

PERANCANGAN DAN SIMULASI KENDALI OTONOM UNTUK TAKE-OFF DAN LANDING PADA QUADCOPTER DRONE

DESIGN AND SIMULATION AUTONOMY CONTROL FOR TAKE-OFF AND LANDING ON QUADCOPTER DRONE

Abdul Manan¹, Agung Nugroho Jati S.T., M.T. ² Randy Erfa Saputra S.T., M.T.³

^{1,2,3} Prodi S1 Teknik Komputer, Fakultas Teknik Elektro, Universitas Telkom

¹ abdulmanan@student.telkomuniversity.ac.id, ² agungnj@telkomuniversity.ac.id, ³ resaputra@telkomuniversity.ac.id

Abstrak

Unmanned Aerial Vehicle (UAV) atau pesawat tanpa awak merupakan salah satu jenis robot udara yang saat ini sedang mengalami perkembangan yang pesat, salah satunya adalah quadcopter. Pada quadcopter dibutuhkan auto take-off dan landing secara otomatis dan presisi pada sistem control auto take-off dan landing yang akurat dipengaruhi PID (Proportional-Integral-Derivative) yang stabil ketika di udara. Untuk waktu yang dibutuhkan quadcopter untuk take-off dan landing pada ketinggian tertentu dengan menggunakan Extended Kalman Filter hasil yang didapat quadcopter lebih stabil pada perubahan grafik plot imu variabel X, Y dan Z pada percobaan simulasi yang digunakan dengan pengaruh kecepatan angin 10km/h hingga 20km/h, quadcopter untuk take-off dan landing hingga terbang lebih stabil dari pada penggunaan tanpa extended kalman filter Hasil pada penelitian tugas akhir ini mendapatkan sistem extended kalman filter dengan performansi kestabilan nilai X, Y dan Z mendekati 0 pada grafik imu quadcopter ketika take-off dan landing pada saat mencapai ketinggian tertentu.

Kata Kunci : *Take-off, Landing, Robot operating system, Extended Kalman Filter*

Abstract

Unmanned Aerial Vehicle (UAV) or unmanned aircraft is one type of air robot that is currently experiencing rapid development, one of which is a quadcopter. In a quadcopter, automatic take-off and landing are required and precise control systems for auto take-off and landing are accurately influenced by PID (Proportional-Integral-Derivative) which is stable when in the air. For the time required for the quadcopter to take-off and landing at a certain height using the Extended Kalman Filter, the results obtained by the quadcopter are more stable in changing the imu plot graphs for variables X, Y and Z in the simulation experiment used with the influence of wind speeds of 10 km / h to 20 km. / h, the quadcopter for take-off and landing to fly is more stable than the use without the extended kalman filter. The results in this final project study an extended kalman filter system with the stability performance of X, Y and Z values close to 0 on the imu quadcopter chart when take-off and landing when it reaches a certain height.

Keywords: *Take-off, Landing, Robot operating system, Extended Kalman Filter*

1. Pendahuluan

Unmanned Aerial Vehicle (UAV) atau yang biasa kita kenal dengan pesawat tanpa awak saat ini sedang mengalami perkembangan yang pesat yaitu untuk keperluan sipil atau keperluan pribadi sekedar hanya untuk hobi atau bahkan digunakan untuk keperluan yang lebih besar seperti pada dunia militer [1]. UAV memiliki banyak tipe salah satunya adalah quadcopter.

Sistem quadcopter adalah sistem nonlinier karena aerodinamika dari empat rotor yang memerlukan algoritma kontrol untuk menstabilkan pada posisi take-off dan landing saat ini. Beberapa peneliti telah menggunakan algoritma kontrol PID untuk menstabilkan quadcopter.

Tujuan utama sistem ini adalah dapat meningkatkan kestabilan dan ke presisian quadcopter saat auto take-off dan auto landing yang dikarenakan target penelitian ini adalah untuk menjaga kestabilan dan keseimbangan pada waktu take-off dan landing pada simulasi dengan faktor external yang ada pada simulasi serta dapat menjaga kestabilan posisi PID pada ketinggian tertentu, penyebab dari tidak stabilnya quadcopter pada ketinggian tertentu adalah adanya perubahan ketinggian dan faktor yang mempengaruhinya.

2. Dasar Teori

2.1 Quadcopter

Quadcopter merupakan salah satu jenis unmanned aerial vehicle (UAV) yang memiliki kemampuan untuk melakukan take-off dan landing secara vertikal atau yang biasa lebih dikenal dengan istilah VTOL (Vertical Take-off Landing) [2].

2.2 Kendali Proportional, Integral, Derivative (PID)

Kontrol PID merupakan suatu tipe kontrol yang paling klasik. PID menerapkan mekanisme umpan balik yang banyak digunakan dalam sistem pengaturan industri. Sebuah kontroler PID menghitung nilai kesalahan sebagai perbedaan antara variabel proses terukur dan set point yang diinginkan [3].

2.3 Sistem Extended Kalman Filter

Dalam teori estimasi, *filter* Kalman yang diperluas (EKF) adalah nonlinear dari *filter* Kalman yang linier tentang perkiraan rata-rata saat ini [4]. Algoritma Extended Kalman Filter (EKF) telah dikembangkan yang menggunakan laju giroskop, akselerometer, kompas, GPS, pengukuran kecepatan udara dan tekanan udara untuk memperkirakan posisi, kecepatan, dan orientasi sudut dari quadcopter [5].

2.4 ROS (Robotic Operating System)

ROS (Robotic Operating System) adalah sebuah framework open-source yang dapat diaplikasikan ke dalam robot. Dengan menggunakan ROS para pengembang robotika dapat dengan mudah untuk mengembangkan dan mengimplementasikan ide-idenya dan menghemat banyak sumber daya [6].

2.4.1 Distribusi ROS

Distribusi ROS adalah versi dari kumpulan paket pada ROS, seperti pada distribusi linux. Tujuan dari distribusi ROS adalah agar para pengembang dapat bekerja dengan codebase yang stabil sebab setiap jenis robot memiliki kebutuhan yang berbeda [7].

2.4.2 ROS APIs

APIs (Application Programming Interface system) adalah sekumpulan perintah, fungsi, serta protokol yang dapat digunakan oleh programmer saat membangun suatu sistem. Dengan menggunakan APIs, programmer dapat menggunakan fungsi standar untuk berinteraksi dengan sistem operasi.

2.4.3 ROS Tools

ROS menyediakan beberapa tools yang membantu untuk membuat, membangun, memprogram, dan menjalankan program robotika. Berbagai tools dari visualisasi, plotting, dan masih banyak tools yang lainnya tersedia. Berbagai tools dari visualisasi, plotting, dan masih banyak tools yang lainnya tersedia.

2.4.4 Sensor

Ada beberapa sensor robot yang didukung oleh ROS resmi dan banyak lagi yang didukung oleh ROS. Meskipun tidak komprehensif, sensor fitur didokumentasikan yang stabil [7].

1. Pose Estimation (GPS/IMU)

- *adi_driver: Analog Devices Inc. IMU(ADIS16470) and accelerometer(ADXL345)*
- *Applanix Position and Orientation System for Land Vehicles*
- *Bosch Sensortec BMA180 3-axis accelerometer*

2.5 Komunikasi Mavlink

Protokol Micro Air Vehicle link (MAVLink) adalah open source, protokol jaringan point-to-point yang digunakan untuk membawa telemetri dan untuk memerintahkan dan mengendalikan banyak pesawat tak berawak. Micro Air Vehicle Link Communication Protocol memungkinkan entitas untuk berkomunikasi melalui saluran nirkabel [8].

2.5.1 Default PX4 MAVLink UDP Ports

- Port UDP 14540 digunakan untuk komunikasi dengan API offboard. API offboard koneksi pada port.
- Port UDP 14550 digunakan untuk komunikasi dengan stasiun kontrol darat. GCS diharapkan koneksi pada port ini. QGroundControl port ini secara default.
- TCP Port 4560 digunakan untuk komunikasi dengan simulator. PX4 port ini, dan simulator diharapkan untuk memulai komunikasi dengan data ke port ini [9].

2.6 Mavros

Driver komunikasi untuk berbagai plot otomatis dengan protokol komunikasi MAVLink. menyediakan jembatan UDP MAVLink untuk stasiun kontrol seperti QGroundControl, mavros memungkinkan komunikasi yang panjang saat MAVLink sedang berjalan menggunakan ROS [10].

2.7 Robot_localization

Robot_localization adalah node estimasi keadaan, yang masing-masing merupakan implementasi dari estimator keadaan nonlinier untuk robot yang bergerak dalam ruang 3D. Ini berisi dua keadaan estimasi node, *ekf_localization_node* dan *ukf_localization_node* [11].

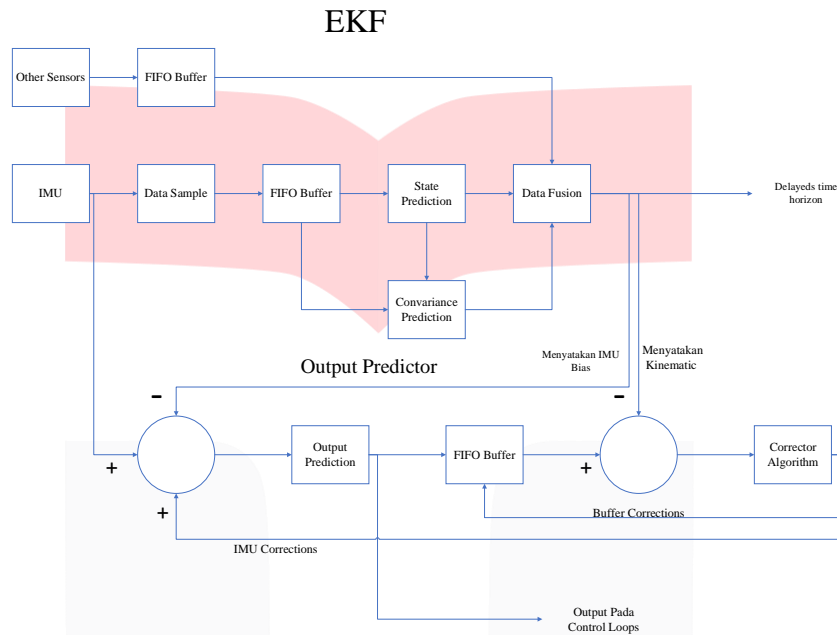
2.8 Odometry

Odometry adalah penggunaan data dari sensor gerak untuk memperkirakan perubahan posisi dari waktu ke waktu. Ini digunakan dalam robotika oleh beberapa robot berkaki atau beroda untuk memperkirakan posisi mereka relatif terhadap lokasi awal [12].

3. Pembahasan

3.1 Diagram Block EKF (Extended Kalman Filter)

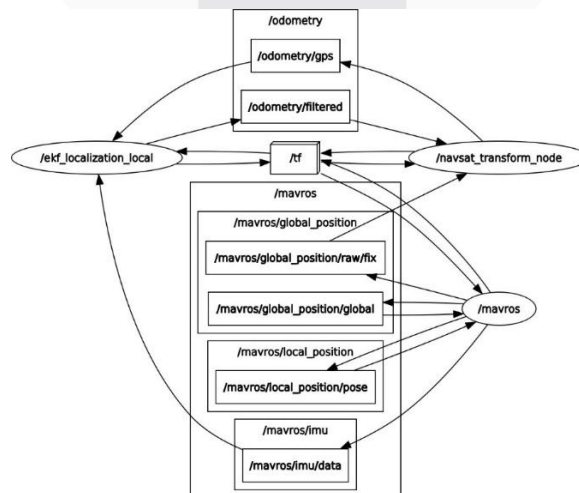
Pada Diagram Block ini menggambarkan *Input*, *Process*, dan *Output* pada Sistem EKF (*Extended Kalman Filter*).



Gambar 1 Diagram Block EKF [13]

3.2 Hubungan Antara Node EKF

Ada 3 node yang berjalan pada ROS yaitu `ekf_localization_local`, `navsat_transform_node`, dan `mavros` untuk menjalankan EKF.



Gambar 2 Grafik node dan topic pada system

4. Implementasi dan pengujian sistem

4.1 Hasil Pengujian EKF pada IMU

A. Pengujian ketinggian 5meter dengan kecepatan angin 10km/h.



Gambar 3 Plot Odometry imu 5m



Gambar 4 Plot Odometry imu EKF 5m

B. Pengujian ketinggian 5meter dengan kecepatan angin 20km/h.



Gambar 5 Plot Odometry imu 5m



Gambar 6 Plot Odometry imu EKF 5m

Pada pengujian pertama EKF take-off dan landing pada IMU Berdasarkan grafik pada Gambar dengan parameter ketinggian berbeda 5 meter dan 10 meter dengan kecepatan angin rata-rata 10km/h dan 20km/h dapat dilihat bahwa jika variabel X,Y dan Z pada pengujian 5 meter dengan kecepatan angin 10km/h dengan kecepatan angin 20km/h pada saat take-off dan landing dengan pengaruh angin tersebut drone menjadi tidak stabil ketika akan take-off atau landing dan memiliki kestabilan lebih tinggi dengan penggunaan EKF. Artinya pada pengujian ketinggian 5 meter dengan kecepatan angin yang berbeda berhasil untuk memperbaiki data imu.

A. Pengujian ketinggian 10meter dengan kecepatan angin 10km/h.



Gambar 7 Plot Odometry imu 10m

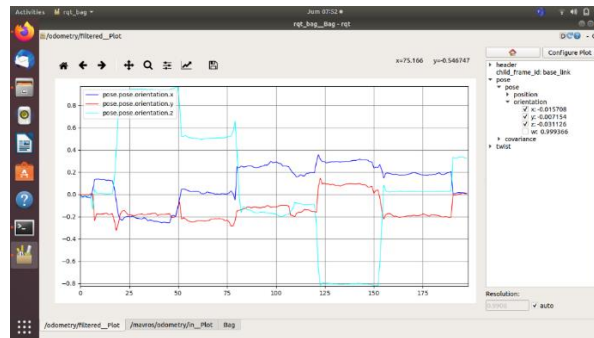


Gambar 8 Plot Odometry imu EKF 10m

B. Pengujian ketinggian 10meter dengan kecepatan angin 20km/h.



Gambar 9 Plot Odometry imu 10m



Gambar 10 Plot Odometry imu EKF 10m

Pengujian 2 dengan ketinggian 10 meter dengan kecepatan angin rata-rata juga sama dengan pengujian sebelumnya masih pada hasil grafik yang di dihasilkan berbeda dengan pengujian ketinggian 5 meter dengan pengaruh angin 10km/h dan 20km/h t drone menjadi lebih tidak satbil ketika akan take-off dan landing dengan terbang lebih tinggi 10 meter dan pada nilai grafik X,Y dan Z mendapatkan accuracy atau kestabilan yang tinggi pada penggunaan EKF saat take-off dan landing. Namun pengujian terbaik adalah pada pengujian pada saat pengujian EKF setelah grafik yang dihasilkan lebih baik pada grafiknya sebelum performansi menggunakan EKF yang tidak stabil pada nilai X,Y dan Z sensor imu.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan pada Tugas Akhir ini, maka dapat ditarik kesimpulan bahwa:

1. Hasil data imu yang dihasilkan dengan penggunaan EKF perbandingan grafik yang dihasilkan cukup berbeda pada ketinggian 5 meter dan 10 meter dengan pengaruh angin pada simulasi dengan kecepatan rata-rata 10km/h dan 20km/h saat tidak menggunakan EKF dengan hasil grafik EKF lebih mendekati 0 pada data plot imu variabel X,Y dan Z.
2. Simulasi menggunakan EKF mampu memberikan respons posisi drone yang lebih stabil dengan mempertahankan posisi X dan Y lebih baik pada take-off dan landing dengan faktor angin yang mempengaruhi.
3. Kontroler PID dengan EKF mampu mempertahankan drone saat take-off dan landing dengan baik namun belum mampu mempertahankan posisi terhadap gangguan angin pada lintasan simulasi pada Qgroundcontrol.

5.2 Saran

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan pada Tugas Akhir ini, maka saran yang dapat diusulkan adalah:

1. Dari pengujian tersebut drone lebih baik ketika take-off dan landing menggunakan data sensor yang sudah di filter dengan Extended Kalman Filter dan perlu beberapa data yang mungkin di filter menggunakan EKF selain GPS dan IMU agar drone dapat bergerak lebih baik dari sebelumnya.
2. Perlu pengujian langsung terhadap drone ketika take-off dan landing dengan kondisi sebenarnya saat drone take-off menggunakan EKF dan tidak menggunakan EKF.

3. Perlu adanya faktor external pada simulasi yang mempengaruhi drone pada saat terbang seperti angin, cuaca dan Interferensi signal.

Daftar Pustaka

- [1] Sustek, Michal Úředníček, Zdeněk, "The Basics of Quadcopter Anatomy," *MATEC Web of Conferences*, vol. 210, pp. 1-7, 2018.
- [2] Lindberg, Robert, "MASTER'S THESIS Autonomous Takeoff and Landing for Quadcopters," 2015.
- [3] Yang, Shaowu Scherer, Sebastian A. Zell, Andreas, "An onboard monocular vision system for autonomous takeoff, hovering and landing of a micro aerial vehicle," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 69, no. 1-4, pp. 499-515, 2013.
- [4] Khazraj, Hesam Faria Da Silva, F. Bak, Claus Leth, "A performance comparison between extended Kalman Filter and unscented Kalman Filter in power system dynamic state estimation," *Proceedings - 2016 51st International Universities Power Engineering Conference, UPEC 2016*, Vols. 2017-Janua, pp. 1-6, 2017.
- [5] Lugo, Jacobo Jiménez Zell, Andreas, "Framework for autonomous on-board navigation with the AR.Drone," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 73, no. 1-4, pp. 401-412, 2014.
- [6] IsaacSaito, "wiki ROS.org," 22 Maret 2016. [Online]. Available: <http://wiki.ros.org/APIs..>
- [7] Foote, Tully, "Ros.org," Open Source Robotics Foundation, 11 June 2020. [Online]. Available: <http://wiki.ros.org/>. [Accessed 11 June 2020].
- [8] Atoev, Sukhrob Kwon, Ki Ryong Lee, Suk Hwan Moon, Kwang Seok, "Data analysis of the MAVLink communication protocol," *2017 International Conference on Information Science and Communications Technologies, ICISCT 2017*, Vols. 2017-Decem, pp. 1-3, 2017.
- [9] Dronecode, "Simulation," Linux Foundation, 17 July 2020. [Online]. Available: <https://dev.px4.io/v1.9.0/en/simulation/#sitl-simulation-environment>.
- [10] Dronecode, "PX4 Dev Team," Dronecode, The Linux Foundation, 17 07 2020. [Online]. Available: <https://dev.px4.io/v1.9.0/en/>.
- [11] Analytics., Charles River, "robot_localization," Sphinx 1.6.7., 2016. [Online]. Available: http://docs.ros.org/melodic/api/robot_localization/html/index.html.
- [12] Mohamed, Sherif A.S.Hagbayan, Mohammad Hashem Westerlund, Tomi Heikkonen, Jukka Tenhunen, Hannu Plosila, Juha, "A Survey on Odometry for Autonomous Navigation Systems," *IEEE Access*, vol. 7, pp. 97466-97486, 2019.
- [13] Riseborough, Paul, "'PX4 State Estimation' PX4 Developer Summit Zurich 2019," in *PX4 Autopilot - Open Source Flight Control*, 2019.