

## Kaji Pengolahan Citra Pada *Quadcopter V9* untuk Misi Terbang Objek *Tracking and Following*

Mia Pertiwi<sup>1</sup>, Budi Hartono<sup>2</sup>

<sup>1</sup>Jurusan Teknik Mesin, Politeknik Negeri Bandung, Bandung 40012

<sup>1</sup>E-mail : mia.pertiwi.aer21@polban.ac.id

<sup>2</sup>E-mail : buhar@polban.ac.id

### ABSTRAK

Penggunaan *quadcopter* untuk misi terbang *object tracking and following* telah menjadi fokus riset teknologi, dengan penerapan teknologi pengolahan citra pada sektor transportasi, pertanian, industri, dan UAV untuk deteksi objek secara *autonomous*. Tujuan pengembangan *Quadcopter v9* adalah meningkatkan kemampuan *tracking and following* menggunakan pengolahan citra berbasis Python 3.11 dan pustaka OpenCV pada mini-PC Raspberry Pi 4 B. Metode melibatkan kamera webcam c70 pada *quadcopter* untuk memperoleh data citra dan pengolahan citra menggunakan nilai HSV. *Software in the loop* (SITL) digunakan untuk merepresentasikan dan meniru operasi sistem sebelum uji terbang *quadcopter*, menggunakan *software* Gazebo dan QgroundControl. Hasil simulasi SITL menunjukkan *quadcopter* dapat mengikuti objek dengan kecepatan 0,1 m/s. Uji terbang dilakukan pada ketinggian 2, 3, dan 4 meter untuk melacak objek berbentuk lingkaran merah berdiameter 40 dan 60 cm yang bergerak dengan kecepatan konstan. Hasil pengujian menunjukkan *Quadcopter v9* mampu melacak dan mengikuti objek, meskipun performa dipengaruhi oleh *delay*, *lagging*, dan kondisi angin, sehingga kecepatan bervariasi. Kondisi lingkungan dan kualitas komponen berperan penting dalam efektivitas misi terbang *object tracking and following*.

Kata Kunci: *quadcopter*, *image processing*, *tracking and following*, Python 3.11, OpenCV, SITL

*The use of quadcopters for object tracking and following missions has become a focus of technology research. technology, with the application of image processing technology in the transportation, agriculture, industry, and UAVs for autonomous object detection. The purpose of developing Quadcopter v9 is to improve the tracking and following capability using Python-based image processing. following capabilities using Python 3.11-based image processing and OpenCV library on a RPC mini-PC. OpenCV library on a Raspberry Pi 4 B mini-PC. The method involves a c70 webcam camera on the quadcopter to obtain image data and image processing using HSV values. HSV VALUES. Software in the loop (SITL) was used to represent and mimic the system operation before the quadcopter flight test. mimic the system operation before the quadcopter flight test, using Gazebo and QgroundControl software. and QgroundControl software. SITL simulation results show that the quadcopter can follow an object with a speed of 0.1 m/s. objects with a speed of 0.1 m/s. Flight tests were conducted at heights of 2, 3, and 4 meters to track a red circle-shaped object with a diameter of 40 and 60 cm. moving at a constant speed. The test results show that the Quadcopter v9 is able to track and follow objects, although performance is affected by delay, lagging, and wind conditions, resulting in variable speeds. Environmental conditions and component quality play an important role in effectiveness of the object tracking and following flight mission.*

**Keywords:** *quadcopter*, *image processing*, *tracking and following*, Python 3.11, OpenCV, SITL

### 1. PENDAHULUAN

Perkembangan teknologi transportasi di masa kini semakin krusial sehingga perusahaan berinovasi menciptakan mobil yang dapat berjalan sendiri dengan tingkat keamanan yang tinggi. Sistem *self-driving* pada *autonomous car* dengan menggunakan algoritma HSV berbasis Raspberry

Pi 4 model B berhasil dirancang sesuai spesifikasi desain sistem *self-driving* dan bisa berjalan sesuai rute yang sudah ditentukan [1].

**Hartono, et al. (2023)** menyusun penelitian yang berjudul “*Autonomous Quadcopter Image Processing for Simulated Search and Rescue Flights*”. Pada penelitian tersebut, *quadcopter*

memanfaatkan *software Mission Planner*, *quadcopter* mendemonstrasikan kemampuan penerbangan otonom meliputi *take off*, mempertahankan ketinggian yang ditentukan, dan melakukan *image processing* di antara *waypoint* yang telah ditentukan [2].

**F. N. Mahendra (2017)** menyusun laporan Tugas Akhir berjudul “*Quadcopter* dengan Sistem Pengolahan Citra: Kaji Pengolahan Citra untuk Misi Terbang *Object Tracking and Following*”. Tugas Akhir tersebut menggunakan HSV (*Hue, Saturation, Value*) di dalam pengujiannya, *quadcopter* melakukan *take off* secara otomatis dengan ketinggian bervariasi dimulai dari 1 – 5 meter dan dapat mendeteksi semua objek hanya pada ketinggian 2 meter. *Image processing* diaktifkan dan kamera berhasil mendeteksi objek. Kemudian objek yang telah terdeteksi akan ditarik secara manual dan *quadcopter* tersebut akan mengikuti objek secara *autonomous* [3].

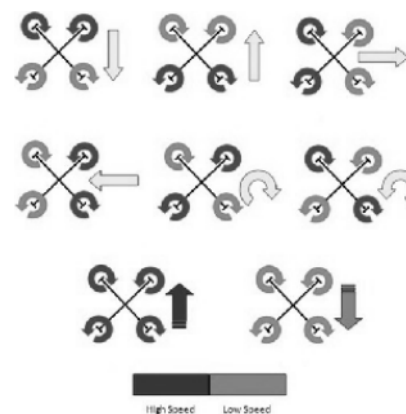
**A. T. Nugraha (2017)** menyusun penelitian *quadcopter* yang digunakan oleh Trisna memiliki metode kontrol *output feedback* yang digunakan pada *inner loop*, sedangkan struktur kontrol *Command-Generator Tracker* digunakan pada *outer loop*. Pada penelitian tersebut sistem kontrol dapat menstabilkan sudut sikap dari *quadcopter* dan dapat membuat *quadcopter* melakukan track sinyal referensi yang diberikan. Pada posisi tanpa gangguan, nilai maksimum *tracking* adalah 0.1 meter dan pada saat ada gangguan sebesar 0.28 meter. Sistem kontroler juga dapat membuat *quadcopter* melakukan *tracking* dengan referensi lintasan yang berbentuk persegi dan *spiral (helix)* [4].

Pada penelitian ini, sistem *quadcopter* dengan sistem *image processing* yang memiliki kemampuan untuk melakukan misi terbang *object tracking and following* dengan objek berbentuk lingkaran berwarna merah dengan diameter 40 cm dan 60 cm. *Quadcopter* dirakit dengan menggunakan mini-PC Raspberry Pi 4 Model B dan kamera webcam c270 untuk pendeteksiannya, Pixhawk 2.4.8 32bit sebagai *flight controller*, serta *Mission Planner* sebagai GCS (*Ground Control Station*). *Quadcopter* diterbangkan dengan menggunakan 2 jenis baterai yaitu 3 *cell* dan 4 *cell* untuk efisiensi waktu ketika uji terbang. Jenis baterai 3 *cell* digunakan untuk *power* ke Raspberry Pi, dan jenis baterai 4 *cell* digunakan untuk *power* ke Pixhawk.

## 2. LANDASAN TEORI

### 2.1 *Quadcopter*

*Quadcopter*, atau disebut juga *quadrotor*, adalah helikopter dengan empat rotor. Dalam *quadcopter* tersebut rotor diarahkan ke atas dan ditempatkan dalam formasi persegi dengan jarak yang sama dari pusat massa *quadcopter*. *Quadcopter* dikendalikan dengan menyesuaikan kecepatan sudut rotor yang diputar menggunakan motor listrik. *Quadcopter* didesain khas untuk kendaraan udara tanpa awak (UAV) kecil karena strukturnya yang sederhana. *Quadcopter* digunakan dalam pengawasan, pencarian dan penyelamatan, inspeksi konstruksi, dan lainnya [5]. Metode kontrol membutuhkan informasi yang akurat dari pengukuran posisi yang dilakukan dengan *accelerometer*, dan alat pengukur lainnya, seperti GPS. Gambar 1 merupakan konfigurasi dari gerakan *quadcopter*.



Gambar 1 Konfigurasi gerakan *quadcopter*

### 2.2 *Flight Controller*

*Flight controller* merupakan perangkat keras untuk memproses sistem *quadcopter*. Sebagian besar menggunakan prosesor yang beragam dimulai dari 8 bit sampai 32 bit. *Flight controller* berinteraksi dengan *Ground Control System* (GCS) [6] melalui titik komunikasi, misalnya, *Pulse Width Modulation* (PWM), *Controller Area Network* (CAN), dan *Universal Asynchronous Receiver/Transmitter* (UART) [7]. Gambar 2 merupakan *flight controller* yang digunakan dengan versi 2.4.8 32 bit.



Gambar 2 Flight controller 2.4.8 32 bit

### 2.3 Pengolahan Citra

Citra merupakan suatu representasi gambar, kemiripan, dan imitasi dari objek. Citra sebagai keluaran suatu sistem perekaman data dapat bersifat *optic* berupa gambar, bersifat analog berupa sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan. Pengolahan citra adalah proses untuk memperbaiki kualitas, mengambil dan mengubah informasi pada suatu citra [8]. Meskipun sebuah citra kaya informasi, namun sering kali citra yang kita miliki mengalami penurunan mutu (*degradasi*), misalnya mengandung cacat atau derau (*noise*), warnanya terlalu kontras, kabur (*blurring*), kurang tajam dan sebagainya. Citra menjadi lebih sulit diinterpretasi karena informasi yang disampaikan oleh citra tersebut menjadi berkurang.

### 2.4 Misi Terbang Objek *Tracking and Following*

Misi terbang yang dilaksanakan adalah *quadcopter* terbang secara *autonomous* dari *take off*, kemudian menuju posisi *altitude hold*. Setelah itu, ketika objek terdeteksi, maka *quadcopter* akan secara *autonomous* melakukan uji terbang *object tracking and following*. Hal ini mensimulasikan sistem operasi pelacakan objek yang bergerak dan diimplementasikan pada *quadcopter*. Jika objek tidak terdeteksi, maka *quadcopter landing* melalui *mission planer*.

## 3. METODOLOGI

### 3.1 Persiapan Perangkat Keras

Persiapan perangkat keras mencakup perakitan *quadcopter*, kalibrasi *quadcopter*, uji terbang manual, dan uji terbang PID *autotune*. Proses perakitan *quadcopter* melibatkan penyusunan dan

penggabungan berbagai komponen hingga menjadi sebuah *quadcopter* yang siap diterbangkan. Frame yang dipilih yaitu konfigurasi wide-x. Kalibrasi *quadcopter* dilakukan melalui *Mission Planer* untuk mengkalibrasi level, *accelerometer*, *compass*, radio, ESC, *flight mode*, dan *fail safe*. *Mission Planer* merupakan aplikasi atau *software* yang berguna sebagai sistem kendali UAV dan juga berperan sebagai *Ground Control Station (GCS)*. *Mission Planer* digunakan untuk pengendalian serta monitoring *quadcopter* selama misi penerbangan berlangsung. Aplikasi ini juga berfungsi untuk memprogram *quadcopter* yang gerakannya dikendalikan oleh mode autopilot [9]. Uji terbang manual dilakukan untuk memastikan bahwa semua komponen *quadcopter* berfungsi dengan baik dan proses kalibrasi telah dilakukan dengan benar, sehingga *quadcopter* dapat terbang dengan aman. Uji terbang PID *autotune* bertujuan untuk meningkatkan performa terbang *quadcopter*, sehingga dapat terbang stabil dan terkendali. *Quadcopter* mendapatkan *supply* energi dari baterai yang dimaksud dengan baterai adalah alat yang berfungsi untuk menyimpan energi listrik dalam bentuk kimia, setelah itu dikonversi menjadi energi listrik untuk memperoleh arus listrik yang diperlukan [10]. ESC merupakan sebuah modul rangkaian elektronik yang berfungsi untuk mengatur suplai arus putaran motor dengan mengatur suplai arus yang disesuaikan dengan kebutuhan *brushless motor* [11]. Propeler merupakan komponen sistem propulsi yang berfungsi untuk mentransmisikan daya dengan mengonversi gerakan rotasi menjadi gaya dorong (*thrust*) [12]. *Quadcopter* terbang dengan mendapatkan sinyal dari GPS. GPS merupakan sistem navigasi dengan menggunakan satelit. Sensor bekerja dengan menerima data yang berisi waktu dan posisi satelit ketika data dikirimkan, melalui sinyal yang dipancarkan oleh satelit GPS [13]. *Quadcopter* membutuhkan kamera untuk merekam gambar dari sebuah objek sebagai input dari sistem *image processing*. Kamera merupakan sebuah alat yang digunakan untuk keperluan mengabadikan suatu objek yang akan dijadikan sebuah gambar ataupun video dalam keadaan diam ataupun bergerak sebagai hasil proyeksi pada sistem lensa [14]. Di dalam *Quadcopter V9* terdapat *flight controller* yang memberikan kemampuan untuk mengontrol *quadcopter* terbang secara *autonomous*. Dalam proses terbangnya, *flight controller* dapat memberikan fleksibilitas yang tinggi [15].

Gambar 3 merupakan hasil perakitan *Quadcopter V9* yang telah dilaksanakan.



Gambar 3 Hasil perakitan *quadcopter*

### 3.2 Pemrograman Image Processing

Tahapan berikutnya adalah perancangan program *image processing* menggunakan bahasa pemrograman Python, pustaka Dronekit-Python dan pustaka OpenCV. Proses dimulai dengan merancang serta memprogram sistem *image processing*. Raspberry Pi merupakan modul komputer kecil dengan *input* dan *output* yang mirip dengan *mikrokontroler*. Meskipun Raspberry Pi berukuran kecil, namun dapat mengendalikan program berat yang dipakai dalam permainan komputer [1]. Langkah awal yang dilakukan oleh Raspberry Pi adalah mengonversi warna dari RGB ke HSV, kemudian menentukan warna yang ingin diidentifikasi dengan mengatur *Hue*, *Saturation*, dan *Value*, serta mencari nilai minimum dari kontur objek yang ditampilkan di dalam *frame*. Selanjutnya, ditambahkan fungsi untuk menemukan titik tengah dari kontur dan menjalankan perintah sesuai dengan posisi kontur pada *frame*. Raspberry Pi akan mengirimkan perintah kepada Pixhawk untuk mengatur kecepatan motor sesuai dengan posisi kontur pada *frame* dan mode *hovering*.

### 3.3 Integrasi Perangkat Keras dan Sistem Image Processing

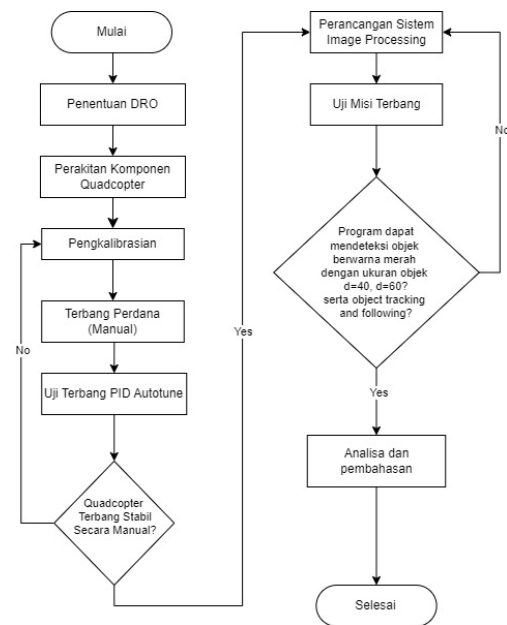
Dalam proses perancangan sistem untuk uji terbang *object tracking and following* ini, diperlukan perangkat yang mampu menjalankan program pengolahan citra. Perangkat yang mendukung kebutuhan tersebut adalah Raspberry Pi 4 Model B. Raspberry Pi ini perlu dikonfigurasi dengan perangkat lain, seperti *flight controller*, laptop, dan perangkat lunak yang

diperlukan. Raspberry Pi perlu dihubungkan ke laptop untuk menginstal sistem operasi (OS) yang digunakan untuk menjalankan program pengolahan citra. Untuk menghubungkan Raspberry Pi dengan laptop, diperlukan perangkat tambahan seperti kartu microSD (*Secure Digital*), adapter, dan kabel LAN (*Local Area Network*). Beberapa perangkat yang diperlukan untuk melakukan konfigurasi antara Raspberry Pi dan laptop adalah:

1. Raspberry Pi Imager, digunakan untuk menginstal OS pada Raspberry Pi
2. SD Card, untuk menyimpan data saat proses instalasi pada Raspberry Pi 4.
3. Card Reader, alat yang memungkinkan laptop membaca SD Card.
4. VNC Viewer, perangkat lunak untuk memonitor dan menjalankan Raspberry Pi menggunakan IP Scanner.
5. PuTTY, untuk mengakses Raspberry Pi melalui SSH.
6. Advanced IP Scanner, digunakan untuk mencari alamat IP Raspberry Pi yang terhubung ke jaringan Wi-Fi.

### 3.4 Uji Terbang *Object Tracking and Following*

Berikut Gambar 4 merupakan tahap penjelasan metodologi yang digunakan dalam penyelesaian masalah dalam laporan karya ilmiah ini:



Gambar 4 *Flowchart object tracking and following*

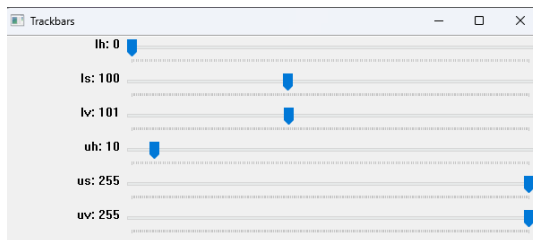


Dalam uji terbang ini, *quadcopter* yang dilengkapi dengan kamera akan mencari objek pada saat posisi *take off* dan *hovering*. Gambar yang diambil oleh kamera akan dikirim ke *ground control station* melalui Raspberry Pi. Jika pada *ground control station* terdeteksi bahwa kamera menemukan objek berupa lingkaran merah, maka *quadcopter* akan mengikuti objek tersebut dan melakukan *tracking and following*. Jika objek tidak terdeteksi, maka *quadcopter* akan landing secara otomatis.

## 4. HASIL PENGUJIAN

### 4.1 Hasil Pengolahan Citra

Setelah melakukan kalibrasi dan uji terbang manual, langkah selanjutnya yaitu pemrograman *image processing*. Proses awal dalam pengolahan citra dimulai dengan menentukan nilai HSV dari objek yang akan dipantau. Mengingat objek yang ingin dipantau berwarna merah, dibutuhkan metode yang efisien dan akurat untuk menentukan nilai HSV dari warna tersebut. Dengan menggunakan pemrograman VSCode pada terminal Raspberry Pi, coding dapat dijalankan. Pencarian nilai HSV dari objek yang ingin dipantau dilakukan melalui penggunaan trackbar, terlihat pada Gambar 6, yang memiliki enam kontrol geser untuk menentukan nilai *upper* dan *lower* HSV.



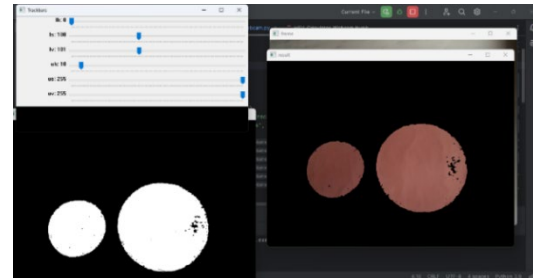
Gambar 5 Trackbar nilai HSV



Gambar 6 Objek yang dideteksi

Pada Trackbar tersebut mendapatkan nilai lh (0), ls (100), lv (101), uh (10), us (255), uv(255) nilai tersebut didapatkan dengan menggeser trackbar untuk memastikan nilai yang pas untuk warna merah. Setelah mendapatkan nilai HSV dari objek yang dilacak, langkah berikutnya adalah

mengintegrasikan nilai-nilai tersebut ke dalam program pelacakan agar dapat mengenali objek dengan nilai HSV tersebut secara khusus. Setelah program dijalankan, akan muncul tampilan yang menunjukkan objek lingkaran merah telah terdeteksi, seperti yang terlihat pada Gambar 7 dan Gambar 8.



Gambar 8 Hasil nilai HSV

Pada Gambar 8 yang berwarna merah merupakan (*input*) objek yang di deteksi oleh pemrograman HSV dan berwarna putih merupakan hasil (*output*) dari pemrograman HSV yang telah dilakukan.

### 4.2 Simulasi SITL

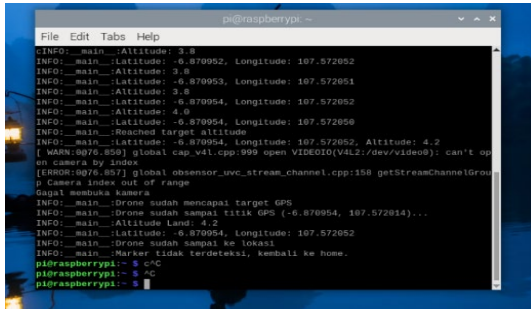
Simulasi merupakan suatu proses atau teknik yang digunakan untuk merepresentasikan dan meniru operasi dari suatu sistem nyata atau konseptual. Dilakukan dengan menggunakan model matematis atau komputer untuk mendapatkan hasil yang meniru perilaku dari sistem yang sebenarnya. Simulasi ini dilakukan sebelum uji terbang secara langsung ke *quadcopter*, berfungsi untuk melihat apakah pemrograman dapat berjalan sesuai dengan misi terbang yang diharapkan. Simulasi SITL menggunakan *software* Qgroundcontrol dan Gazebo untuk melihat pergerakan *quadcopter* secara *autonomous*.

Pada simulasi ini, *quadcopter* dapat mendeteksi objek dan bergerak dengan kecepatan 0.1 m/s. Konteks pergerakan tersebut, *altitude* tetap (konstan) dengan seiring berjalannya waktu. Sementara itu, *latitude* dan *longitude* juga menunjukkan ke konstanan mengindikasikan bahwa lokasi geografis tetap dan tidak berubah sepanjang pengamatan.

### 4.3 Hasil Uji Terbang

Uji coba *quadcopter* dengan misi terbang *object tracking and following* dilengkapi dengan kamera webcam c270 untuk merekam gambar sebuah objek sebagai *input* dari sistem *image processing* untuk mendeteksi dan mengikuti objek. Uji

terbang dilakukan di lapangan basket Politeknik Negeri Bandung dengan diameter objek 40 cm dan 60 cm. Pengujian tersebut menggunakan laptop sebagai *Ground Control Station (GCS)* untuk mengaktifkan program misi terbang dan memantau pergerakan *quadcopter* melalui *Mission Planner*. Tampilan pada laptop dapat dilihat pada Gambar 9.



Gambar 9 Tampilan *Ground Control Station*

#### 4.4 Hasil Deteksi Objek

Tabel 1 merupakan hasil pendeteksian objek lingkaran berwarna merah dengan diameter 40cm

Tabel 1 Hasil pendeteksian 40 cm

Ketinggian [m]	Deskripsi
2	objek terdeteksi
3	objek terdeteksi
4	objek tidak terdeteksi

Pada ketinggian 2 meter, objek dapat terdeteksi namun objek sempit terlipat sehingga *quadcopter* mendeteksi kurang jelas, ketika ketinggian 3 meter objek terdeteksi namun karena kondisi lingkungan yang berangin sehingga mengakibatkan *quadcopter* hanya mendeteksi sebagian, dan ketika pada ketinggian 4 meter objek tidak terdeteksi karena keberadaan bayangan di area pengujian membuat program kehilangan fokus dan mendeteksi bayangan sebagai gantinya.

Pada Tabel 2 merupakan hasil pendeteksian objek lingkaran berwarna merah dengan diameter 60cm.

Tabel 2 Hasil pendeteksian 60 cm

Ketinggian [m]	Deskripsi
2	objek terdeteksi
3	objek terdeteksi
4	objek terdeteksi

Pada ketinggian 2 meter, objek terdeteksi dan program dapat mengikuti pergerakan objek dengan cukup baik, meskipun kadang-kadang *quadcopter* kesulitan jika objek bergerak terlalu cepat atau keluar dari bingkai pendeteksian, ketika

ketinggian 3 meter objek terdeteksi dan pergerakan objek dapat diikuti dengan baik oleh *quadcopter* walaupun dalam kondisi lingkungan yang berangin, dan ketika pada ketinggian 4 meter objek berhasil di deteksi namun spanduk dengan lingkaran merah sempit terlipat yang menyebabkan program kehilangan fokus.

## 5. KESIMPULAN

*Quadcopter* berhasil dirakit dengan konfigurasi wide-x, dan mampu terbang secara *autonomous* dan dengan misi terbang *object tracking and following* berhasil mengikuti objek lingkaran berwarna merah dengan diameter 40 cm dan 60 cm. Namun *quadcopter* belum dapat terbang dengan ideal karena dipengaruhi faktor kualitas komponen dan faktor cuaca dan lingkungan saat uji terbang.

Sistem pengolahan citra telah berhasil dirancang, baik dalam simulasi SITL maupun dalam uji terbang *quadcopter*. Pengolahan citra dijalankan dengan menggunakan pemrograman VSCode pada terminal Raspberry Pi dan telah berhasil dengan baik, dapat mendeteksi objek nilai lh (0), ls (100), lv (101), uh (10), us (255), uv(255).

Uji terbang *autonomous quadcopter* telah berhasil dilakukan. Baik dalam simulasi SITL maupun saat uji terbang. Dalam simulasi SITL *quadcopter* dapat terbang dan mengikuti objek dengan kecepatan 0.1 m/s. Dalam uji terbang perangkat keras, *quadcopter* hanya mendeteksi pada ketinggian 2 dan 3 meter pada diameter 40, sedangkan pada diameter 60 cm *quadcopter* dapat mendeteksi seluruh ketinggian hal tersebut dikarenakan pengaruh cuaca dan komponen *quadcopter* yang kurang mumpuni seperti dalam perbedaan putaran motor *quadcopter*.

## DAFTAR PUSTAKA

- [1] F. B. Setiawan, P. U. Sutrisno, L. H. Pratomo, and S. Riyadi, "Penerapan Algoritma HSV pada Autonomous Car untuk Sistem Self-Driving Berbasis Raspberry Pi 4," *Jurnal Rekayasa Elektrika*, vol. 18, no. 4, pp. 255–262, 2022, doi: 10.17529/jre.v18i4.27495.
- [2] B. Hartono, P. N. Bandung, M. R. Zuhri, P. N. Bandung, N. Fauzan, and P. N. Bandung, "Autonomous *Quadcopter* Image Processing for Simulated Search and Rescue Flight," *Search and Rescue*

- (SAR) Flight Simulations Utilizing Low-Budget Autonomous Quadcopter and Image Processing Technology, 2023.
- [3] F. H. Mahendra, *Quadcopter dengan Sistem Pengolahan Citra: Kaji Pengolahan Citra untuk Misi Terbang Objek Tracking and Following*. 2020.
- [4] A. T. Nugraha, "Tracking Quadcopter Menggunakan Metode Command-Generator Tracker Dengan Efek Integrator," *Jurnal Teknologi Elektro*, vol. 8, no. 2, pp. 123–130, 2017.
- [5] M. Latif and B. Hairil, "Perancangan Sistem Autonomous Quadcopter," *Program Studi Mekatronika*, no. November, pp. 1–5, 2014.
- [6] E. Ebeid, M. Skriver, and J. Jin, "A Survey on Open-Source Flight Control Platforms of Unmanned Aerial Vehicle," *Proceedings - 20th Euromicro Conference on Digital System Design, DSD 2017*, no. September 2018, pp. 396–402, 2017, doi: 10.1109/DSD.2017.30.
- [7] N. Taryana, U. A. Albayumi, and A. Rohim, "Implementasi Prototype Alat Uji Flight Control Actuator Pesawat Berbasis Mikrokontroler Arduino Uno," 2017.
- [8] T. Theodoridis and J. Kraemer, "Pengolahan Citra," pp. 9–28, 2014.
- [9] L. Herlambang, E. Kuncoro, and M. M. Huda, "The Implementation of Autonomous Waypoint Reconnaissance Plane (Unmanned Aerial Vehicle) UAV Galak-24 use with Mission Planner," *Jurnal Telkommil*, vol. 2, no. Oktober, pp. 47–55, 2021, doi: 10.54317/kom.v2ioktober.183.
- [10] M. Nasution, "Muslih Nasution Karakteristik Baterai Sebagai Penyimpan Energi Listrik Secara Spesifik," *Journal of Electrical Technology*, vol. 6, no. 1, pp. 35–40, 2021.
- [11] T. B. Prakoso, "Implementasi Dji Naza M-Lite Pada Quadcopter," *Jurnal Mikrotek*, vol. 1, no. 4, pp. 192–197, 2015.
- [12] Unimar, "Pengertian Propeller," maret. [Online]. Available: [http://repository.unimar-amni.ac.id/1750/2/12.BAB\\_2\\_awal.rtf](http://repository.unimar-amni.ac.id/1750/2/12.BAB_2_awal.rtf)
- [13] Suprianto, "Komponen dan Prinsip Kerja Quadcopter," oktober 13. [Online]. Available: <https://blog.unnes.ac.id/antosupri/komponen-dan-prinsip-kerja-quadcopter/>
- [14] R. W. Bowman, B. Vodenicharski, J. T. Collins, and J. Stirling, "Flat-Field and Colour Correction for the Raspberry Pi Camera Module," *Journal of Open Hardware*, vol. 4, no. 1, pp. 1–19, 2020, doi: 10.5334/joh.20.
- [15] R. D. Kusumanto, A. N. Tomponu, and S. Pambudi, "Klasifikasi Warna Menggunakan Pengolahan Model Warna HSV Abstrak," *Jurnal Ilmiah Teknik Elektro*, vol. 2, no. 2, pp. 83–87, 2011.