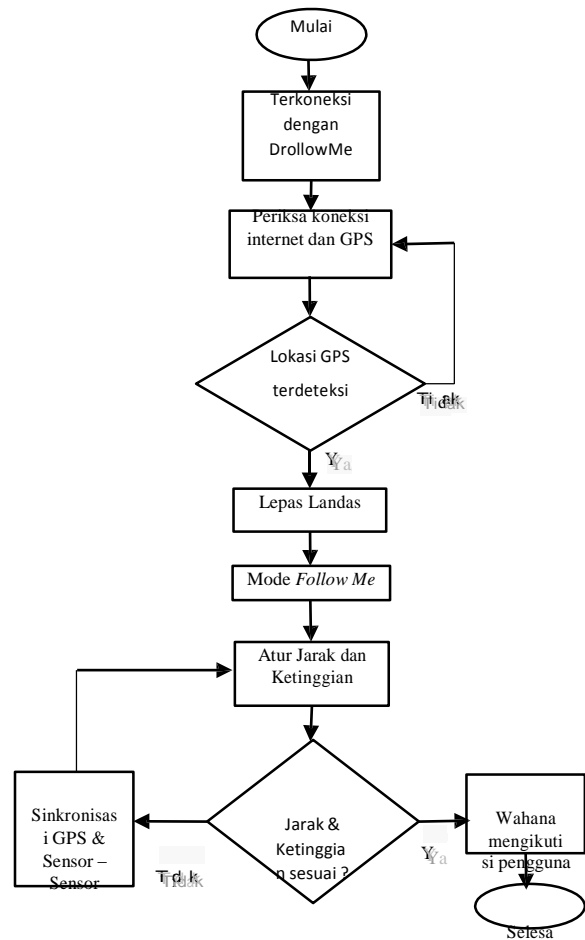
Java. Pustaka inti *android* menyediakan hampir semua fungsi yang terdapat pada pustaka Java serta beberapa pustaka khusus *android*.

- **Mesin Virtual Dalvik**, *Dalvik* merupakan sebuah mesin virtual yang dikembangkan oleh Dan Bornstein yang terinspirasi dari nama sebuah perkampungan yang berada di Iceland. *Dalvik* hanyalah interpreter mesin virtual yang mengeksekusi file dalam format *Dalvik Executable (\*.dex)*.<sup>[5]</sup>

e. *Linux Kernel*

*Linux kernel* adalah layer dimana inti dari operating sistem dari *Android* itu berada. Berisi file-file sistem yang mengatur *processing, memory, resource, drivers*, dan sistem-sistem operasi *Android* lainnya. *Linux kernel* yang digunakan *Android* adalah *linux kernel release 2.6*.<sup>[5]</sup>

2.4 Perancangan Sistem



Gambar 2.3 Flowchart Sistem

Gambar 2.3 menjelaskan tentang bagaimana cara kerja dari sistem dan *software* yang digunakan untuk mode *Follow Me*. Aplikasi *Android* sangat dibutuhkan dalam sistem ini. Pada tahap pertama koneksikan wahana dengan aplikasi *Android* yang sudah terpasang pada *smartphone*. Komunikasi wahana dengan *smartphone* menggunakan komponen *Mav-Link Protocol 433 MHz Transceiver*. Setelah wahana terkoneksi dengan aplikasi, lalu periksa sambungan *GPS (Global Positioning System)* pada wahana dan *smartphone*. Koneksi internet dibutuhkan saat proses ini. Lalu masuk kasus tentang pendeteksian lokasi

GPS. Jika tidak terdeteksi, maka ulangi proses pemeriksaan koneksi internet dan GPS. Jika terdeteksi, maka wahana siap lepas landas dengan menekan tombol *Take Off* pada aplikasi Android.

Setelah wahana berhasil lepas landas, tekan tombol *Follow Me* untuk masuk ke mode *Follow Me*. Di mode ini, aplikasi Android akan menampilkan pilihan untuk mengatur jarak antara wahana dengan *smartphone* dan ketinggian antara wahana dengan tanah. Atur jarak dan ketinggian sesuai yang diinginkan. Jika jarak dan ketinggian tidak sesuai dengan nilai yang dimasukkan, maka sistem akan mensinkronisasikan GPS dan sensor – sensor yang terdapat di dalam *Flight Controller* yaitu sensor *altitudemeter* dan *barometer*. Tetapi jika sesuai, mode *Follow Me* berhasil dijalankan dan wahana akan mengikuti si pengguna yang membawa *smartphone*

kemanapun dia pergi.

**2.4 Perancangan Mekanik**

*Frame* wahana yang digunakan berbahan dasar PCB (*Printed Circuit Board*) jenis FR-4, dan lengannya menggunakan aluminium batang sejumlah 4 buah dan dipasang motor *brushless* pada masing – masing lengannya. *Landing gear* menggunakan plat aluminium. Untuk *propeller* nya menggunakan ukuran 10x5. ESC (*Electronic Speed Controller*) yang digunakan yaitu Turnigy 30 A. Dan menggunakan baterai yang berkapasitas 4 Cell, 4000 mah.

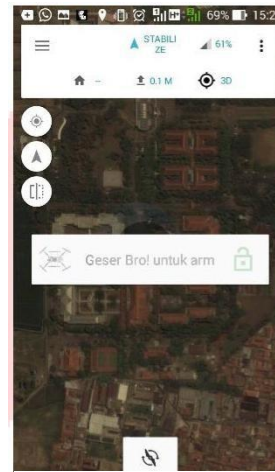


**Gambar 2. 4** Wahana yang Digunakan

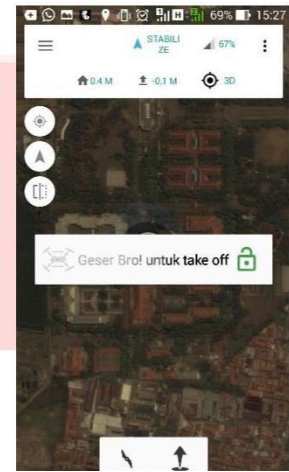
**2.6 Perancangan Software Android Menggunakan Aplikasi Android Studio**

Aplikasi Android yang digunakan bernama Drollowme. Aplikasi ini dibuat menggunakan *software* Android Studio. Drollowme berfungsi sebagai antarmuka pengguna dengan wahana, dan juga dapat mengontrol wahana dalam mode *arm*, *take off*, *follow me*, *landing* dan *disarm*. Tampilan *map* pada aplikasi ini menggunakan *library* dari *Google Maps*

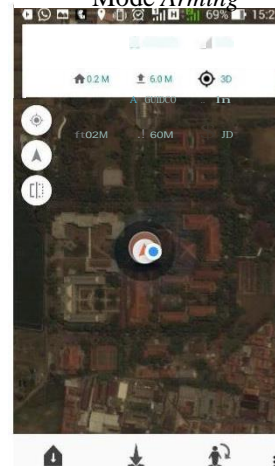
dengan memasukkan API (*Application Programming Interface*) Key.



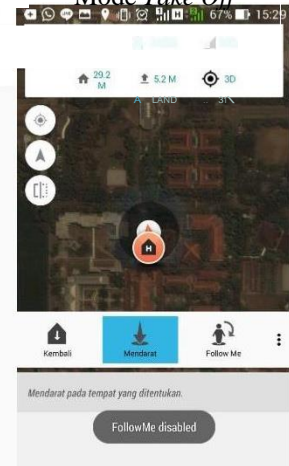
**Gambar 2. 5** Tampilan Mode Arming



**Gambar 2. 6** Tampilan Mode Take Off



**Gambar 2. 7** Tampilan Mode Follow Me



**Gambar 2. 8** Tampilan Mode Landing

**3. Uji Coba dan Analisa**

**3.1 Pengujian Jarak Operator Terhadap Wahana**

Pada tahap ini operator mengoperasikan aplikasi android yang telah dibuat, pertama masuk mode *arm* dan terbang. Setelah wahana lepas landas dengan ketinggian yang telah ditentukan yaitu 6 meter, lalu masuk ke mode *follow me*. Wahana akan mendeteksi letak GPS pada perangkat *smartphone* operator. Kemudian masukkan jarak yang diinginkan. Adapun data uji jarak terbang wahana dengan operator sebagai berikut :

**Tabel 3.1** Hasil Uji Jarak Wahana Terhadap Operator

No. Pengujian	Nilai Jarak Masukan (m)	Jarak Sesungguhnya (m)	Selisih Jarak (m)
1	6	5	1
2	6	4,43	1,57
3	6	7	1
4	6	5,8	0,2
5	6	4,8	1,2

Berdasarkan data hasil uji coba diatas, jarak yang dimasukkan pada aplikasi android tidak sesuai dengan jarak sesungguhnya, terjadi selisih jarak sekitar 1 meter. Ini dikarenakan GPS wahana dan operator terdapat keterbatasan, selain itu memiliki nilai toleransi akurasi penetapan lokasi secara horizontal sebesar 0,9 meter dilihat dari data *sheet* GPS Ublox Neo M8N. Sehingga mengakibatkan ketidak sesuaian nilai jarak yang dimasukkan dengan nilai jarak sesungguhnya.



**Gambar 3. 1** Pengujian Jarak Wahana Terhadap Operator

Pada gambar 3.1 operator memasukkan nilai jarak yaitu 6 meter, tetapi setelah diukur menggunakan alat ukur nilai jarak sesungguhnya antara operator dan wahana yaitu 4,8 meter. Terjadi selisih sekitar 1,2 meter.

**3.2 Pengujian Akurasi GPS**

Pada tahap ini, setelah wahana memasuki mode *follow me*, operator akan bergerak sesuai jalur yang ditetapkan. Wahana akan mendeteksi letak GPS operator dan secara otomatis mengikuti pergerakan dari operator. Dalam tahap pengujian ini akan dilihat kesesuaian jalur yang dilewati operator dengan wahana.



**Gambar 3. 2** Pengujian Akurasi GPS Pada Saat Wahana Lepas Landas



**Gambar 3. 3** Pengujian Akurasi GPS, Wahana Mengikuti Operator



**Gambar 3. 4** Pengujian Akurasi GPS, Wahana Mengikuti Jalur Operator

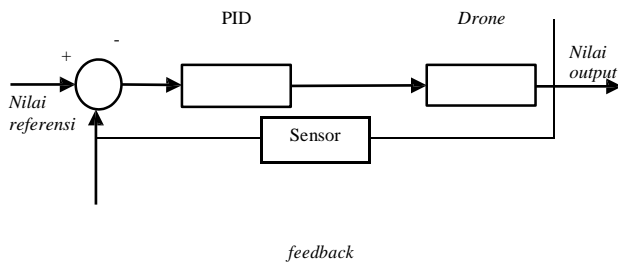
Pada gambar – gambar diatas, dapat dilihat jalur operator yang berwarna hijau dan jalur wahana yang berwarna ungu. Operator ditandai dengan lambang berbentuk balon berwarna hijau yang bertuliskan *Guided Mode* dan wahana ditandai dengan lambang berbentuk *quadcopter*. Setelah melakukan

beberapa pengujian, jalur terbang wahana selama mengikuti operator tidak terlalu tepat dengan jalur operator. Terdapat beberapa meter jarak antara jalur hijau dan ungu. Tetapi wahana berhasil mengikuti operator dengan sukses.

**3.3 Analisis PID**

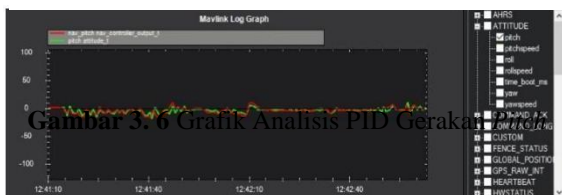
Selama wahana terbang, peran sistem kontrol sangat penting diperhatikan. Sistem kontrol yang digunakan adalah PID. Data PID bisa dilihat dan diamati berdasarkan data log yang diunduh dari perangkat *flight controller*. Nilai PID sudah dimasukkan pada saat proses *auto tuning*. Sehingga data PID yang diamati yaitu respon wahana ketika gerakan *roll*, *pitch* dan *yaw*. Secara garis besar,

diagram blok sistem kontrol wahana sebagai berikut :



**Gambar 3. 5** Diagram Blok Sistem PID Pada Wahana

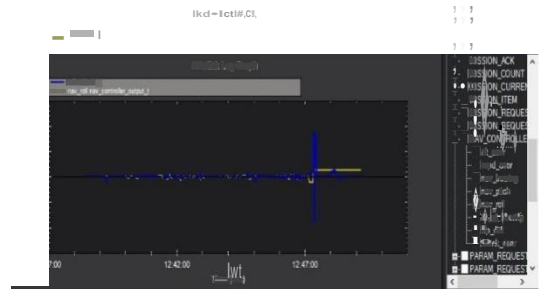
Pada diagram blok sistem PID diatas, menjelaskan bahwa cara kerja dari menganalisis nilai PID adalah nilai yang sudah ditetapkan dari proses *auto tuning* sebelumnya disimpan pada *flight controller*, lalu pada saat wahana diterbangkan nilai – nilai PID pada saat terbang akan disesuaikan dengan nilai – nilai PID yang sudah ditetapkan / nilai referensi. Sensor – sensor yang berperan sebagai *feedback* yaitu sensor *gyroscope*, *barometer* dan *accelerometer*. Adapun hasil pengujian analisis PID ditunjukkan berupa grafik – grafik dibawah ini :



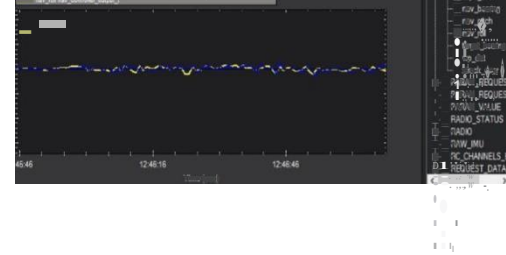
**Gambar 3. 7** Grafik Analisis PID Gerakan *Pitch* Acuan Terhadap Nilai Sebenarnya

Pada gambar grafik *pitch* diatas menjelaskan bahwa, garis merah adalah nilai acuan untuk gerakan *pitch*, dan garis hijau adalah nilai sebenarnya ketika wahana terbang. Nilai acuan didapat dari pengaturan PID wahana pada saat *auto tuning*. Sudut referensi yaitu 0°. Jadi, dilihat dari data grafik diatas bahwa PID wahana ketika terbang sudah bagus dan sesuai, karena garis pada grafik saling berhimpit dan sedikit mengalami *error*.

Pada gambar grafik *roll* diatas menjelaskan



**Gambar 3. 8** Grafik Analisis PID Gerakan *Roll*



bahwa, garis biru adalah nilai acuan untuk gerakan *roll*, dan garis kuning adalah nilai sebenarnya ketika

**Gambar 3. 9** Grafik Analisis PID Gerakan *Roll* Acuan Terhadap Nilai Sebenarnya

wahana terbang. Nilai acuan didapat dari pengaturan PID wahana pada saat *auto tuning*. Sudut referensi yaitu 0°. Jadi, dilihat dari data grafik diatas bahwa PID wahana ketika terbang sudah bagus dan sesuai, karena garis pada grafik saling berhimpit dan sedikit mengalami *error*.

**4. Kesimpulan**

Berdasarkan hasil perancangan dan pengujian dari tugas akhir ini, maka dapat diambil kesimpulan antara lain :

1. Berdasarkan hasil pengujian jarak antara operator dan wahana dapat disimpulkan bahwa, nilai jarak yang dimasukkan tidak sesuai dengan nilai jarak sesungguhnya setelah diukur dengan alat ukur. Ini dikarenakan adanya keterbatasan nilai akurasi penetapan lokasi GPS secara horizontal jenis Ublox Neo M8N yaitu sebesar 0,9 meter. Persentase akurasi GPS terhadap pengujian jarak sebesar 80%.
2. Berdasarkan hasil pengujian akurasi GPS dapat disimpulkan bahwa, jalur

pergerakan operator dengan wahana sudah sesuai. Hanya saja masih terdapat sedikit jarak yang keluar dari jalur operator. Dilihat dari salah satu data koordinat GPS *latitude* wahana adalah -6.974261000 dan *latitude smartphone* adalah -6.974260000 serta nilai *longitude* wahana adalah 107.632245400 dan *longitude smartphone* adalah 107.632245700, terdapat selisih nilai koordinat antara kedua perangkat. Hal ini yang mengakibatkan jalur terbang wahana tidak terlalu tepat dengan jalur operator. Persentase akurasi GPS terhadap pengujian jalur terbang sebesar 90%.

3. Berdasarkan analisis PID dapat disimpulkan bahwa, nilai PID yang sudah diatur dengan mode *auto tuning* dengan nilai PID pada saat terbang sudah sesuai dan sedikit terdapat *error*. Dan dilihat dari parameter *nav\_pitch*, *nav\_roll*, *aspd\_error* dan *xtrack\_error* nilai menunjukkan angka 0. Tetapi pada parameter *alt\_error* nilai menunjukkan angka  $\pm 2.5$ . Nilai ini masih tergolong nilai *error* yang kecil, karena batas aman *alt\_error* adalah 0 sampai -5. Jadi kondisi wahana pada saat melakukan misi *follow me* sudah cukup stabil.
4. Aplikasi android *DrollowMe* sudah berhasil sebagai *interface* yang mengintegrasikan *quadcopter* dengan si pengguna menggunakan mode terbang *follow me* secara *autonomous* yang mempunyai tampilan sederhana, sehingga memudahkan pengguna untuk mengaplikasikannya.

## 5. Daftar Pustaka

- [1] Utama Wiguna, Rizky, M. Komarudin dan Agus Trisanto. 2013. Sistem Kendali *Holding Position* Pada *Quadcopter* Berbasis Mikrokontroler ATmega 328P. Bandar Lampung. Universitas Lampung
- [2] Budjiman, Arrief. 2014. Perancangan dan Implementasi *Autonomous Landing Quadcopter* Menggunakan Kontrol PID. Bandung. Universitas Telkom
- [3] <http://gizmodo.com/new-autonomous-3d-robotics-drone-follows-you-wherever-y-1631694870> (diakses pada tanggal 14 April 2015)
- [4] <http://www.wsj.com/articles/the-drones-on-autopilot-that-follow-your-lead->

usually-1419357666 (diakses pada tanggal 14 April 2015)

- [5] Dwi Rahmantio, Ikhsan. 2014. Perancangan dan Realisasi Sistem Kontrol Gerak Pada Perangkat *Mobile Monitoring* Berbasis Android. Bandung. Universitas Telkom