

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/324537037>

Perhitungan Kapasitas Baterai dan Arus Komponen pada Ar.Drone Quadcopter untuk Estimasi Waktu dan Jarak Terbang

Article · January 2018

CITATIONS

0

READS

1,019

3 authors, including:



Gembong Edhi Setyawan
Brawijaya University

29 PUBLICATIONS 11 CITATIONS

[SEE PROFILE](#)



Barlian Henryranu Prasetyo
Brawijaya University

22 PUBLICATIONS 8 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Quadcopter [View project](#)

Perhitungan Kapasitas Baterai dan Arus Komponen pada Ar.Drone *Quadcopter* untuk Estimasi Waktu dan Jarak Terbang

Yusril Dewantara¹, Gembong Edhi Setyawan², Barlian Henryranu Prasetyo³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹yusrildewantara2@gmail.com, ²gembong @ub.ac.id, ³barlian@ub.ac.id

Abstrak

AR.Drone quadcopter 2.0 adalah salah satu jenis pesawat tanpa awak yang berukuran kecil. *Quadcopter* sering digunakan oleh praktisi teknologi dibidang aerodinamika, digunakan oleh penghobi RC(remote control), digunakan untuk sarana pada bidang *search and rescue*, atau digunakan pada bidang militer. Namun tidak jarang terjadi hal yang tidak terduga seperti *quadcopter* yang tiba tiba jatuh. Salah satu penyebab dari hal tersebut adalah karena kurangnya fitur yang mendukung bagi pengguna untuk melakukan pemantauan kapasitas baterai untuk mengetahui sisa waktu terbang yang dapat dilakukan oleh *quadcopter* serta seberapa jauh *quadcopter* dapat terbang. Untuk mengatasi hal tersebut maka dibuatlah sistem yang dapat membuat *quadcopter* dapat beroperasi menggunakan *keyboard laptop* sebagai *controller*, dapat memantau energi baterai yang digunakan untuk dapat mengestimasi waktu dan mengestimasi perpindahan ketika *quadcopter* beroperasi. Estimasi waktu terbang *quadcopter* menggunakan parameter berupa kapasitas baterai, dan arus dari komponen-komponen pada *board quadcopter* yang dijumlahkan dengan arus pada motor *brushless* yang dikonversi dari PWM (*Pulse Width Modulation*). Dan untuk mendapatkan nilai yang menunjukkan sisa waktu terbang *quadcopter* digunakan persamaan hukum *coloumb*. Untuk estimasi perpindahan dilakukan berdasarkan percobaan pengujian terbang *quadcopter* yang melakukan operasi terbang dalam waktu satu menit. Hasil dari penelitian ini menginformasikan bahwa kesalahan ketepatan gerakan menggunakan *keyboard* adalah 0 %, kesalahan ketepatan estimasi waktu adalah 1.14 %, dan kesalahan ketepatan estimasi perpindahan adalah sebesar 29.75 %.

Kata kunci: kapasitas baterai, arus listrik, estimasi waktu, estimasi jarak, *quadcopter*

Abstract

AR.Drone quadcopter 2.0 is one type of unmanned aircraft that is small. *Quadcopter* is often used by technology practitioners in the field of aerodynamics, used by RC(remote control) hobbyists, used for facilities in the field of *search and rescue*, or used in military fields. But not infrequently unexpected things like *quadcopter* that suddenly fell. One of the causes of this is the lack of features that support the user to monitor battery capacity to find out the rest of the flight time the *quadcopter* can do and how far the *quadcopter* can fly. To solve that problem, a system that can make the *quadcopter* operate using a laptop keyboard as a controller can monitor the energy of the battery and used to estimate the time and estimate the displacement when the *quadcopter* operates have been built. Flight time estimation uses parameters of battery capacity, and current from components on *quadcopter* board that summed with current on a brushless motor which converted from PWM (*Pulse Width Modulation*). And to get the value that shows the remaining time of flying *quadcopter* is used *coloumb's law*. For estimation of displacement is done based on *quadcopter* flight test experiments that perform flying operations within a minute. The results of this study informed that the error of movement accuracy using the keyboard is 0%, the error of time fly estimation accuracy is 1.14%, and the error of displacement estimation accuracy is 29.75%

Keywords: battery capacities, time estimation, displacements estimation, *quadcopter*

1. PENDAHULUAN

Quadcopter merupakan pesawat tanpa awak yang mempunyai 4 motor sebagai

penggerak. Dan *quadcopter* mampu melakukan *take-off* dan *landing* secara vertikal. 2 motor pada *quadcopter* akan berputar searah jarum jam dan 2 lainnya berputar berlawanan arah jarum

jam. Hal tersebut mengakibatkan gaya tarik dan gaya dorong sehingga *quadcopter* dapat terbang.

Penelitian teknologi dibidang *quadcopter* sedang banyak dikembangkan. Pengembangan banyak menggunakan algoritma sistem cerdas, menggunakan *embedded system*, dan menggunakan pengolahan citra. Seperti penelitian yang dilakukan oleh pribadi (2017) tentang pengendalian *quadcopter* menggunakan *google cardboard*, penelitian oleh Jorge (2011) yang mengembangkan *quadcopter* agar dapat terbang mengikuti garis, dan penelitian yang dilakukan oleh Qetkeaw (2011) tentang penstabil kamera yang dipasang pada *quadcopter* dengan tujuan gambar yang diambil stabil.

Pengguna *quadcopter* sering memanfaatkan *quadcopter* untuk misi terbang (*mission flight*). Tetapi tidak jarang juga terjadi hal yang tidak terduga seperti *quadcopter* yang tiba-tiba jatuh, salah satu penyebab hal tersebut adalah karena kurangnya fitur yang membuat pengguna mengetahui sisa waktu terbang dan sejauh mana *quadcopter* dapat terbang. Maka dari itu dilakukan penelitian untuk menerapkan sebuah sistem yang dapat melakukan perhitungan kapasitas baterai dan arus pada komponen *quadcopter*.

Penelitian mengenai perhitungan kapasitas baterai dan arus komponen pernah dilakukan oleh Turo (2015). Penelitian tersebut menggunakan (BMS) *Battery Management System* dengan memanfaatkan sensor pintar pada baterai. Tujuan dari penelitian tersebut adalah untuk mengetahui operasi yang dapat dilakukan pada alat militer.

Penelitian lain juga pernah dilakukan oleh Williard, et al (2011) tentang perhitungan kapasitas baterai dan arus komponen UAV. Parameter yang digunakan dalam sistem ini adalah komponen-komponen dari UAV yang mengakibatkan proses pengosongan baterai. Penelitian dilakukan dengan mengembangkan *software* dan *hardware* pada sistem *monitroing* baterai dengan tujuan agar sistem dapat menyajikan data untuk performa UAV yang lebih baik.

Pada penelitian ini dibuat sistem yang mampu melakukan perhitungan kapasitas baterai dan arus komponen *quadcopter*. Tujuan dari sistem ini adalah untuk mengestimasi waktu dan perpindahan yang dapat dilakukan saat *quadcopter* terbang. Dan supaya performa *quadcopter* dapat menjadi lebih optimal.

2. ANALISIS HARDWARE PADA QUADCOPTER

Menurut Williard, et al (2011) *Quadcopter* adalah *portable microelectronic system*. Pada *quadcopter* terdapat banyak sekali perangkat elektronik yang berukuran sangat kecil, baik motor, komponen sensor dan transduser, serta kamera. Perangkat-perangkat kecil tersebut diintegrasikan satu dengan yang lain sehingga menjadi sebuah sistem sehingga *quadcopter* dapat melakukan operasi terbang. Informasi mengenai komponen-komponen ditunjukkan pada tabel 1, dan letak komponen ditunjukkan oleh gambar 1.



Gambar 1. Letak komponen pada AR.Drone

Tabel 1. Komponen pada AR.Drone

Arus (mA)	Fungsi
182	Wifi Module
1200	Compass
8	Accelerometer
46	Pressure Sensor
0.005	Ultrasonic Sensor
65	Microcontrollers
250	DDR SDRAM
200	Flash Memory - 128 MB
2	3-axis Gyro Motion Sensor
350	Power Management + USB
182	Front Camera
46	Bottom Camera

Table 2. Spesifikasi Motor

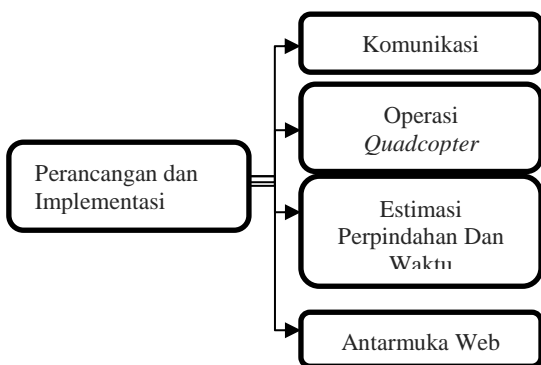
Spesifications	Value
Jenis Motor	Inrunner Motor
Kecepatan	5000 RPM
Daya Motor	14.5 Watss

Pada gambar 1 ditunjukkan letak dari komponen-komponen AR.Drone 2.0. Dimana terdapat *module* kamera, terdapat *board* yang terintegrasi dengan sensor sensor, serta motor

brushless. Dan pada tabel 1 dan 2 ditunjukkan informasi mengenai komponen elektronik berupa pemroses aritmetik, sensor dan transduser, module kamera, dan motor *brushless* yang terletak didalam *board AR.Drone 2*.

3. PERANCANGAN DAN IMPLEMENTASI

Tahap perancangan dan implementasi pada penelitian ini adalah komunikasi yang akan membahas mengenai komunikasi antara *quadcopter* dan laptop, gerakan *quadcopter* yang akan membahas bagaimana *quadcopter* dapat berpindah, kemudian tahap untuk mengestimasi jarak dan waktu agar operasi terbang dapat dilakukan maksimal. Dan yang terakhir adalah antarmuka *web*. Alur perancangan dan implementasi ditunjukkan pada gambar 2.



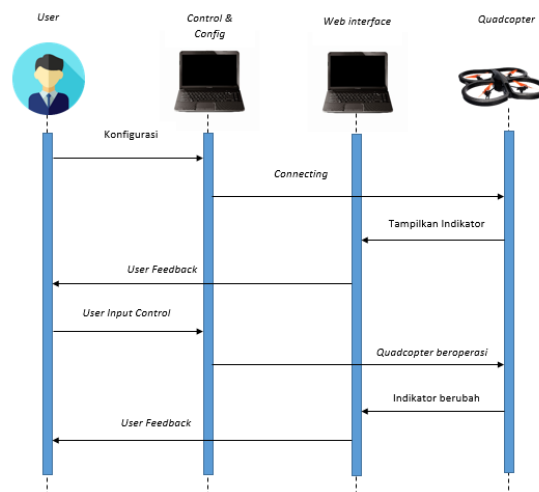
Gambar 2. Diagram Perancangan dan Implementasi

Gambar 2 merupakan diagram perancangan dan implementasi pada sistem. Terdapat sub bab komunikasi yang membahas mengenai bagaimana *quadcopter* dan komputer dapat saling bertukar data. Sub bab operasi *quadcopter* membahas mengenai bagaimana *quadcopter* dapat terbang dan beroperasi sesuai dengan masukkan pengguna melalui *keyboard laptop*. Sub bab estimasi perpindahan dan estimasi waktu membahas mengenai bagaimana sistem dapat melakukan estimasi waktu dan jarak. Dan pada sub bab antarmuka *web* membahas mengenai bagaimana aplikasi *web* mendapat data berupa navigasi data, dan kinerja baterai dari *quadcopter*.

3.1 Komunikasi

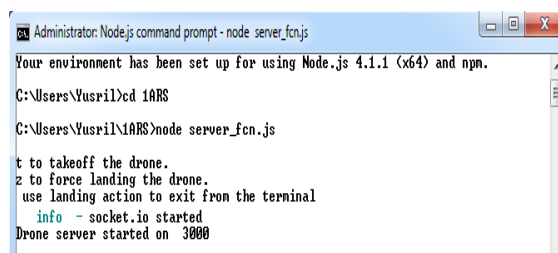
Parrot AR.Drone 2.0 melakukan koneksi dengan perangkat seperti *smartphone* atau *laptop* menggunakan sambungan *wifi*. *Parrot AR.Drone* telah menyiapkan alamat IP server 192.168.1.1 dan *client* 192.168.1.2 yang dapat

digunakan pada *laptop*, *tablet pc*, dan *smartphone*.



Gambar 3. Urutan sambungan dan pengendalian

Gambar 3 menjelaskan mengenai alur komunikasi antara *quadcopter* dan laptop. Pertama, pengguna yang ditunjukkan oleh nomor 1 akan melakukan konfigurasi koneksi antara laptop dan *quadcopter* yang masing-masing ditunjukkan oleh nomor 2 dan nomor 3. Selanjutnya user akan memanggil program utama melalui *node js console terminal* yang ditunjukkan oleh gambar nomor 4. Selanjutnya pengguna dapat memanggil *interface web* dengan menggunakan *port* yang sudah diset



Gambar 4. Implementasi Komunikasi.

Gambar 4 menjelaskan mengenai *feedback* yang didapat dari *console terminal node js*. *Feedback* yang didapat menunjukkan bahwa komunikasi antara *quadcopter* dan komputer telah berhasil, operasi terbang dapat dilakukan serta menampilkan aplikasi *web* dengan memanggil *port* yang tertera pada *console terminal*

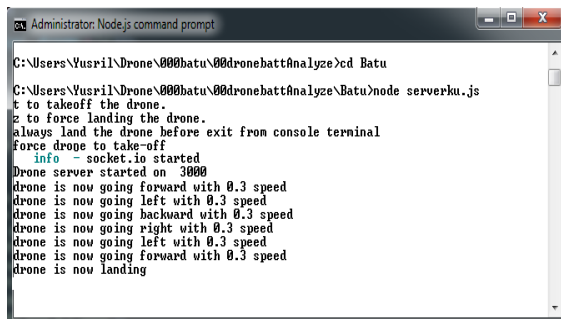
3.2 Operasi Quadcopter

Tahap yang penting dari sistem adalah bahwa *quadcopter* haruslah bisa melakukan operasi terbang. Tabel 3 menunjukkan operasi terbang yang bisa dilakukan dalam sistem.

Tabel 3. Gerakan *Quadcopter*

Key	Fungsi
C	Untuk menghentikan sistem
T	Untuk <i>take-off</i>
Z	Untuk <i>landing</i>
W	Untuk terbang maju
A	Untuk terbang ke-arah kiri
S	Untuk mundur
D	Untuk terbang ke arah kanan
N	Untuk gerakan <i>counterclockwise</i>
B	Untuk gerakan <i>clockwise</i>
M	Untuk menghentikan <i>quadcopter</i> / <i>hover</i>
O	Untuk mengurangi kecepatan
P	Untuk menambah kecepatan
U	Untuk gerakan naik
J	Untuk gerakan turun

Operasi dilakukan dengan menggunakan *keyboard laptop*. *Keyboard* dipilih karena penggunaannya yang sama dengan pengendalian yang sering digunakan pada *game* serta tidak membutuhkan perangkat tambahan. Pengendalian *keyboard* pada *quadcopter* juga mendukung fungsi *interrupt* sehingga apabila terjadi kesalahan terbang dapat segera disesuaikan kembali.

**Gambar 5.** Console Terminal Operasi *Quadcopter*

Gambar 5 menunjukkan status terbang *quadcopter*. Setelah koneksi antara *quadcopter* dan komputer berhasil, dan program utama di panggil maka console terminal akan memberikan *feedback* berupa operasi terbang *quadcopter*.

3.3 Estimasi Jarak Dan Waktu

Estimasi waktu dilakukan dengan cara menghitung arus yang dipergunakan motor *brushless*, kamera serta sensor dan transduser. Arus untuk kamera serta sensor dan transduser dihitung menggunakan persamaan 1 dan 2, sedangkan untuk menghitung arus motor digunakan persamaan 3, 4, 5, dan 6. Jika semua

komponen diketahui arusnya kemudian akan dijumlah arus totalnya. Dan setelah didapat nilai arus total kemudian disubstitusikan kedalam persamaan 6.

$$I_{total} = I_1 + I_2 + I_3 \dots + I_n \quad (1)$$

$$\frac{1}{I_{total}} = \frac{1}{I_1} + \frac{1}{I_2} + \frac{1}{I_3} + \dots + \frac{1}{I_n} \quad (2)$$

$$Duty\ Cycle = \frac{T_{on}}{T_{on} + T_{off}} \quad (3)$$

$$V_{out} = Duty\ cycle \times V_{ref} \quad (4)$$

$$P = V \times I \quad (5)$$

$$I = V/R \quad (6)$$

$$t = Q/I \quad (7)$$

Persamaan satu digunakan untuk mencari arus total pada rangkaian parallel, sedangkan persamaan 2 digunakan untuk mencari arus total pada rangkaian seri. Persamaan 1 dan 2 berkebalikan dengan persamaan untuk mencari hambatan pada rangkaian seri dan parallel, hal ini disebabkan karena pada rangkaian parallel nilai tegangan akan sama dan nilai arus akan berbeda, begitu juga pada rangkaian parallel, nilai arus akan sama dan nilai tegangan akan berbeda. Pada persamaan 1 dan 2 variabel I merupakan arus dari akronim *Intensity* yang dinyatakan dalam satuan *mA* (*mili Ampere*).

Persamaan 3 dan 4 digunakan untuk mencari nilai tegangan yang diubah dari nilai *pwm*. Pada persamaan 3 terdapat *Duty Cycle* yang merupakan siklus sinyal, sedangkan T_{on} adalah waktu saat sinyal bernilai 1, dan T_{off} adalah waktu saat sinyal bernilai 0. Pada persamaan 4 terdapat V_{out} yang merupakan tegangan keluaran yang dinyatakan dengan satuan volt, *Duty cycle* yang merupakan siklus dari sinyal, dan V_{ref} yang merupakan tegangan referensi dengan satuan volt.

Persamaan ke 5 digunakan untuk mencari arus referensi dengan P adalah daya yang merupakan akronim dari *power*, V adalah tegangan keluaran dengan satuan volt, dan I merupakan arus referensi dengan satuan *mA* (*milliAmps*). Pada persamaan 6 digunakan untuk mencari hambatan referensi, dengan V adalah voltase input dari baterai dengan satuan volt, I adalah arus referensi dengan satuan *mA*, dan R adalah hambatan dengan satuan ohm. Setelah didapat nilai V_{out} , dan hambatan referensi. Maka didapat nilai arus output dari masing masing motor.

Persamaan 7 digunakan untuk mencari nilai waktu. Variabel t merupakan waktu dari akronim

time dalam yang dinyatakan dalam satuan menit, variabel Q menunjukkan jumlah muatan yang merupakan akronim dari *Quantity* yang dinyatakan dalam satuan *mAh*(milli Amper hour), dan variabel I adalah arus dari akronim *Intensity* yang dinyatakan dalam satuan *mA*(milli Ampere).

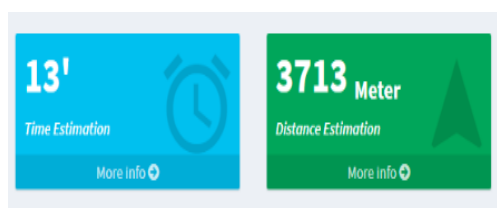
Estimasi perpindahan didapat dengan cara menghitung rata-rata dari jarak yang ditempuh dalam satu menit dalam percobaan terbang pada operasi maju, mundur, kekiri, dan kekanan, sebanyak 10 kali. Setelah nilai rata-rata jarak per menit didapat maka dapat disubstitusikan kedalam persamaan 5. Tabel 4 menunjukkan rata-rata percobaan dari ke 10 percobaan dengan gerakan maju, mundur, ke kiri, dan ke kanan dengan input kecepatan 0.3 hingga 0.7.

Tabel 4. Percobaan

Gerakan	Input Kecepatan					Rata-rata
	0.3	0.4	0.5	0.6	0.7	
Maju	21	23	32	50	47	34.6
Mundur	28	24	33	42	53	36
Kiri	26	24	29	42	51	34.4
Kanan	27	24	27	55	49	36.4
Rata-rata perpindahan per menit						35.35

$$S = t \times v \quad (8)$$

Persamaan 8 merupakan rumus kecepatan linear dengan variabel S menunjukkan jarak dinyatakan dalam satuan meter, variabel t adalah waktu yang merupakan akronim dari *time* dinyatakan dalam satuan menit, sedangkan variabel v adalah kecepatan (jarak/ waktu) yang merupakan akronim dari *velocity*.

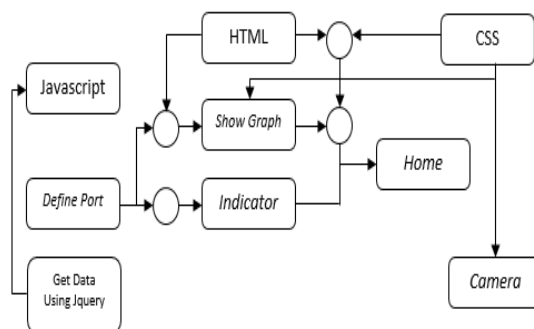


Gambar 6. Implementasi Estimasi Perpindahan dan Waktu

Gambar 6 menunjukkan hasil implementasi dari estimasi waktu dan estimasi jarak. Estimasi waktu ditunjukkan dengan satuan menit, dan pada estimasi perpindahan ditunjukkan dengan satuan meter. Estimasi waktu dan perpindahan ditempatkan paling atas pada antarmuka web, sehingga pengguna akan langsung berfokus pada estimasi waktu dan estimasi perpindahan.

3.4 Antarmuka Web

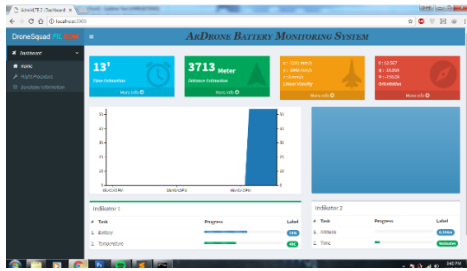
Antarmuka web dibuat untuk media *monitoring*. Dalam antarmuka yang dibuat, memuat informasi mengenai kinerja baterai yang disajikan dalam bentuk *real-time graphic*, estimasi jarak dan waktu, data navigasi *quadcopter* yang menunjukkan kecepatan linear pada sumbu x, y, dan z, serta orientasi perubahan posisi pada *pitch*, *roll*, dan *yaw* yang disajikan kedalam bentuk angka. Selain itu terdapat indikator berupa kapasitas baterai, suhu, ketinggian, serta waktu aktif *quadcopter* yang disajikan kedalam bentuk *progress bar*.



Gambar 7. Implementasi Antarmuka Web

Gambar 7 menunjukkan aliran antarmuka yang saling berhubungan antara elemen HTML, CSS, *Jquery*, *Javascript*. HTML digunakan untuk membentuk layout web, CSS digunakan untuk mengatur tampilan warna, *Jquery* digunakan untuk men-*subscribe* data yang dikirim dari server, dan *javascript* digunakan agar server dan tampilan web dapat berkomunikasi.

Dalam penggunaan antarmuka web, data dari *quadcopter* dikirimkan menggunakan *websocket* sehingga dapat ditampilkan kedalam antarmuka web. Karena tanpa *websocket* data dari *quadcopter* tidak akan bisa ditampilkan pada web. Hal itu disebabkan karena fungsi *websocket* yang memungkinkan melakukan pertukaran data tanpa harus mengirimkan *request* terlebih dahulu, sehingga data dapat dikirimkan secara terus menerus. Gambar 7 menunjukkan tampilan antarmuka web yang terintegrasi dengan *quadcopter*.



Gambar 8. Implementasi Antarmuka Web

Gambar 8 menunjukkan implementasi dari antarmuka *web*. Kotak warna hitam yang terletak paling kiri menunjukkan informasi. Kotak dengan warna biru muda yang terletak diatas menunjukkan estimasi waktu, kotak warna hijau menunjukkan estimasi jarak, kotak berwarna jingga menunjukkan kecepatan *linear* x, y, dan z, sedangkan kotak warna merah menunjukkan orientasi pada sumbu *roll*, *pitch* dan *yaw*. Di tengah terdapat grafik yang menunjukkan kinerja baterai, dan disebelah grafik digunakan untuk menampilkan kamera. Dibagian paling bawah terdapat indicator, berupa kapasitas baterai, temperatur mesin, ketinggian serta waktu aktif *quadcopter*.

4. PENGUJIAN DAN HASIL

Pada pengujian dan analisis membahas mengenai pengujian dari apa yang telah diimplementasikan, sedangkan pada bagian analisis akan dibahas mengenai analisis tentang sistem telah diuji. Pengujian dan analisis pertama adalah pengujian ketepatan gerakan *quadcopter*, pengujian dan analisis kedua adalah ketepatan estimasi waktu, dan yang ketiga adalah pengujian dan analisa ketepatan estimasi perpindahan.

4.1 Hasil Ketepatan Gerakan *Quadcopter*

Tabel 5. Hasil Ketepatan Gerakan Quadcopter

Gerakan	Percobaan ke-							
	1	2	3	4	5	6	7	8
Maju	✓	✓	✓	✓	✓	✓	✓	✓
Mundur	✓	✓	✓	✓	✓	✓	✓	✓
Kiri	✓	✓	✓	✓	✓	✓	✓	✓
Kanan	✓	✓	✓	✓	✓	✓	✓	✓

Pada tabel 5 ditunjukkan bahwa perintah yang diberikan pada *quadcopter* dari *keyboard* laptop dilaksanakan tanpa kesalahan. Dari keempat gerakan yaitu maju, mundur, ke kiri dan ke kanan yang diuji sebanyak 4 kali didapatkan bahwa keberhasilan ketepatan gerakan yang

diperintahkan pada *quadcopter* dari *keyboard* laptop adalah 100%.

4.2 Hasil Ketepatan Estimasi Waktu

Tabel 6. Ketepatan Estimasi Waktu

Gerakan	Input Kecepatan (%)					Rata-rata
	0.3	0.4	0.5	0.6	0.7	
Hover	1	1	0.7	0.8	1.2	0.94
Maju	1.1	1.2	1.1	1.5	1.1	1.2
Mundur	1.5	1.7	1	1.2	1.5	1.38
Kiri	1.1	1.3	0.7	1.1	1.5	1.14
Kanan	1.2	1	0.9	1.3	1	1.08
Total Error						1.148

Tabel 6 menunjukkan hasil dari kesalahan ketepatan estimasi waktu. Ditunjukkan rata-rata kesalahan dari pengujian sebanyak 10 kali pada input kecepatan 0.3 hingga 0.7, dan dengan gerakan *hover*, maju, mundur, ke kiri, dan ke kanan. Selanjutnya hasil rata rata dari 10 kali pengujian di rata-rata kembali sehingga didapat total error sebesar 1.148 %. Total error di dapat dari perhitungan menggunakan persamaan 9.

$$\overline{\text{total error}} = \frac{\sum_{i=1}^5 a_i}{5} \quad (9)$$

Persamaan 9 merupakan persamaan yang digunakan untuk menghitung total error pada ketepatan estimasi waktu. Dengan batas bawah adalah 1, batas atas adalah 5, dan a_i adalah suku yang merepresentasikan rata-rata *error* dari pengujian pada gerakan yang disebutkan.

4.3 Hasil Pengujian Jarak Tempuh Per Satuan Waktu

Tabel 7. Jarak tempuh per satuan waktu

Gerakan	Input Kecepatan (%)					Rata-rata
	0.3	0.4	0.5	0.6	0.7	
Maju	37.7	32.1	0.9	44.2	38.6	30.7
Mundur	15.1	29.2	9.4	21.6	44.2	23.9
Kiri	26.4	34.9	26.4	58.4	44.2	38.0
Kanan	29.2	29.2	20.7	13.1	38.6	26.2
Total Error						29.7

Tabel 7 menunjukkan presentase kesalahan pada pengujian ketepatan estimasi perpindahan *quadcopter*. Nilai kesalahan tersebut didapat dengan melihat selisih antara hasil pengujian dengan nilai akhir dari percobaan. Tabel 6 menyajikan data berupa error dari gerakan maju, mundur, ke kanan, dan ke kiri dengan input kecepatan 0.3 hingga 0.7. Rata-rata *error* dari

setiap gerakan kemudian dirata-rata kembali dengan menggunakan persamaan 10. Sehingga didapat total error sebesar 29.7 %

$$\overline{\text{total error}} = \frac{\sum_{i=1}^5 a_i}{5} \quad (10)$$

Persamaan 10 merupakan persamaan yang digunakan untuk menghitung total error pada ketepatan estimasi waktu. Dengan batas bawah adalah 1, batas atas adalah 5, dan a_i adalah suku yang merepresentasikan rata-rata *error* dari pengujian pada gerakan yang disebutkan.

5. KESIMPULAN

Dari hasil analisis data yang telah dilakukan pada tahap pengujian dan analisis, maka didapat kesimpulan dari apa yang telah dilakukan. Sistem yang dibuat dengan tujuan agar kinerja *quadcopter* menjadi lebih optimal. Tampilan *web* dibuat dengan menggunakan bahasa pemrograman *javascript* dan dengan menggunakan antarmuka menunjukkan hasil yang cukup baik dengan *error* yang relatif kecil. Dimana pengendalian dengan menggunakan keyboard laptop menunjukkan hasil yang sangat baik karena kesalahan pengendalian adalah 0%. Pada ketepatan estimasi waktu pada saat *quadcopter* melakukan operasi terbang menunjukkan tingkat kesalahan yang sangat kecil yaitu 1.148 %. Dan pada pengujian dan analisis mengenai ketepatan estimasi perpindahan yang dilakukan pada saat *quadcopter* terbang menunjukkan tingkat kesalahan sebesar +/- 29 %

DAFTAR PUSTAKA

- ALbert, M. & David, J., 2006. *Electronic Principles, (7th Edition)*. Standford: McGraw-Hill Education.
- Hadi, S W., Setyawan, G E. & Maulana, R., 2018. Sistem Kendali Navigasi Ar.Drone Quadcopter Dengan Prinsip Natural. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIHK)*, Volume II, pp. 380-386.
- Kuphaldt, T., 2006. *Lessons In Electric Circuits, Volume I – DC*. Bellingham: Design Scinece License.
- Muttaqin, A., Prasetyo, B H. & Setiawan, E., 2012. *Sinyal Rangkaian*. 1st ed. Malang: Program Teknologi Informasi dan Ilmu Komputer.
- Pallas, F A., Setyawan, G E. & Prasetyo, B H., 2018. Sistem Kendali Navigasi Quadcopter Menggunakan Suara Melalui. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIHK)*, Volume II, pp. 732-738.
- Pena, K., 2014. Accountability for Private Security Contractor Drone Operators on the U.S.-Mexico Border: Applying Lessons Learned from the Middle East. *Public Contract Law Journal*, Volume XLIV, p. 137.
- Piskroski, S., Brulez, N., Eline, P. & D'Haeyer, F., 2012. *AR.Drone Developer Guide Revision SDK 2.0*. San Andreas: Parrot.
- Pribadi, D A., Jonemaro, E M A. & Setyawan, G E., 2017. Implementasi Pengendalian Quadcopter Dengan Prinsip Virtual Reality Menggunakan Google Cardboard. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIHK)*, Volume I, pp. 1451-1458.
- Setyawan, G E., Setyawan, E. & Kurniawan, W., 2015. Sistem Kendali Ketinggian Quadcopter Menggunakan PID. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIHK)*, Volume II, pp. 125-131.
- Turo, T., 2015. Monitoring of Military Vehicel Battery. *IEEE*, pp. 1-4.
- Williard, N., 2011. Predicting Remaining Capacity of Batteries for UAVs and Electric Vehicle. *IMAPS Advanced Technology Workshop on High Reliability Microelectronics for Military Applications*, pp. 17-19.