

RANCANG BANGUN DAN IMPLEMENTASI PROTOTIPE PENDETEKSI DAN PEMADAM API MENGGUNAKAN IMAGE PROCESSING PADA QUADCOPTER

DESIGN AND IMPLEMENTATION PROTOTYPE OF FIRE DETECTION AND EXTINGUISHER BASED IMAGE PROCESSING ON QUADCOPTER

Sepfrans Josua Hutasoit¹, Erwin Susanto, S.T., M.T.², Ph.D., Ramdhan Nugraha, S.Pd., M.T.³

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

¹sepfrans.josua@gmail.com, ²erwinelektro@telkomuniversity.ac.id, ³ramdhan@telkomuniversity.ac.id

Abstrak

Unmanned Aerial Vehicle (UAV) merupakan robot penjelajah udara tanpa menggunakan pilot di dalamnya. Salah satu jenis UAV adalah *quadcopter*. Saat ini, *quadcopter* telah berkembang dengan sangat pesat. Keunggulan *quadcopter* dapat terbilang sangat membantu. Pemakaian *quadcopter* dapat dilakukan di berbagai macam tempat bahkan di tempat-tempat yang sulit sekalipun. Karena keunggulan tersebut, *quadcopter* banyak digunakan dalam berbagai macam aplikasi. Salah satu jenis aplikasi yang sedang dikembangkan adalah *quadcopter* pemadam api.

Berdasarkan latar belakang tersebut, penulis akan merancang bangun dan mengimplementasi sebuah prototipe yaitu deteksi dan pemadam api menggunakan *image processing* pada *quadcopter*. Pada *image processing*, penulis akan memakai kamera sebagai visual gambar pada *quadcopter*. Hasil gambar tersebut juga akan dikirim langsung ke *ground station*. Dari setiap hasil gambar, *quadcopter* akan menganalisis apakah gambar yang diambil memiliki titik apinya atau tidak. Jika pada visual gambar terdapat titik api, *quadcopter* akan memadamkannya. Pada *image processing*, penulis akan memakai metode *Color Filtering HSV (Hue, Saturation, Value)* sebagai perbandingan warna pada area jangkauan *quadcopter*.

Hasil tugas akhir ini adalah setiap area yang dapat dijangkau oleh *quadcopter* dapat mendeteksi api jika terdapat titik api maupun kebakaran. Riset ini akan digunakan di tempat yang rentan akan terjadi titik api sehingga api tidak menyebar lebih luas.

Kata kunci : *Unmanned Aerial Vehicle, quadcopter, remote control, image processing, Ground Station, Color Filtering, HSV (Hue, Saturation, Value).*

Abstract

Unmanned Aerial Vehicle (UAV) is an air explorer robot without using a pilot inside. One kind of UAV is *quadcopter*. Currently, *quadcopter* has been growing very rapidly. *Quadcopter* excellence can be calculated very helpful. The use of *quadcopter* can be done in all kinds of places even in places that are difficult ones. Because of these advantages, *quadcopter* widely used in various applications. One type of application being developed is a *fire extinguisher quadcopter*.

Based on this background, the author will design and implement a prototype that detects and extinguish a fire using *image processing* on *quadcopter*. On the *image processing*, the author will use the camera as a visual image on *quadcopter*. The image results are sent directly to a *ground station*. From each of the image results, *quadcopter* will analyze whether the images taken have fire points or not. If on the visual contained fire points, *quadcopter* will extinguish it. On the *image processing*, the author will use a *Color Filtering HSV (Hue, Saturation, Value)* method as a comparison of the color on the coverage area of *quadcopter*.

The results of this thesis are every area that can be reached by *quadcopter* can detect a fire if contained fire points or fires. This research will be used in places that are vulnerable to fire points so that the fire did not spread more widely.

Keyword : *Unmanned Aerial Vehicle, quadcopter, remote control, image processing, Ground Station, Color Filtering, HSV (Hue, Saturation, Value).*

1. Pendahuluan

1.1 Latar Belakang

Perubahan iklim dari musim dingin menuju musim panas membuat beberapa lokasi di Indonesia menjadi rentan terhadap kebakaran. Titik api yang semula sedikit di saat musim dingin bertambah pesat di musim kemarau, terutama titik api yang medannya sulit dijangkau seperti di hutan. Hal seperti inilah yang menjadi penyebab titik api di Indonesia semakin banyak. Berdasarkan informasi dari WWF Indonesia, Indonesia memiliki hamparan hutan yang luas, dengan luas sebesar 99,6 juta hektar atau dalam persentase 52,3% dari total wilayah Indonesia[1].

Pada abad ini, ilmu pengetahuan sudah dapat dibilang sangat berkembang, baik itu pada bidang kesehatan, industri, militer, dll. Hal ini menambah keuntungan pada masyarakat. Sejumlah pakar teknologi dunia mengembangkan robot sebagai alat dalam mempermudah pekerjaan manusia. Salah satunya adalah pengembangan robot *quadcopter* dalam mengamati tempat-tempat yang hampir tidak dapat dijangkau oleh manusia. *Quadcopter* ini masih dalam tahap penelitian lanjut dari dunia robotika.

Quadcopter adalah pengembangan dari helicopter yang hanya memiliki sebuah *rotor*. Teknologi yang terdapat dalam *quadcopter* ini menggunakan sinkronisasi antara keempat *rotor* yang dikonfigurasi dalam bentuk *frame plus (+)* dimana *rotor* depan dan belakang berputar searah jarum jam, sedangkan *rotor* sebelah kanan dan kiri bergerak berlawanan arah jarum jam[2]. Kegunaan *quadcopter* sudah banyak digunakan di berbagai aplikasi. Pada penelitian ini, aplikasi yang ingin dikembangkan adalah *quadcopter* dengan kemampuan *image processing* dalam mendeteksi dan memadamkan api.

Berdasarkan latar belakang yang telah diuraikan tersebut maka tugas akhir ini dirancang dan diimplementasikan pada prototipe yaitu deteksi dan pemadam api menggunakan *image processing* pada *quadcopter*. Sistem ini bertujuan mengambil deteksi api pada gambar di area *quadcopter*. Kemudian *quadcopter* menuju titik api dan setelah di titik api, *quadcopter* akan memadamkan api tersebut. Sistem menggunakan *Raspberry Pi 2* sebagai otak dalam mengolah dan mengirim gambar yang ditangkap dari kamera. Untuk mendapatkan informasi koordinat yang tepat bagi pemetaan udara menggunakan *Flight Controller* yang terhubung ke modul GPS dan *Compass*. Metode dalam pengolahan gambar yang diambil dari kamera, menggunakan metode *Color Filtering HSV (Hue, Saturation, Value)*.

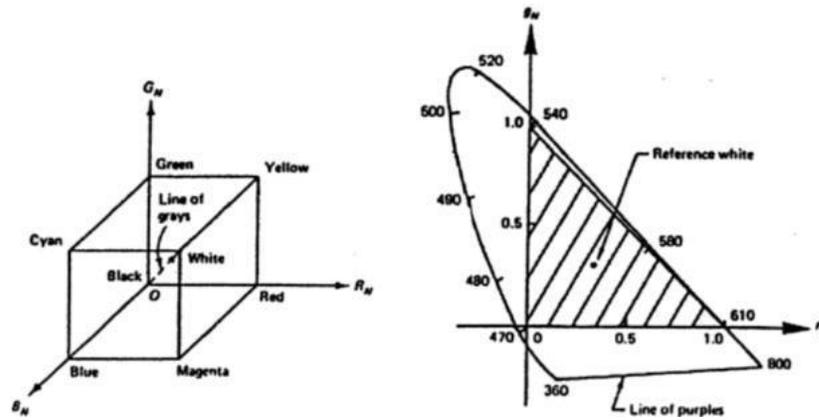
2. Dasar Teori

2.1 Citra RGB

NTSC (National Television System Committee) adalah komite nasional yang menciptakan standar warna (R_N , G_N , B_N) untuk pesawat penerima televisi. Komite ini menentukan tiga buah phosphor utama dari *spectrum* sinar yang ada yaitu merah, hijau, dan biru. Untuk warna referensi adalah warna putih dimana merupakan gabungan (*tristimulus*) dari tiga buah warna utama dengan nilai $R_N = G_N = B_N = 1$. Tabel 2.1 berikut menunjukkan nilai gabungan dari ketiga warna tersebut untuk beberapa warna-warna utama dari koordinat warna NTSC sedangkan r_N, g_N, b_N adalah nilai-nilai krominannya[6].

Tabel 2.1 Gabungan Nilai Warna Dasar dan Krominan untuk Warna Pokok pada Sistem Penerima Utama NTSC[6]

	Merah	Kuning	Hijau	Cyan	Biru	Magenta	Putih	Hitam
R_N	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0
G_N	0.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0
B_N	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0
r_N	1.0	0.5	0.0	0.0	0.0	0.5	0.333	0.333
g_N	0.0	0.5	1.0	0.5	0.0	0.0	0.333	0.333
b_N	0.0	0.0	0.0	0.5	1.0	0.5	0.333	0.333



Gambar 2.3 Tristimulus Warna dan Diagram Krominan Sistem Penerima Utama NTSC[6]

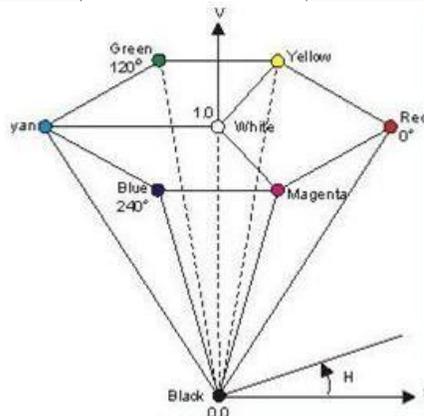
Koordinat ruang warna untuk sistem transmisi NTSC (Y, I, Q) dibangun untuk fasilitas transmisi gambar berwarna yang menggunakan jalur televisi *monochrome* yang telah ada tanpa menambah lebar pita yang diperlukan. Koordinat Y adalah melambangkan *luminance* sedangkan I dan Q melambangkan *hue* dan *saturation* dari sebuah warna yang memiliki lebar pita jauh lebih kecil dibandingkan sinyal *luminance*. Komponen I, Q ditransmisikan pada sebuah jalur sub-pembawa menggunakan modulasi *quadrature* sehingga *spectrum* dari I, Q tidak saling tumpang tindih dengan spektrum Y dan lebar pita yang diperlukan untuk transmisi tidak berubah. Hubungan sistem Y, I, Q dengan sistem R_N, G_N, B_N merupakan transformasi *linear* seperti berikut[6].

$$(2.1)$$

Untuk menentukan sistem kebalikannya adalah dengan mentransformasinya terhadap matrik inversnya.

2.2 Ruang Warna HSV

HSV(*Hue, Saturation, Value*) *color model* adalah transformasi non linear ruang warna pada RGB. Warna tersebut merupakan kombinasi dari 3 nilai, yaitu *Hue* (H), *Saturation* (S), dan *Value* (V). Nilai-nilai pada warna ini dapat dilihat pada representasi ruang warna HSV berikut.



Gambar 2.4 Representasi Ruang Warna HSV[6]

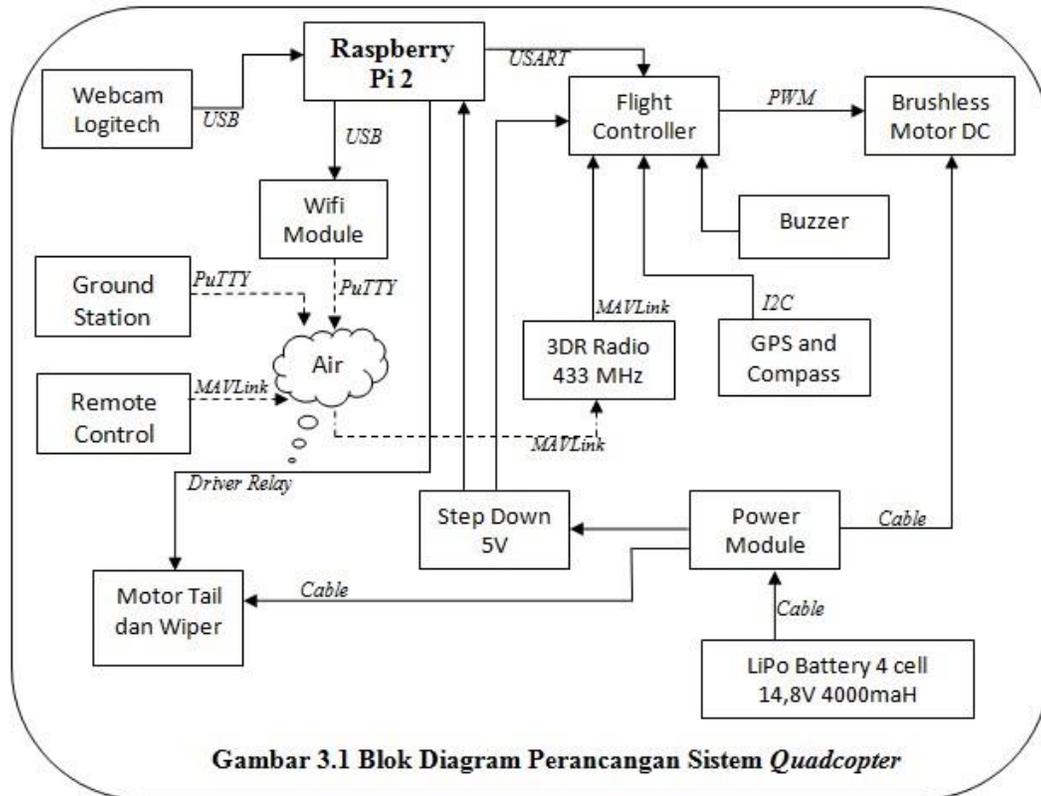
Penjelasan mengenai ketiga nilai tersebut sebagai berikut :

- *Hue* (H) : Jenis-jenis pada warna (misalnya merah, hijau, atau kuning). Ini direpresentasikan juga sebagai tingkat sudut yang nilainya berkisar dari 0° sampai 360° (meskipun untuk beberapa aplikasi dinormalisasi dari 0% sampai 100%)
- *Saturation* (S) : merupakan direpresentasikan sebagai jarak dari sumbu cahaya hitam-putih. Nilainya berkisar dari 0% sampai 100%.
- *Value* (V) : direpresentasikan sebagai tinggi pada poros hitam putih. Kemungkinan jarak nilai berkisar dari 0% sampai 100%. Nilai 0 selalu hitam. Berdasarkan pada *saturation*, 100 bisa menjadi putih atau tingkat *saturation* yang lebih bahkan kurang.

Suatu warna dengan nilai *value* 100% akan tampak secerah mungkin dan suatu warna dengan nilai *value* 0 akan tampak segelap mungkin. Sebagai contoh jika *hue* adalah merah dan *value* bernilai tinggi maka warna kelihatan cerah tetapi ketika nilai *value* rendah maka warna tersebut akan kelihatan gelap[6].

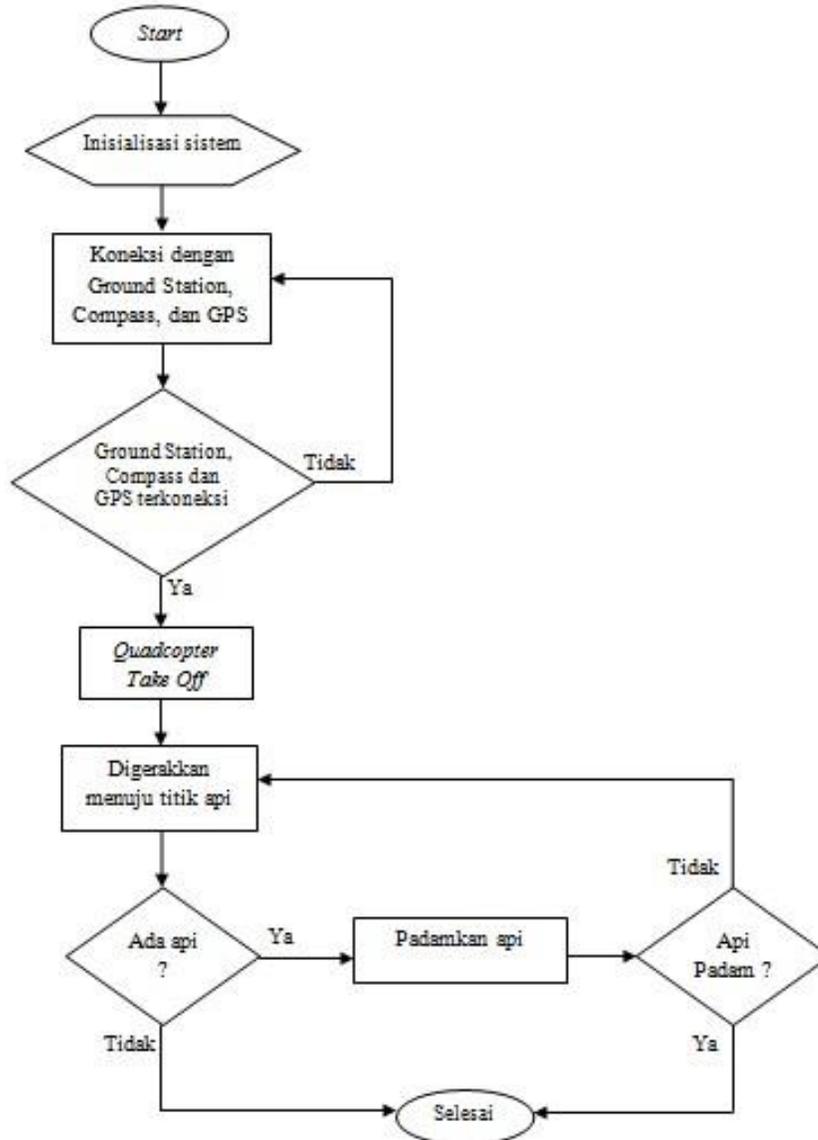
3. Perancangan dan Implementasi

3.1 Perancangan Sistem



Pada perancangan sistem di atas terdapat Raspberry Pi 2 sebagai otak dari seluruh sistem. Komunikasi Raspberry Pi 2 ke *ground station* menggunakan wifi modul yang dihubungkan melalui port USB di Raspberry Pi 2. Sedangkan komunikasi ke *flight controller* menggunakan USART (rx dan tx). Untuk *input* data, menggunakan webcam Logitech C525 yang dihubungkan juga ke Raspberry Pi 2 menggunakan kabel USB. Untuk sistem *take off* dan *landing* menggunakan *Remote Control* yang terhubung melalui frekuensi radio ke *Radio Telemetry* 433 MHz, kemudian dari *Radio Telemetry* 433MHz terhubung ke *flight controller*. Pada *quadcopter* digunakan GPS dan *Compass* yang terhubung ke *Flight Controller* melalui komunikasi I2C. *Buzzer* berfungsi sebagai sistem peringatan yang akan disetting pada *flight controller*. *Flight controller* juga berperan untuk mengatur kecepatan pwm motor saat melakukan *yaw*, *pitch*, *roll*, dan *throttle*. Untuk sistem pemadam api, menggunakan motor EDF yang terhubung ke Raspberry Pi 2.

3.2 Perancangan Perangkat Lunak (Software)



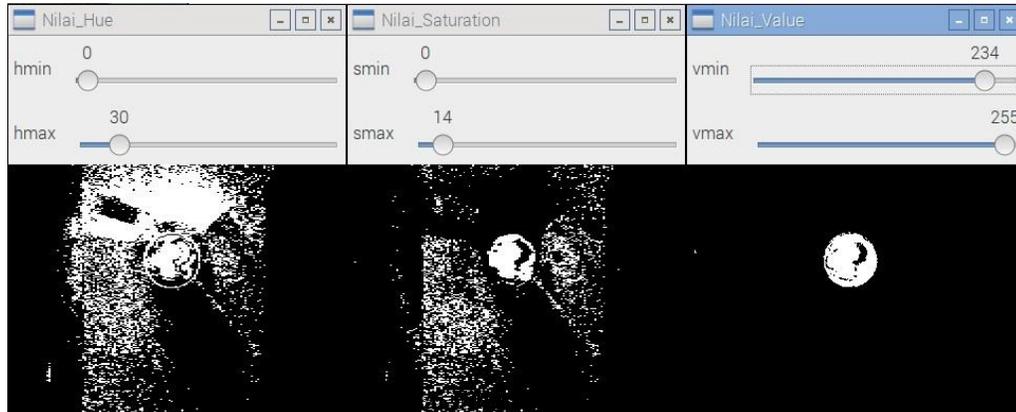
Gambar 3.2 Flowchart Software dan Sistem

Pada flowchart di atas menjelaskan mengenai cara kerja keseluruhan sistem dan software quadcopter. Pada komunikasi antara quadcopter dengan ground-station menggunakan Radio Telemetry 433MHz. Pada tahap pertama, sistem akan melakukan inisialisasi. Kemudian sistem akan memeriksa koneksi Ground Station, GPS(Global Positioning System), dan Compass pada quadcopter. Jika Ground Station, GPS dan Compass tidak terdeteksi, maka sistem akan memeriksa ulang kembali koneksi tersebut. Jika sudah terdeteksi, sistem akan menuju perintah selanjutnya yaitu quadcopter melakukan take off(lepas landas) menggunakan Remote Control. Setelah quadcopter berhasil take off(lepas landas), quadcopter akan mulai mengambil gambar pada area yang dilaluinya menggunakan kamera webcam Logitech C525. Setelah mengambil gambar, quadcopter akan mendeteksi pada gambar apakah ada api atau tidak. Untuk melakukan perintah deteksi api tersebut terdapat pada pesan yang dikirim melalui Ground Station. Jika terdeteksi api, maka quadcopter menuju perintah selanjutnya yaitu quadcopter akan menuju titik api dan memadamkan api tersebut. Setelah melakukan perintah tersebut, quadcopter kembali mendeteksi gambar apakah masih ada api atau tidak. Jika tidak terdeteksi api, quadcopter akan melakukan perintah landing.

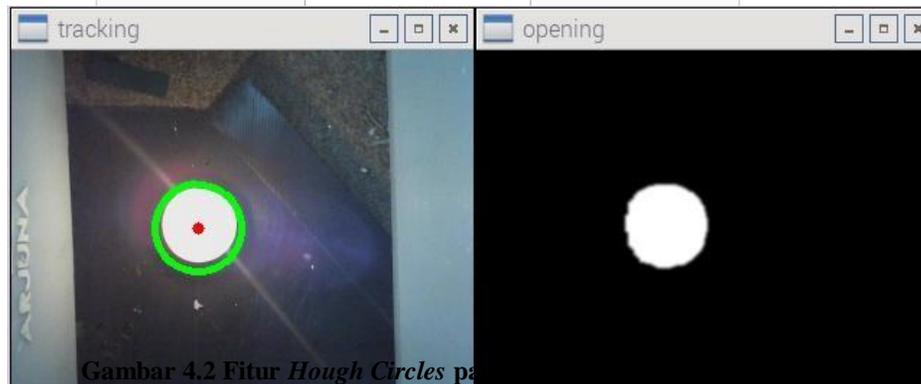
4. Pengujian dan Analisis Sistem

4.1 Pengujian Color Filtering HSV

Pada pengujian *color filtering* HSV ini dilakukan setting nilai *minimum* dan *maximum* pada *Hue*, *Saturation*, dan *Value*. Sebelumnya batas nilai *minimum* dan *maximum* berkisar dari 0 hingga 255. Agar nilai tersebut dapat diubah sesuai dengan warna objek yang akan dideteksi yaitu menggunakan *trackbar*. Objek yang digunakan pada pengujian kali ini adalah lilin. Setelah mendapatkan nilai yang tepat untuk *filtering* HSV ini, penguji mendapatkan hasil *filter* gambar yang tepat untuk warna objek yang akan dideteksi. Untuk pengujian kali ini, nilai yang telah didapat adalah *hmin*(*Hue minimum*) 0, *hmax*(*Hue maximum*) 30, *smin*(*Saturation minimum*) 0, *smax*(*Saturation maximum*) 14, *vmin*(*Value minimum*) 234, dan *vmax*(*Value maximum*) 255. Untuk lebih jelas bisa dilihat pada gambar 4.1.



Setelah mendapatkan hasil *filter* yang bagus pada deteksi objek tersebut, dilakukan penambahan fitur deteksi untuk melihat ketepatan objek yang akan dideteksi pada gambar asli. Fitur tersebut adalah *Hough Circles*. Fitur ini akan memberikan gambar lingkaran pada objek yang dideteksi. Sebelumnya, besar lingkaran tersebut telah disetting pada pemrograman. Lingkaran tersebut terbagi atas 2 yaitu titik pusat yang berwarna merah dan lingkaran objek yang berwarna hijau. Untuk lebih jelas bisa dilihat pada gambar 4.2.



Gambar 4.2 Fitur *Hough Circles* pada

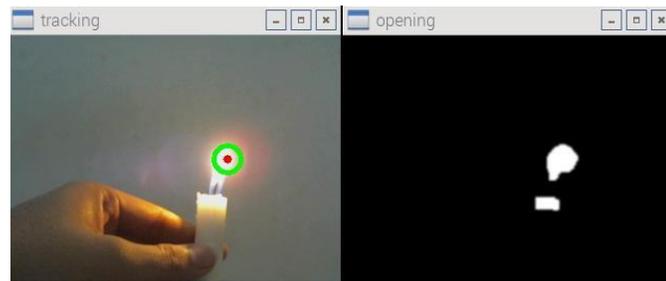
4.2 Pengujian Intensitas Cahaya

Pengujian ini dilakukan untuk melihat pengaruh intensitas cahaya terhadap *color filtering* HSV. Sebelumnya telah diberikan nilai HSV yaitu *hmin* 0, *hmax* 30, *smin* 0, *smax* 14, *vmin* 234, dan *vmax* 255. Pengujian dilakukan di dalam ruangan (*indoor*) dan pada 11.00 WIB. Untuk proses pengujian dilakukan berurutan sesuai pada tabel 4.1.

Tabel 4.1 Tahap Pengujian Intensitas Cahaya

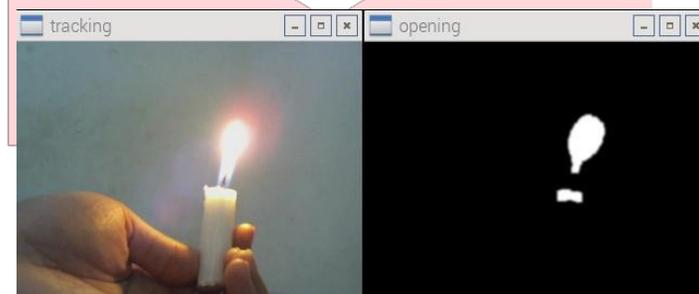
Pengujian	Pintu	Lampu
1	Tertutup	Hidup
2	Terbuka	Hidup
3	Terbuka	Mati
4	Tertutup	Mati

Pertama adalah pengujian saat pintu tertutup dan lampu menyala. Pengujian dilakukan dengan jarak 20 cm dari kamera ke objek. Hasil pengujian dapat dilihat pada gambar 4.3.



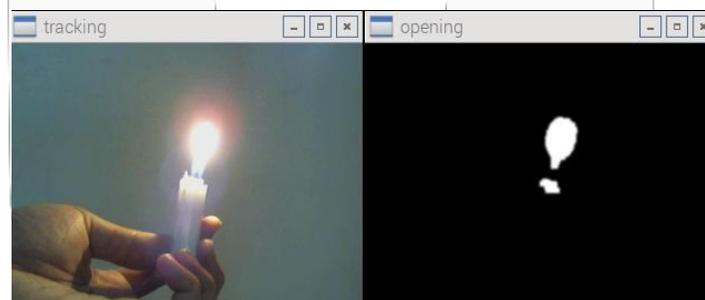
Gambar 4.3 Pengujian 1 Intensitas Cahaya

Pengujian kedua adalah kondisi saat pintu terbuka dan lampu menyala. Pengujian dilakukan dengan jarak 20cm dari kamera ke objek. Hasil pengujian dilihat pada gambar 4.4.



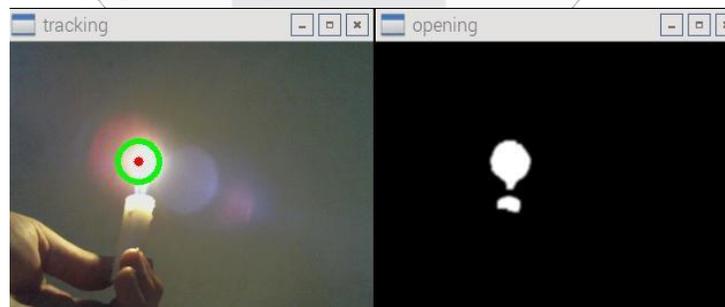
Gambar 4.4 Pengujian 2 Intensitas Cahaya

Pengujian selanjutnya adalah kondisi saat pintu terbuka dan lampu mati. Pengujian dilakukan dengan jarak 20cm dari kamera ke objek. Hasil pengujian dilihat pada gambar 4.5.



Gambar 4.5 Pengujian 3 Intensitas Cahaya

Pengujian yang terakhir adalah kondisi saat pintu tertutup dan lampu mati. Pengujian dilakukan dengan jarak 20cm dari kamera ke objek. Hasil pengujian dilihat pada gambar 4.6.



Gambar 4.6 Pengujian 4 Intensitas Cahaya

Hasil pengujian 1 sampai 4 menunjukkan bahwa ada 2 pengujian yang objeknya tidak terdeteksi dan 2 pengujian yang objeknya terdeteksi. Pengujian yang objeknya tidak terdeteksi adalah pengujian 2 dan pengujian 3. Persamaan pada kedua pengujian tersebut dilakukan dalam kondisi pintu terbuka. Hal ini bisa disimpulkan bahwa

intensitas cahaya dari luar(*outdoor*) sangat berpengaruh terhadap deteksi objek karena kedua pengujian tersebut dilakukan dalam kondisi pintu terbuka.

Pada objek yang terdeteksi terdapat pada pengujian 1 dan pengujian 4. Persamaan pada kedua pengujian tersebut adalah pintu dalam kondisi tertutup dan perbedaannya adalah lampu dalam kondisi hidup dan mati. Berdasarkan perbedaan yang telah didapat, dapat disimpulkan bahwa intensitas cahaya dalam ruangan(lampu), tidak berpengaruh dalam deteksi objek.

5. Penutup

5.1 Kesimpulan

Kesimpulan berdasarkan pengujian dan analisis sistem yang telah dilakukan pada tugas akhir ini adalah sebagai berikut:

1. Mekanik quadcopter baik dimensi maupun berat sangat berpengaruh terhadap kestabilan kontrol quadcopter, dimana saat terjadi perubahan mekanik, diperlukan autotune kembali agar quadcopter dapat bergerak dengan stabil sesuai nilai baru yang telah didapatkan.
2. Performansi terbaik kestabilan quadcopter yang telah didapat berdasarkan hasil dari autotune yang telah dilakukan adalah saat *Roll* (P sebesar 0.15, I sebesar 0.1, D sebesar 0.004, IMAX sebesar 200), saat *Pitch* (P sebesar 0.15, I sebesar 0.1, D sebesar 0.004, IMAX sebesar 200), saat *Yaw* (P sebesar 0.15, I sebesar 0.1, D sebesar 0, IMAX sebesar 100).
3. Sistem pendeteksian objek menggunakan metode HSV sudah dapat dilakukan jika nilai pada threshold objek tepat. Tetapi metode ini tidak dapat dilakukan pada kondisi yang terlalu terang atau gelap. Dapat dikatakan metode HSV ini sangat rentan terhadap intensitas cahaya.
4. Rata-rata error pada GPS sebesar 48,42 cm.
5. Komunikasi antara *flight controller* dengan Raspberry Pi 2 sudah sangat baik.
6. Penggunaan Raspberry Pi 2 sebagai sistem pengolah data gambar masih dapat dikatakan lambat dikarenakan spesifikasinya yang masih standar.
7. Sistem pemadam api pada *quadcopter* sudah dapat dikatakan akurat karena telah dapat memadamkan semua api dari total 7 kali percobaan.

Daftar Pustaka

- [1] Kementerian Kehutanan, "Statistik Kehutanan Indonesia 2011,"2012.ISBN 979-606-073-6.
- [2] Quadcopter(2014). Terdapat di: <http://www.udara.web.id/2014/quadcopter-adalah#.VkLp1tIrK00> (diakses pada tanggal 11 November 2015).
- [3] Kusuma, Whanindra, Effendi, Rusdhianto A.K. dan Iskandar, Eka. 2012. Perancangan dan Implementasi Kontrol Fuzzy-PID pada Pengendalian Auto Take-Off Quadcopter UAV. Surabaya. Institut Teknologi Sepuluh November.
- [4] *Altitude*(2014). Terdapat di: http://www.socialledge.com/sjsu/images/thumb/2/26/CmpE244_S14_Quadcopter_Quad_motion1.JPG/450px-CmpE244_S14_Quadcopter_Quad_motion1.JPG (diakses pada tanggal 11 November 2015).
- [5] Raspberry Pi 2(2015). Terdapat di: https://en.wikipedia.org/wiki/Raspberry_Pi (diakses pada 20 Juli 2016)
- [6] Putra, Darma. 2010. Pengolahan Citra Digital. Yogyakarta : ANDI
- [7] DroneKit(2015). Terdapat di: <http://python.dronekit.io/about/overview.html> (diakses pada tanggal 20 Juli 2016)
- [8] Qgroundcontrol (2010). Tersedia di: <http://qgroundcontrol.org/mavlink/start> (diakses pada tanggal 24 Juli 2016)
- [9] *Image Processing*(2015). Terdapat di: <http://dedyfitriandy.blogspot.co.id/2015/03/materi-image-processing.html> (diakses pada tanggal 22 Juli 2016)
- [10] *Morphology Operation*(2016). Terdapat di: <http://www.mathworks.com/help/images/morphological-filtering.html> (diakses pada tanggal 25 Juli 2016)