

Perancangan Sistem Lepas Landas Pada Quadrotor UAV Menggunakan Pengenalan Wajah Pemilik

Put Gani Ayub Tamudia, Drs. Elia Kendek Allo, MSc., Sherwin R. U. A. Sompie, ST., MT., Ir. Benefit S. Narasiang
Jurusan Teknik Elektro-FT. UNSRAT, Manado-95115, Email: ganitamudia@yahoo.com

Abstrak—Sebuah kendaraan udara tak berawak quadrotor (UAV) adalah pesawat yang daya angkatnya dihasilkan oleh empat rotor. Pengendalian quadrotor dicapai dengan memvariasikan kecepatan putaran keempat rotor relatif terhadap satu sama lain. Kendali otomatis pada kendaraan udara tak berawak biasanya dicapai menggunakan seperangkat sensor untuk memberikan estimasi posisi, orientasi, dan kecepatan yang dapat diandalkan. Tujuan dari skripsi ini adalah untuk menerbangkan quadrotor setelah wajah dikenali oleh komputer. Metode yang dirancang meliputi sebuah laptop dengan webcam untuk mengenali wajah kemudian signal dikirim ke quadrotor yang telah terpasang accelerometer dan gyroscope untuk mengestimasi orientasi. Algoritma pengenalan wajah diprogram untuk memproses informasi yang dibutuhkan dan selanjutnya sinyal keluaran dikirimkan melalui pemancar agar quadrotor dapat melayang. Hasil estimasi penerbangan dan pengenalan wajah disajikan dengan kelebihan sistem, limitasi, dan kekurangan.

Kata Kunci: UAV, quadrotor, eigenface, accelerometer, gyroscope, kalman filter, proportional-integral-derivative (PID) controller.

I. PENDAHULUAN

Unmanned Aerial Vehicle (UAV) atau wahana udara tak berawak telah banyak terbukti sebagai alat yang bermanfaat bagi masyarakat saat ini. Beberapa aplikasi penting dari UAV antara lain pengawasan dan pengamatan dalam bidang militer, pemetaan wilayah pada bidang lingkungan hidup, dan laporan survey pada daerah bencana alam. Tipe UAV dalam tugas akhir ini berkaitan dengan quadrotor helikopter dimana berbeda dengan helikopter biasa karena mempunyai empat horisontal rotor dan tidak memiliki vertikal rotor. Pengendalian arah quadrotor dilakukan dengan merubah-ubah kecepatan putaran tiap rotor. Untuk membuat pengendalian kestabilan yang dapat diandalkan, maka quadrotor dilengkapi dengan sensor pengukuran inersia yaitu accelerometer dan gyroscope.

Quadrotor yang diimplementasikan pada umumnya dibuat oleh perusahaan komersil, hobi, dan peneliti. Kebanyakan quadrotor dioperasikan menggunakan kendali jarak jauh berbentuk analog maupun digital *joystick*. Bentuk kendali seperti ini masih mengharuskan kendali penuh operator terhadap quadrotor. Untuk alasan ini maka ada dorongan modernisasi quadrotor agar menjadi lebih otonom.

Otonomisasi dapat dicapai dengan memberikan kecerdasan buatan pada quadrotor berupa pengenalan visual, pengenalan suara ataupun sistem stabilitas yang canggih. Dalam skripsi ini saya memfokuskan pada salah satu aplikasi penting dari *computer vision* yaitu pengenalan wajah. Untuk mencapai tujuan yang diinginkan maka quadrotor diharapkan dapat tinggal

landas berdasarkan masukan berupa pengenalan wajah pemilik yang terdeteksi oleh komputer *ground station* dan mampu menstabilkan keadaan (*attitude*) diudara sampai kurun waktu yang diprogram.

II. DASAR TEORI

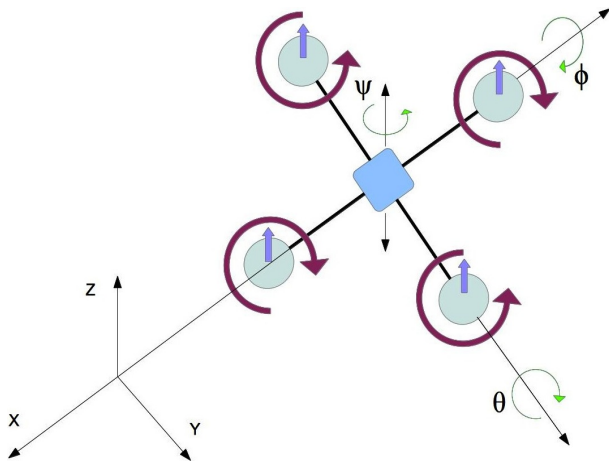
Pada bab ini dijelaskan tentang model dinamik quadrotor, komponen-komponen utama yang diperlukan, metode stabilitas quadrotor, dan sistem pengenalan citra wajah. Pada seksi ini juga akan dibahas beberapa keterampilan matematika yang akan dibutuhkan untuk mengerti proses dan cara membangun algoritma PCA. Pokok-pokok pembicaraan akan dicakup secara independen satu sama lain. Tidak semua teknik ini digunakan dalam PCA, namun teknik yang secara eksplisit tidak digunakan pada hakekatnya menyediakan dasar bagi teknik-teknik yang paling penting.

A. Model Dinamik Quadrotor

Seksi berikut ini telah dirangkum pembahasan mengenai statistika dan matriks algebra yang merupakan bagian paling fundamental dari PCA. Keseluruhan subjek dari statistika didasarkan pada pembelajaran mengenai pengumpulan, organisasi, analisis, interpretasi, dan penyajian data.

Pada bagian ini dijelaskan konsep umum yang digunakan untuk memodelkan quadrotor. Pada Gambar 1 diperlihatkan sistem koordinat yang dipakai dalam rancangan. Manuver pada quadrotor dilakukan dengan mengubah-ubah kecepatan putaran tiap motor. Sebagai ilustrasi pada gambar tersebut diperlihatkan arah dan kecepatan putaran masing-masing motor. Saat quadrotor melayang (*hover*) maka kecepatan putaran disetel sama untuk semua motor.

- *Attitude* quadrotor merupakan sikap orientasi pada titik tengah massa quadrotor. Untuk menaikan atau menurunkan quadrotor secara vertikal, keempat rotor dikontrol dengan kecepatan yang sama. Terdapat tiga kritikal kontrol sistem pada dinamika penerbangan, yaitu *roll*, *pitch*, dan *yaw*.
- *Roll* (ϕ) didefinisikan sebagai perputaran pada sumbu longitudinal, yaitu sumbu yang melalui bagian belakang ke bagian depan quadrotor. Perputaran sumbu ini mengakibatkan quadrotor melenggok kekiri atau kekanan. Hal ini diperoleh dengan meningkatkan dorongan (*thrust*) pada salah satu motor yang terpasang pada sumbu Y dan mempertahankan atau mengurangi dorongan pada



Gambar 1: Sistem koordinat yang digunakan.

motor yang satunya sedangkan dua motor pada sumbu X kecepatan putarannya tetap dipertahankan.

- *Pitch* (θ) didefinisikan sebagai perputaran pada sumbu lateral, yaitu sumbu yang melalui bagian kiri ke bagian kanan quadrotor. Perputaran sumbu ini menyebabkan quadrotor mengangguk keatas atau kebawah. Hal ini diperoleh dengan meningkatkan dorongan (*thrust*) pada salah satu motor yang terpasang pada sumbu X dan mempertahankan atau mengurangi dorongan pada motor yang satunya sedangkan dua motor pada sumbu Y kecepatan putarannya tetap dipertahankan.
- *Yaw* (ψ) didefinisikan sebagai perputaran pada sumbu vertikal yang tegak lurus terhadap lengan-lengan quadrotor pada titik pusat gravitasi quadrotor. Perputaran pada sumbu ini menyebabkan quadrotor bergerak kesisi kanan atau kesisi kiri. Yaw diperoleh dengan meningkatkan putaran motor-motor pada sumbu X dibandingkan dengan motor-motor pada sumbu Y ataupun sebaliknya. Hal ini mengakibatkan quadrotor berputar pada sumbu Z

B. Sistem Kontrol Stabilitas Quadrotor

Quadrotor membutuhkan suatu sistem kontrol stabilitas yang memadai untuk dapat terbang. Sensor *Inertial Measurement Unit* (accelerometer dan gyroscope) merupakan input yang dibutuhkan untuk memberitahu sudut putaran rangka quadrotor, namun kedua sensor ini memiliki noise. Performa dari kendali sikap (attitude) bergantung pada noise dan level guncangan dari sistem. Gyroscope akan menyimpang dalam waktu lama yaitu ketika berada dalam keadaan diam. Accelerometer tidak menyimpang dalam waktu lama namun kurang stabil. Untuk mengatasi ini digunakan algoritma Kalman Filter sehingga pembacaan nilai sudut lebih akurat. Kemudian hasilnya diinputkan ke PID kontroler.

- **Kalman Filter**
Kalman filter dibutuhkan untuk menggabungkan sensor gyroscope dan accelerometer untuk mendapatkan nilai sudut yang lebih akurat. Kalman filter dikenal juga sebagai *Linear Quadratic Estimation* (LQE) adalah suatu set persamaan matematika yang menyediakan kom-

putasi rekursif yang efisien untuk mengestimasi suatu proses keadaan. Algoritma ini menggunakan seri pengukuran yang teramati dalam satuan waktu, terkandung derau (*noise*), dan berbagai ketidakakuratan, dan menghasilkan estimasi yang cenderung lebih akurat dari hanya berdasarkan pengukuran tunggal.

Kalman filter bekerja dalam dua tingkat proses: pada tingkat prediksi, Kalman Filter menghasilkan estimasi pada variabel keadaan saat itu, bersama dengan ketidakpastiannya. Saat keluaran dari pengukuran berikutnya didapat, maka estimasi ini dibaharui menggunakan bobot rata-rata (*weighted average*), dengan bobot lebih diberikan kepada estimasi dengan kepastiannya lebih tinggi. Karena sifat rekursif dari algoritma ini, maka algoritma ini dapat berjalan pada waktu nyata menggunakan hanya masukan pengukuran saat itu dan hasil kalkulasi keadaan sebelumnya saja; tidak ada tambahan informasi terdahulu yang dibutuhkan.

Prediksi (*Predict*)

$$\hat{x}_{t|t-1} = F_t \hat{x}_{t-1|t-1} + B_t u_t \quad (1)$$

$$P_{t|t-1} = F_t P_{t-1|t-1} F_t^T + Q_t \quad (2)$$

Baharui (*Update*)

$$K_t = P_{t|t-1} H_t^T (H_t P_{t|t-1} H_t^T + R_t)^{-1} \quad (3)$$

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t (z_t - H_t \hat{x}_{t|t-1}) \quad (4)$$

$$P_{t|t} = (I - K_t H_t) P_{t|t-1} \quad (5)$$

Dimana,

\hat{x} : *Estimated state.*

F : *State transition matrix.*

u : *Control variables.*

B : *Control matrix.*

P : *State variance matrix.*

Q : *Process variance matrix.*

z : *Measurement variables.*

H : *Measurement matrix.*

K : *Kalman gain.*

R : *Measurement variance matrix.*

Kalman Filter mengasumsi model sistem dinamik linear. *Kalman Gain* adalah suatu fungsi relatif kepastian dari pengukuran dan estimasi keadaan saat itu, dan dapat disetel untuk mendapatkan kinerja tertentu. Untuk nilai *gain* tinggi, algoritma ini menempatkan bobot yang lebih pada pengukuran, sehingga mengikuti nilai tersebut dengan lebih teliti. Untuk nilai *gain* rendah, algoritma ini mengikuti prediksi model lebih teliti. Pada keadaan ekstrim, *gain* dengan nilai satu menyebabkan algoritma mengabaikan estimasi keadaan sepenuhnya, sedangkan nilai *gain* nol menyebabkan nilai pengukuran yang diabaikan.

- **PID Kontroler**

PID kontroler dibutuhkan untuk menentukan presisi derajat kestabilan quadrotor yang diinginkan berdasarkan pengaturan kecepatan motor. Masukan dari PID kontroler

adalah keluaran dari Kalman filter dan keluaran dari PID diskala kedalam nilai kecepatan motor. Jika parameter PID kontroler (proporsional, integral, dan derivatif) dipilih dengan keliru maka sistem akan menjadi tidak stabil seperti terjadi osilasi yang berlebihan atau respon waktu lama untuk itu diperlukan tuning yang baik.

$$u(t) = MV(t) = k_p e(t) + k_i \int_0^t e(\tau) + k_d \frac{d}{dt} e(t) \quad (6)$$

dimana:

k_e : Proporsional gain

k_i : Integral gain

k_d : Derivatif gain

e : Error = SP - PV

t : Waktu

Proporsional kontroler k_p akan memberi efek dengan mengurangi *rise time* tetapi tidak akan pernah menghilangkannya sehingga *steady-state error* tetap ada. Integral kontroler k_i akan berefek pada penghilangan *steady state error* tetapi mungkin akan membuat respon peralihan memburuk. Derivatif kontroler k_d akan memberikan efek peningkatan stabilitas pada sistem, mengurangi *overshoot*, dan memperbaiki respon peralihan.

C. Pengenalan Wajah Menggunakan Eigenface

PCA merupakan prosedur matematika untuk mengidentifikasi pola dalam data dan mengekspresi data sesuai persamaan dan perbedaan. PCA mereduksi dimensi dari kumpulan data dengan mencari kumpulan data baru yang memiliki variabel yang lebih sedikit tetapi memiliki hampir keseluruhan informasi yang dibutuhkan. Pada pokok bahasan matriks algebra akan dijelaskan tentang properti penting dari matriks yang merupakan bagian fundamental dari PCA, yaitu nilai eigen dan vector eigen.

PCA mentransformasikan variabel-variabel yang berkorelasi menjadi variabel-variabel non-korelasi yang dinamakan *principal components*. Pada pengenalan wajah *principal components* disebut juga *eigenfaces*. Principal component analysis diterapkan pada matriks simetrik persegi (matrik kovarian atau matrik korelasi dari suatu data) dan bisa juga langsung pada data matrix menggunakan *singular value decomposition*. Eigenface adalah salah satu algoritma pengenalan pola wajah yang berdasarkan principal component analysis. Untuk menghasilkan eigenface, sekumpulan citra digital dari wajah manusia diambil pada resolusi yang sama (misal $m \times n$) kemudian dinormalisasikan. Selanjutnya citra tadi diperlakukan sebagai vektor dimensi $m \times n$ dimana komponennya diambil dari nilai piksel citra. Berikut ini adalah langkah-langkah dalam menghasilkan eigenfaces.

Langkah pertama adalah mengumpulkan citra wajah. Satu citra wajah terdiri dari $m \times n$ pixel dimana pada dasarnya merupakan n baris dan m kolom dari suatu matriks, kemudian distandardkan ukurannya menjadi 90×120 pixels. Agar dapat membandingkan citra, maka citra dengan dimensi 90×120 pixels tersebut harus dikonversi menjadi satu kolom vector

10800×1 .

$$\mathbf{I} = \begin{bmatrix} i_{1,1} & i_{1,2} & \cdots & i_{1,120} \\ i_{2,1} & i_{2,2} & \cdots & i_{1,160} \\ \vdots & \vdots & \ddots & \vdots \\ i_{90,1} & i_{90,2} & \cdots & i_{90,120} \end{bmatrix} \rightarrow \Gamma = \begin{bmatrix} \gamma_{1,1} \\ \vdots \\ \gamma_{90,1} \\ \gamma_{1,2} \\ \vdots \\ \gamma_{90,2} \\ \gamma_{1,120} \\ \vdots \\ \gamma_{90,120} \end{bmatrix}$$

Berikutnya setiap citra vektor ditempatkan saling berdampingan dalam satu set matriks S.

$$[\Gamma_1 \quad \Gamma_2 \quad \cdots \quad \Gamma_n] \rightarrow S$$

Selanjutnya menghitung rata-rata wajah (*Average Face*) dengan cara menjumlahkan semua citra wajah $\Gamma_1 + \Gamma_2 + \cdots + \Gamma_n$ dan dibagi total citra.

$$\Psi = \frac{1}{n} \sum_{i=1}^n \Gamma_i \quad (7)$$

Setelah rata-rata wajah diperoleh maka citra wajah Γ_i dikurangi rata-rata wajah Ψ hasilnya ditempatkan pada matriks A.

$$\Phi = \Gamma_i - \Psi \quad (8)$$

$$[\Phi_1 \quad \Phi_2 \quad \cdots \quad \Phi_n] \rightarrow A$$

$$\begin{aligned} C &= \frac{1}{n} \sum_{j=1}^n \Phi_j \Phi_j^T \\ &= \frac{1}{n} AA^T \end{aligned} \quad (9)$$

Langkah berikut mencari eigenvalues dan eigenvectors.

$$\begin{aligned} A &= U \Sigma V^T \\ AA &= U \Sigma V^T U \Sigma V^T \\ AA^T &= U \Sigma V^T V \Sigma^T U^T \\ AA^T &= U \Sigma^2 U^T \end{aligned} \quad (10)$$

$$\begin{aligned} A &= U \Sigma V^T \\ AA &= U \Sigma V^T U \Sigma V^T \\ A^T A &= V \Sigma^T U^T U \Sigma V^T \\ A^T A &= V \Sigma V^T \end{aligned} \quad (11)$$

$$A v_i = \sigma_i u_i \quad (12)$$

$$\omega_i = u_i^T \Phi \quad (13)$$

$$\Omega_i = \begin{bmatrix} \omega_1^i \\ \omega_2^i \\ \vdots \\ \omega_j^i \end{bmatrix}, \quad i = 1, 2, \dots, m \quad (14)$$

Sekarang setelah *face space* Ω dibuat maka citra dapat diinput untuk dikenali. Berikut ini adalah prosedurnya.

1) Transformasi

$$\Phi = \Gamma - \Psi \tag{15}$$

2) Bandingkan

$$\omega_n = u_i^T \Phi \tag{16}$$

3) Transformasikan bobot kedalam vektor

$$\Omega = [\omega_1 \ \omega_2 \ \dots \ \omega_n] \tag{17}$$

4) Menentukan jarak minimum Euclidian

$$e_r = ||\Omega - \Omega_n|| \tag{18}$$

III. PERANCANGAN DAN IMPLEMENTASI SISTEM

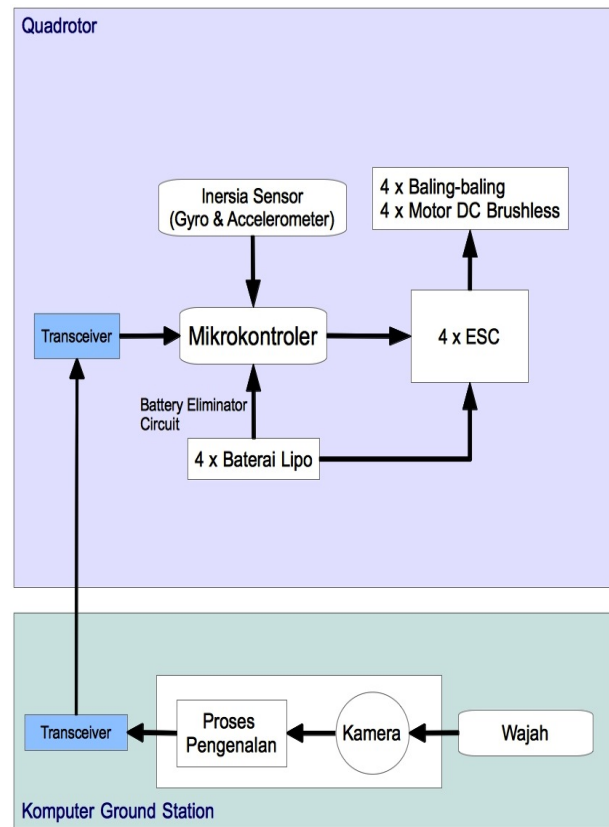
Perancangan dan Implementasi Sistem terbagi atas dua sub kategori yaitu:

1. Perancangan dan Implementasi Perangkat Keras
2. Perancangan dan Implementasi Perangkat Lunak

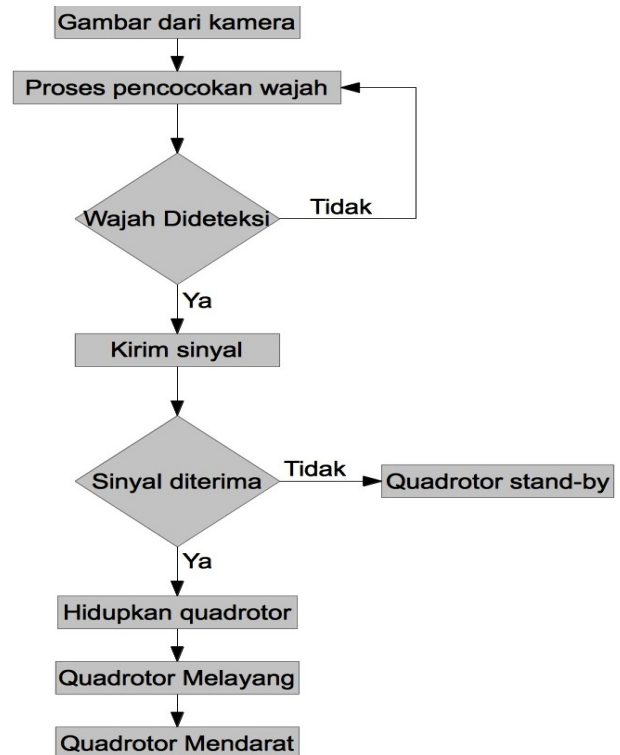
Perancangan dan konstruksi quadrotor dimulai dengan membangun rangka quadrotor. Bahan yang dipilih harus ringan dan kokoh. Konstruksi rangka quadrotor dibuat dari pipa-pipa kotak aluminium yang berukuran 1x1cm dan disambung menggunakan baut. Blok diagram quadrotor diperlihatkan pada gambar 2. Komponen utama yang dipakai untuk membangun quadrotor adalah motor, esc, baling-baling, baterai, rangka aluminium, kontroler, sensor inersia, dan alat komunikasi. Untuk menerbangkan quadrotor dibutuhkan suatu bentuk rangkaian kontrol untuk mengatur stabilitas. Sensor inersia diperlukan untuk menjaga keseimbangan quadrotor. Sensor inersia yang digunakan adalah digital gyroscope dan digital accelerometer. Untuk memanfaatkan kedua sensor ini maka harus dilakukan penyatuan nilai menggunakan algoritma kalman filter. Penyatuan nilai ini diperlukan karena gyroscope akan menyimpang dalam waktu lama tetapi stabil dalam waktu singkat, sedangkan accelerometer stabil dalam waktu lama tetapi sedikit menyimpang dalam waktu singkat. Sensor inersia ini dapat memberikan nilai keluaran dari rotasi roll dan pitch. Rotasi *yaw* tidak bisa karena accelerometer membaca gravitasi bumi.

Komputer *ground station* akan mengirimkan sinyal melalui alat pemancar jika ada wajah pemilik yang dikenali. Selanjutnya mikrokontroler memproses sinyal masukan yang ditangkap oleh penerima. Quadrotor akan diterbangkan dengan tetap menjaga keseimbangan berdasarkan *setpoint* pada PID kontroler sampai adanya perintah pendaratan oleh komputer *ground station*.

Quadrotor diprogram berdasarkan masukan dari sinyal penerima yang akan memberi tanda supaya quadrotor terbang. Masukan dari gyroscope dan accelerometer dipadukan menggunakan algoritma Kalman Filter untuk memberi nilai sudut roll dan pitch yang lebih baik. Selanjutnya hasil tersebut dimasukkan ke PID Kontroler untuk menstabilkan sikap (*attitude*).



Gambar 2: Blok Diagram Sistem Lepas Landas pada Quadrotor UAV Menggunakan Pengenalan Wajah.



Gambar 3: Diagram Alir

Pada gambar 3 diperlihatkan diagram alir sistem. Dimulai dengan pelatihan komputer untuk mengenali citra wajah. Hasil eigenface kemudian disimpan dalam database. Saat dimulai pengenalan proses dilakukan sama seperti pelatihan tetapi kemudian hasil eigen dari proses pelatihan dibandingkan dengan citra yang masuk berdasarkan jarak euclidean. Jika verifikasi citra tersebut valid maka transmitter yang disambungkan pada laptop akan mengirimkan sinyal ke receiver quadrotor. Kode sinyal tersebut menandakan supaya quadrotor terbang. Mikrokontroler akan menghitung sudut pitch dan roll berdasarkan keluaran accelerometer dan gyroscope. Hasil tersebut digunakan sebagai masukan pada PID kontroler untuk menstabilkan orientasi. Quadrotor diprogram untuk mendarat sendiri berdasarkan lamanya waktu penerbangan.

IV. PENGUJIAN DAN ANALISA SISTEM

Pengujian sumber tegangan dilakukan pada tegangan yang disuplai baterai dan BEC (*Battery Elimination Eircuit*). Tegangan utama dari baterai akan disuplai ke motor, sedangkan BEC akan mensuplai tegangan mikrokontroler, sensor IMU, dan RC receiver. Dari yang diperoleh sumber tegangan yang tertera pada produk hampir sama dengan hasil pengujian. Tegangan yang tertera pada baterai adalah 11.1 volt dan pada BEC 5.25-6 volt (default 5.25).

Pada saat pengujian secara offline dilakukan maka data pitch dan roll diambil. Pengambilan data ini kemudian disesuaikan dengan nilai setpoint pada PID kontroler. Nilai sensor gyro dan accelerometer berbeda sebelum dan saat motor diaktifkan. Ini dikarenakan guncangan motor mempengaruhi nilai sensor-sensor tersebut. Saat percobaan sengaja dilakukan guncangan untuk melihat respon PID kontroler. Hal ini memberikan verifikasi jika sistem bekerja dengan baik dan selain itu memberikan pengertian cara kerja PID kontroler.

Untuk memastikan bahwa quadrotor cukup stabil untuk penerbangan pertama, dibutuhkan tuning pada PID kontroler. Beberapa metode untuk tuning antara lain metode Ziegler-Nichols, tuning berdasarkan software, dan manual tuning. Manual tuning dipilih karena mudah dan langsung bisa dikonfirmasi secara visual kebenaran implementasinya. Untuk PID tuning setiap axis dituning secara terpisah yaitu pada pitch dan roll.

Proses manual tuning PID dimulai dengan menaikkan proportional gain sampai sistem mengalami sedikit osilasi sebelum settling. Berikut integral gain dinaikan untuk membuang steady state error. Terakhir, jika diperlukan differential gain dinaikan untuk membuang sedikit overshoot dan osilasi. Jika saat penerbangan pertama quadrotor terjatuh maka ini dapat menyebabkan kerusakan kontroler on-board dan rangka quadrotor. Untuk itu semua langkah-langkah praktikal diambil untuk memastikan sistem dapat bekerja sesuai keinginan saat penerbangan. Untuk memastikan sistem cukup stabil untuk terbang, maka quadrotor digantung untuk mencegah kecelakaan sebelum akhirnya diujicobakan secara real-time.

Kedua wireless Xbee diset pada baudrate yang sama, dalam hal ini 115.200 Khz. Untuk mengirim karakter ASCII dari laptop maka fungsi `write(fd, "fly", 3);` akan dipanggil. Parameter pertama dari fungsi itu yaitu `fd` menyatakan nama port yang terhubung dengan Xbee transmitter, parameter kedua

yaitu "fly" adalah karakter-karakter atau string ASCII yang hendak dikirim, sedangkan parameter yang terakhir menandakan jumlah karakter tersebut.

Algoritma untuk menangkap kode yang dikirim dari laptop dilakukan dalam fungsi `void loop()` yang jika ada serial data `if(Serial.available())` maka data akan dibaca dari port serial `char getData = Serial.read();` dan jika data tersebut merupakan kode yang dikehendaki `if(getData = 'fly')` akan proses komputasi selanjutnya akan dilakukan. Dari hasil pengujian Xbee Pro dengan menggunakan baudrate 115.200 Hz diperoleh jarak maksimal 43 M.

Pengujian dilakukan pada setiap tahap perancangan untuk memastikan setiap modul beroperasi dengan benar dan untuk menyakinkan bahwa perangkat lunak tidak akan mengalami kemunduran jika ada perubahan-perubahan lebih lanjut dilakukan. Pengujian awal sensor dilakukan tanpa mengaktifkan motor yang terpasang baling-baling. Hal ini dilakukan karena baling-baling yang bergerak berpotensi menyebabkan cedera. Pada pengujian sistem komunikasi radio, perangkat lunak diprogram untuk mengirimkan kode ASCII.

Untuk pengujian perangkat lunak pengenalan wajah dilakukan pada tingkat pencahayaan yang berbeda. Pada saat pelatihan program wajah diposisikan pada sudut pandang yang berbeda untuk memaksimalkan proses pengenalan. Data yang diambil adalah berupa tingkat ketelitian antara wajah yang dilatih dan wajah yang tidak dilatih. Pada Gambar 4 dan 5 diperlihatkan hasil pengujian keseluruhan sistem secara realtime.



Gambar 4: Pengenalan Wajah



Gambar 5: Manuver pada Quadrotor

V. KESIMPULAN

- 1) Pada tugas akhir ini quadrotor mampu terbang berdasarkan kode ASCII yang dikirimkan komputer ground station jika komputer tersebut mengenali wajah pemilik. Sistem pengenalan wajah cenderung bergantung pada tingkat pencahayaan, yaitu jika intensitas cahaya saat training dan pengenalan hampir sama, maka tingkat ketelitian pengenalan wajah sangat tinggi.
- 2) Pada penstabilan quadrotor, kalman filter memberikan hasil sudut yang lebih baik dibandingkan nilai dari masing masing individual sensor accelerometer maupun gyroscope walau demikian PID kontroler cenderung tidak stabil pada waktu lama karena getaran pada rangka quadrotor memberi hasil kurang baik pada accelerometer. Quadrotor juga cenderung berputar pada yaw karena tidak adanya kendali hal ini dapat diatasi dengan menambahkan sensor magnetometer.

DAFTAR PUSTAKA

- [1] B. Gibbs, "*Advanced Kalman Filtering, Least-squares and Modelling*", John Wiley & Sons, Inc., Hoboken, New Jersey, 2011.
- [2] G. Bradski, "*Learning OpenCV*", O'Reilly Media, Sebastopol, 2008.
- [3] K. Baker, "*Singular Value Decomposition Tutorial*", 2005.
- [4] K. Nonami, "*Autonomous Flying Robots*", Springer, Tokyo, 2010.
- [5] L. Smith, "*A Tutorial on Principal Components Analysis*", 2002.