TUGAS AKHIR – EE 184801

PENERAPAN *DRONE* UNTUK PENJEJAKAN KABEL SALURAN UDARA TEGANGAN MENENGAH (SUTM) BERBASIS VISI KOMPUTER

Mar`ie Muhammad
NRP 07111640000065

Dosen Pembimbing
Ronny Mardiyanto, S.T., M.T., Ph.D.
Dr. Ir. Djoko Purwanto, M. Eng.

DEPATERMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

TUGAS AKHIR – EE 184801

PENERAPAN *DRONE* UNTUK PENJEJAKAN KABEL SALURAN UDARA TEGANGAN MENENGAH (SUTM) BERBASIS VISI KOMPUTER

Mar`ie Muhammad
NRP 07111640000065

Dosen Pembimbing
Ronny Mardiyanto, S.T., M.T., Ph.D.
Dr. Ir. Djoko Purwanto, M. Eng.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

FINAL PROJECT – **EE 184801**

DRONE APPLICATION FOR CABLE TRACKING OF SALURAN UDARA TEGANGAN MENENGAH (SUTM) *BASED ON COMPUTER VISION*

Mar`ie Muhammad
NRP 07111640000065

Supervisor(s)
Ronny Mardiyanto, S.T., M.T., Ph.D.
Dr. Ir. Djoko Purwanto, M. Eng.

ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Electrical and Intelligent Information Technology
Sepuluh Nopember Institute of Technology
Surabaya 2020

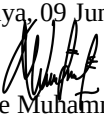
PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "**Penerapan *Drone* untuk Penjejakan Kabel Saluran Udara Tegangan Menengah (SUTM) berbasis Visi Komputer**" adalah benar-benar hasil karya yang dikerjakan secara mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 09 Juni 2020



Mar`ie Muhammad
NRP. 07111640000065

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

PENERAPAN *DRONE* UNTUK PENJEJAKAN KABEL SALURAN UDARA TEGANGAN MENENGAH (SUTM) BERBASIS VISI KOMPUTER

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik

Pada

Bidang Studi Elektronika
Departemen Teknik Elektro
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Menyetujui :

Dosen Pembimbing I

Ronny Mardiyanto, S.T., M.T., Ph.D.
NIP. 198101182003121003

SURABAYA
JUNI, 2020

LEMBAR PENGESAHAN

PENERAPAN *DRONE* UNTUK PENJEJAKAN KABEL SALURAN UDARA TEGANGAN MENENGAH (SUTM) BERBASIS VISI KOMPUTER

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik

Pada

Bidang Studi Elektronika
Departemen Teknik Elektro
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Menyetujui :

Dosen Pembimbing II

Dr. Ir. Djoko Purwanto, M.Eng.
NIP. 196512111990021002

SURABAYA
JUNI, 2020

[Halaman ini sengaja dikosongkan]

PENERAPAN *DRONE* UNTUK PENJEJAKAN KABEL SALURAN UDARA TEGANGAN MENENGAH (SUTM) BERBASIS VISI KOMPUTER

Nama : Mar`ie Muhammad
Pembimbing : 1. Ronny Mardiyanto, S.T., M.T., Ph.D.
2. Dr. Ir. Djoko Purwanto, M.Eng.

ABSTRAK

Perusahaan distribusi listrik harus melakukan inspeksi secara berkala terhadap infrastruktur dan aset saluran listriknya. Pekerjaan ini masih dilakukan secara manual dengan mengerahkan petugas inspeksi ke lapangan. Padahal di era saat ini terdapat teknologi yang tepat untuk digunakan, yaitu dengan menggunakan *drone*. Penerapan *drone* dalam inspeksi saluran listrik sebenarnya telah diterapkan akhir-akhir ini, namun pekerjaannya masih dikontrol secara manual. Untuk mengatasi keterbatasan yang ada, pada tugas akhir ini akan dibahas mengenai *drone* yang dapat melakukan penjejakan kabel SUTM secara mandiri dengan memanfaatkan visi komputer. Tugas akhir yang diusulkan terdiri dari deteksi kabel SUTM dan kontrol PID *drone* dalam melakukan penjejakan kabel SUTM dengan memanfaatkan informasi kabel SUTM yang terdeteksi. Kabel SUTM dapat diidentifikasi dengan cara mengekstrak garis vertikal yang terdapat pada *frame*. Hasil ekstraksi garis vertikal akan melewatkan informasi berupa kabel SUTM dan memblok *noise* pada latar belakang. Kemudian dilanjutkan dengan deteksi tepi menggunakan deteksi tepi Canny. Terakhir, kabel SUTM dideteksi dengan menggunakan transformasi Hough. Informasi kabel SUTM yang terdeteksi dimanfaatkan untuk memandu *drone* dalam melakukan penjejakan kabel SUTM. Skenario pengujian dilakukan dengan menerbangkan *drone* ke atas kabel SUTM secara manual dan mengarahkan posisi bagian depan *drone* sejajar dengan kabel SUTM. Percobaan dilakukan beberapa kali di dua lokasi yang berbeda secara langsung pada lingkungan yang terdapat kabel SUTM, dimana lokasi pertama terdapat sedikit garis vertikal dan lokasi kedua terdapat banyak garis vertikal. Dan hasilnya *drone* dapat melakukan penjejakan kabel SUTM secara mandiri dengan akurasi rata-rata sistem sebesar 77,40%.

Kata kunci : *Drone*, Deteksi saluran listrik, Penjejakan kabel

[Halaman ini sengaja dikosongkan]

DRONE APPLICATION FOR CABLE TRACKING OF SALURAN UDARA TEGANGAN MENENGAH (SUTM) *BASED ON COMPUTER VISION*

Name : Mar`ie Muhammad
Supervisor : Ronny Mardiyanto, S.T., M.T., Ph.D.
Dr. Ir. Djoko Purwanto, M.Eng.

ABSTRACT

Electricity distribution companies must conduct inspections of their power line infrastructures and assets periodically. This work is still done manually by deploying inspection officers to the field. Whereas in the current era, there is an appropriate technology to use, namely by using drones. The application of drones in power line inspections has actually been implemented lately, but the work is still controlled manually. To overcome the existing limitations, this final project will discuss about drones that can do SUTM cable tracking independently by utilizing computer vision. The proposed final project consists of detection of SUTM cables and PID drone control in tracking SUTM cables by utilizing detected SUTM cable information. SUTM cables can be identified by extracting vertical lines contained in the frame. The results of the vertical line extraction will pass the information of SUTM cable and block noise in the background. Then edge detection is done by using Canny edge detection. Finally, the SUTM cable is detected by using the Hough transform. The detected SUTM cable information is used to guide the drone in tracking the SUTM cable. The testing scenario is done by flying the drone onto the SUTM cable manually and directing the front of the drone in line with the SUTM cable. The experiment was carried out several times in two different locations directly in an environment with SUTM cables, where the first location contained a few vertical lines and the second location there were many vertical lines. And the result is that drones can do SUTM cables tracking independently with an average system accuracy of 77.40%.

Key Words : Drone, Power-line detection, Cable tracking

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Dengan mengucapkan puji syukur atas kehadirat Allah SWT atas segala rahmat dan hikmat yang telah diberikan-Nya, penulis dapat menyelesaikan tugas akhir dengan judul "Penerapan *Drone* untuk Penjejakan Kabel Saluran Udara Tegangan Menengah (SUTM) Berbasis Visi Komputer". Tugas akhir ini dilakukan sebagai persyaratan untuk menyelesaikan Pendidikan Strata 1 di Depatemen Teknik Elektro, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember, Surabaya.

Dalam masa penyelesaian tugas akhir ini, penulis mendapatkan banyak bantuan dan dukungan dari berbagai pihak. Maka dari itu, penulis ingin berterima kasih kepada:

1. Keluarga yang selalu memberikan dukungan dalam pengerjaan tugas akhir ini
2. Bapak Ronny Mardiyanto, S.T., M.T., Ph.D. sebagai dosen pembimbing I atas rekomendasi topik tugas akhir dan bimbingan serta arahan dalam pengerjaan tugas akhir
3. Bapak Dr. Ir. Djoko Purwanto, M.Eng. sebagai dosen pembimbing II yang telah membimbing serta memberikan arahan selama pengerjaan tugas akhir
4. Bapak Ir. Tasripan, M.T., Bapak Dr. Ir. Hendra Kusuma, M.Eng.Sc., Bapak Muhammad Attamimi, B.Eng., M.Eng., Ph.D., Bapak Ir. Harris Pringadi, M.T., sebagai dosen penguji tugas akhir yang telah memberikan saran dan pendapatnya untuk perbaikan tugas akhir ini
5. Seluruh dosen dan tenaga didik teknik elektro ITS
6. Teman-teman saya: Salik, Dion dan Rasyid yang telah membantu saya dalam proses pengerjaan tugas akhir ini
7. Rekan-rekan laboratorium A206 dan B202 bidang studi elektronika yang selalu memberikan dukungan dan saran

Tugas akhir ini masih memiliki banyak kekurangan dan memberikan peluang untuk dapat dikembangkan lebih jauh lagi. Oleh karena itu, penulis bersedia untuk menerima saran yang akan diberikan oleh pembaca. Penulis meminta maaf apabila terdapat kesalahan dalam penulisan maupun bentuk lainnya.

Surabaya, 13 Juli 2020

Mar`ie Muhammad

[Halaman ini sengaja dikosongkan]

DAFTAR ISI

HALAMAN JUDUL
PERNYATAAN KEASLIAN
LEMBAR PENGESAHAN
ABSTRAK i
ABSTRACT iii
KATA PENGANTAR v
DAFTAR ISI vii
DAFTAR GAMBAR ix
DAFTAR TABEL xii
BAB 1 PENDAHULUAN 1
1.1. Latar Belakang 1
1.2. Perumusan Masalah 2
1.3. Tujuan Penelitian 2
1.4. Batasan Masalah 2
1.5. Metodologi Penelitian 3
1.6. Sistematika Penulisan 5
1.7. Relevansi 5
BAB 2 TEORI PENUNJANG DAN TINJAUAN PUSTAKA 7
2.1. Pengertian *Drone* 7
2.2. *Mission Planner* 21
2.3. MAVLink 21
2.4. *DroneKit* 23
2.5. Raspberry Pi 4 Model B 23
2.6. Raspberry Pi *Camera Module* 24
2.7. Visi Komputer 25
2.7.1. Konversi Warna RGB menjadi *GRAY* 25
2.7.2. *Adaptive Thresholding* 26
2.7.3. *Morphological Operation* 26
2.7.4. *Structuring Elements* 27
2.7.5. *Canny Edge Detection* 27
2.7.6. *Hough line transform* 29
2.8. OpenCV 30
2.9. Saluran Udara Tegangan Menengah (SUTM) 30
2.10. Tinjauan Pustaka 31
BAB 3 PERANCANGAN SISTEM 41
3.1. Perancangan *Hardware* 44
3.2. Perancangan Visi Komputer 48

3.2.1. Perancangan Alat yang Memproses Visi Komputer 48
3.2.2. Visi Komputer 50
3.2.2.1. *Image Resize* 50
3.2.2.2. *Konversi Warna dari RGB ke Grayscale* 51
3.2.2.3. *Thresholding* 52
3.2.2.4. *Ekstrak Garis Vertikal* 52
3.2.2.5. *Deteksi Tepi* 54
3.2.2.6. *Transformasi Houghline* 54
3.3. Perancangan Kontrol *Drone* 56
BAB 4 HASIL DAN ANALISA DATA 59
4.1. Realisasi Alat 59
4.2. Pengujian Tingkat Kestabilan *Drone* 60
4.3. Pengujian Pendeteksian Kabel SUTM 63
4.3.1. Perbandingan dengan Metode Konvensional 68
4.4. Pengujian Keseluruhan Sistem 71
4.4.1. Pengujian Pengaruh Kecepatan 81
BAB 5 KESIMPULAN DAN SARAN 85
5.1. Kesimpulan 85
5.2. Saran 85
DAFTAR PUSTAKA 87
LAMPIRAN 91

DAFTAR GAMBAR

Gambar 1. 1 Blok diagram metodologi 3
Gambar 2. 1 *Flight controller* pixhawk 8
Gambar 2. 2 Pixhawk *connector* diagram[1] 9
Gambar 2. 3 *Brushless* motor RacerStar BR2212 930kV 13
Gambar 2. 4 FrSky Taranis QX7[3] 14
Gambar 2. 5 *Receiver* FrSky XM Plus[4] 14
Gambar 2. 6 Pin-pin yang terdapat pada *receiver* FrSky XM Plus 15
Gambar 2. 7 GPS Ublox Neo M8N 16
Gambar 2. 8 *Power module* 17
Gambar 2. 9 *Buzzer* yang dihubungkan dengan pixhawk 18
Gambar 2. 10 *Safety switch* yang dihubungkan dengan pixhawk 19
Gambar 2. 11 *Telemetry connector diagram* 20
Gambar 2. 12 *Pinout Telemetry* 20
Gambar 2. 13 Cara menghubungkan *telemetry* dan pixhawk 21
Gambar 2. 14 Raspberry Pi 4 model B 24
Gambar 2. 15 Pinout Raspberry Pi 4 model B 24
Gambar 2. 16 Raspberry Pi *camera module* 25
Gambar 2. 17 Bentuk *structure element* vertikal kernel 1x7 27
Gambar 2. 18 Sudut pembentuk citra garis [13] 29
Gambar 2. 19 Saluran Udara Tegangan Menengah (SUTM) 31
Gambar 3. 1 Ilustrasi rancangan sistem 41
Gambar 3. 2 Blok diagram perangkat keras sistem penjejakan kabel SUTM 42
Gambar 3. 3 Skema rangkaian elektronika sistem 43
Gambar 3. 4 Sistem *wiring* pada *drone* 47
Gambar 3. 5 Sistem *wiring* alat yang memproses visi komputer 49
Gambar 3. 6 Blok diagram visi komputer dalam mendeteksi kabel SUTM 50
Gambar 3. 7 Konversi warna dari RGB ke *Grayscale* 51
Gambar 3. 8 *Frame* hasil *thresholding* 52
Gambar 3. 9 *Structure element* garis vertikal 53
Gambar 3. 10 Hasil ekstraksi garis vertikal 53
Gambar 3. 11 Deteksi tepi Canny 54
Gambar 3. 12 Hasil deteksi garis menggunakan transformasi *Houghline* 55
Gambar 3. 13 Prinsip kerja *drone* 56
Gambar 3. 14 Pergerakan *drone* dengan menggunakan referensi *frame* 56
Gambar 3. 15 Garis kontrol pada *frame* 57

Gambar 3. 16 Blok diagram kontrol PID *drone* 58
Gambar 4. 1 Realisasi alat terlihat dari atas 59
Gambar 4. 2 Realisasi alat terlihat dari depan 59
Gambar 4. 3 Pengujian tingkat kestabilan *drone* 60
Gambar 4. 4 Ketinggian pada pengujian ke-1 tingkat kestabilan *drone* 61
Gambar 4. 5 Posisi pada pengujian ke-1 tingkat kestabilan *drone*..... 61
Gambar 4. 6 Ketinggian pada pengujian ke-2 tingkat kestabilan *drone* 62
Gambar 4. 7 Posisi pada pengujian ke-2 tingkat kestabilan *drone*..... 62
Gambar 4. 8 Lokasi pengujian 1 di Jl. Teknik Kimia ITS 64
Gambar 4. 9 Lokasi pengujian 2 di Jl. Raya Keputih Permai 64
Gambar 4. 10 Ketinggian pengujian deteksi kabel di Jl. Teknik Kimia ITS 65
Gambar 4. 11 Ketinggian pengujian deteksi kabel di Jl. Raya Keputih Permai 65
Gambar 4. 12 Hasil deteksi kabel SUTM di Jl. Teknik Kimia ITS..... 66
Gambar 4. 13 Hasil deteksi kabel SUTM di Jl. Raya Keputih Permai.. 66
Gambar 4. 14 Kesalahan deteksi pertama ketika pengujian dilakukan . 67
Gambar 4. 15 Kesalahan deteksi kedua ketika pengujian dilakukan 67
Gambar 4. 16 Blok diagram metode deteksi garis konvensional 68
Gambar 4. 17 Kesalahan deteksi pengujian ke-1 di Jl. Teknik Kimia ITS menggunakan metode konvensional 70
Gambar 4. 18 Kesalahan deteksi pengujian ke-1 di Jl. Teknik Kimia ITS menggunakan metode konvensional 70
Gambar 4. 19 Ketinggian *drone* pada pengujian ke-1 di Jl. Teknik Kimia ITS 72
Gambar 4. 20 Ketinggian *drone* pada pengujian ke-2 di Jl. Teknik Kimia ITS 72
Gambar 4. 21 Ketinggian *drone* pada pengujian ke-3 di Jl. Raya Keputih Permai 73
Gambar 4. 22 Ketinggian *drone* pada pengujian ke-4 di Jl. Raya Keputih Permai 73
Gambar 4. 23 Hasil deteksi kabel pada pengujian ke-1 di Jl. Teknik Kimia ITS 74
Gambar 4. 24 Hasil deteksi kabel pada pengujian ke-2 di Jl. Teknik Kimia ITS 74
Gambar 4. 25 Hasil deteksi kabel pada pengujian ke-3 di Jl. Raya Keputih Permai. 75
Gambar 4. 26 Hasil deteksi kabel pada pengujian ke-4 di Jl. Raya Keputih Permai. 75

Gambar 4. 27 Grafik garis kontrol terhadap *setpoint* pengujian ke-1 ... 77
Gambar 4. 28 Grafik garis kontrol terhadap *setpoint* pengujian ke-2 ... 78
Gambar 4. 29 Grafik garis kontrol terhadap *setpoint* pengujian ke-3 ... 78
Gambar 4. 30 Grafik garis kontrol terhadap *setpoint* pengujian ke-4 ... 79
Gambar 4. 31 Grafik *roll* terhadap *setpoint* pada pengujian ke-1 79
Gambar 4. 32 Grafik *roll* terhadap *setpoint* pada pengujian ke-2 80
Gambar 4. 33 Grafik *roll* terhadap *setpoint* pada pengujian ke-3 80
Gambar 4. 34 Grafik *roll* terhadap *setpoint* pada pengujian ke-4 81

DAFTAR TABEL

Tabel 2. 1 Port TELEM1, TELEM2 10
Tabel 2. 2 Port GPS..... 10
Tabel 2. 3 Port SERIAL 4/5..... 10
Tabel 2. 4 POWER..... 10
Tabel 2. 5 ADC 6.6V 11
Tabel 2. 6 ADC 3.3V 11
Tabel 2. 7 I2C 11
Tabel 2. 8 CAN 11
Tabel 2. 9 SPI..... 11
Tabel 2. 10 SWITCH 12
Tabel 2. 11 Port Spektrum/DSM..... 12
Tabel 2. 12 *Connector pinout* GPS Ublox Neo M8N 16
Tabel 2. 13 *Connector pinout power module*..... 18
Tabel 2. 14 *Connector pinout buzzer* 18
Tabel 2. 15 *Connector pinout safety switch* 19
Tabel 2. 16 Komposisi pesan MAVLink 22
Tabel 3. 1 Hubungan antara komponen-komponen dengan pixhawk ... 43
Tabel 3. 2 Komponen-komponen penyusun *drone* 44
Tabel 3. 3 Sistem *wiring* pada *drone*..... 46
Tabel 3. 4 Komponen penyusun yang memproses visi komputer..... 48
Tabel 3. 5 Sistem *wiring* alat yang memproses visi komputer..... 49
Tabel 4. 1 Spesifikasi alat 60
Tabel 4. 2 Hasil perbandingan metode yang diusulkan dengan metode konvensional 71
Tabel 4. 3 Akurasi pendeteksian kabel SUTM..... 76
Tabel 4. 4 Hasil pengujian pengaruh kecepatan di Jl. Teknik Kimia ITS (lingkungan yang sedikit garis vertikal)..... 82
Tabel 4. 5 Hasil pengujian pengaruh kecepatan di Jl. Raya Keputih Permai (lingkungan yang terdapat banyak garis vertikal) 82

BAB 1
PENDAHULUAN

Topik bahasan yang akan dikerjakan pada tugas akhir ini adalah penerapan *drone* untuk penjejakan kabel SUTM berbasis visi komputer. Sebelum dilakukan pembahasan secara mendalam metode yang akan digunakan dari judul tugas akhir ini, pada bab ini akan dijelaskan hal-hal yang mendasari pembuatan tugas akhir. Hal-hal mendasar tersebut meliputi latar belakang, perumusan masalah, tujuan penulisan, batasan masalah, metodologi penelitian, sistematika penulisan dan relevansi.

1.1. Latar Belakang

Pertumbuhan pelanggan energi listrik yang terus meningkat dewasa ini telah mendorong perusahaan distribusi listrik seperti PLN untuk melakukan pembangunan jaringan distribusi tenaga listrik dalam jumlah dan tingkat utilisasi yang semakin tinggi. Peningkatan kuantitas dan utilisasi aset distribusi tersebut membutuhkan pengelolaan yang baik pada setiap siklus hidup (*life-cycle*) aset tersebut sehingga pemanfaatan aset distribusi tersebut dapat dioptimalkan dengan performa yang baik dan biaya yang efisien. [1]

Proses penyaluran tenaga listrik 20kV yang menggunakan Saluran Udara Tegangan Menengah (SUTM) sebagai salah satu media penghantar utama mengakibatkan penerimaan energi listrik di sisi konsumen bergantung pada kondisi dan kinerja keandalan SUTM dan peralatan pendukungnya. Dari sudut pandang aset, SUTM memiliki nilai kapitalisasi yang besar dengan persebaran yang luas di seluruh daerah pelayanan PLN. Oleh sebab itu perlu dilakukan pengelolaan yang baik dan tepat terhadap SUTM dan peralatan pendukungnya sepanjang siklus hidup (*life-cycle*) aset tersebut. [2]

Perusahaan Listrik Negara atau PLN masing-masing unit melaksanakan inspeksi berupa pengawasan jaringan *feeder*-nya secara rutin setiap hari dengan mengerahkan petugas Pelayanan Teknik (YanTek) ke lapangan. Pekerjaan Pelayanan Teknik (YanTek) adalah pekerjaan yang melayani pekerjaan teknik dibidang kelistrikan, mulai dari Gardu Induk (sisi tegangan menengah/JTM) sampai dengan ke pelanggan agar keandalan dan kontinuitas pasokan tenaga listrik ke pelanggan tetap terjaga dengan baik [3]. Metode pengawasan Saluran Udara Tegangan Menengah (SUTM) sampai saat ini masih memanfaatkan tenaga manusia.

Padahal di era sekarang ini, terdapat teknologi yang tepat untuk digunakan dalam mempermudah pelaksanaan inspeksi, yakni dengan memanfaatkan *drone*. Dengan menerbangkan *drone* mengikuti jalur SUTM untuk merekam secara visual, kondisi fisik dari jalur tersebut dapat diperoleh secara aktual. *Drone* juga diunggulkan karena dapat menjangkau lokasi jaringan Saluran Udara Tegangan Menengah (SUTM) yang sulit ditempuh melalui darat, misalnya SUTM yang melewati sungai dan hutan.

Penerapan *drone* dalam inspeksi saluran listrik sebenarnya telah diterapkan akhir-akhir ini, namun pekerjaannya masih dikontrol secara manual. Untuk mengatasi keterbatasan yang ada, pada tugas akhir ini akan dibahas mengenai *drone* yang dapat menjejaki jalur kabel SUTM secara mandiri seperti halnya robot *line follower* dalam mengikuti garis melalui program komputer yang dirancang sedemikian rupa dengan memanfaatkan *library* OpenCV.

1.2. Perumusan Masalah

Berdasar pada latar belakang di atas, dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana *drone* dapat mendeteksi kabel SUTM
2. Bagaimana desain *drone* yang dapat menjejaki kabel SUTM secara mandiri dengan memanfaatkan visi komputer

1.3. Tujuan Penelitian

Penelitian pada tugas akhir ini bertujuan sebagai berikut:

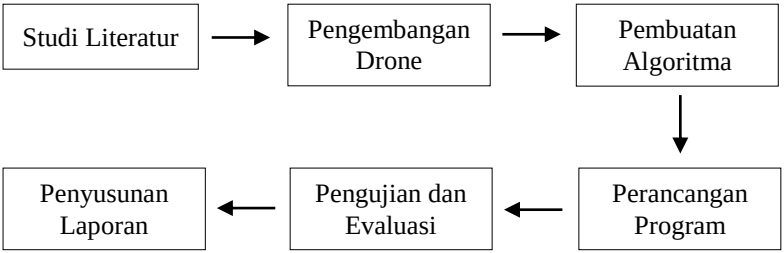
1. *Drone* dapat mendeteksi kabel SUTM
2. *Drone* yang dapat menjejaki kabel SUTM secara mandiri dengan memanfaatkan visi komputer

1.4. Batasan Masalah

Batasan masalah dari tugas akhir ini adalah:

1. Pengoperasian *drone* tidak dijalankan seluruhnya secara otomatis, seperti pada saat *take off* maupun *landing* dilakukan secara manual. Penelitian ini difokuskan hanya pada sistem penjejakan kabel SUTM
2. Pendeteksian dilakukan pada lingkungan yang terdapat kabel SUTM
3. *Drone* melakukan penjejakan pada kabel SUTM yang posisinya lurus dan kondisi kabel yang baik (tidak putus)

1.5. Metodologi Penelitian



Gambar 1. 1 Blok diagram metodologi

Gambar di atas merupakan alur dari pengerjaan tugas akhir ini. Untuk langkah-langkah yang akan dikerjakan lebih lanjut akan dijelaskan sebagai berikut:

1. Studi Literatur

   Studi literatur merupakan tahap dimana dilakukan pengumpulan dan pengkajian teori, data dan penelitian yang terjamin serta relevan untuk mendukung keabsahan tugas akhir ini. Sumber yang diambil dapat berupa jurnal, buku, paper, maupun artikel yang berasal dari badan pemerintahan atau institusi yang terpercaya. Adapun studi literatur tersebut mencakup dua topik utama, yaitu sumber dengan topik *drone* dan sistem penjejakan kabel dengan memanfaatkan visi komputer.

2. Pengembangan *drone*

   Drone akan dilengkapi oleh kamera dan minikomputer berupa raspberry pi sebagai alat yang memproses visi komputer dari gambar yang ditangkap oleh kamera. Kemudian hasil dari gambar yang telah diproses akan menjadi acuan yang menentukan perilaku dari *drone* ketika terbang menjejaki kabel Saluran Udara Tegangan Menengah (SUTM).

3. Pembuatan algoritma untuk mendeteksi dan menjejaki kabel

   Pada tahap pembuatan algoritma akan dirancang dua algoritma yang saling berkaitan satu sama lain, yaitu algoritma untuk mendeteksi sebuah objek diam dalam hal ini berupa kabel Saluran Udara Tegangan Menengah (SUTM) dan algoritma pada *drone* agar terbang mengikuti sebuah objek yang telah dideteksi dengan menggunakan data hasil visi komputer dari algoritma pertama tersebut.

4. Perancangan program

   Hasil dari kedua algoritma di atas selanjutnya akan dituliskan dalam kode program dengan menggunakan bahasa python. Program yang ditulis akan memanfaatkan *library* openCV untuk mengolah gambar dengan visi komputer dan *library dronekit* untuk mengontrol perilaku dari *drone*. Kode program tersebut akan dibuat menggunakan minikomputer berupa raspberry pi 4 model B dengan spesifikasi RAM 2 GB ROM 16 GB. Pemilihan tersebut didasari karena data yang akan diolah adalah data gambar dalam jumlah yang banyak sehingga membutuhkan memori dan kecepatan pengolahan data yang tinggi. Sehingga perilaku *drone* di udara ketika melakukan penjejakan kabel dengan memanfaatkan data yang diproses oleh minikomputer tersebut dapat berjalan dengan baik.

5. Tahap Pengujian dan Evaluasi

   Tahap selanjutnya adalah melakukan pengujian terhadap sistem yang telah dirancang. Pengujian ini akan dilakukan bertahap, yakni tanpa *drone* dan dengan *drone*. Pengujian tanpa *drone* dilakukan dengan menguji kode program yang telah dirancang dengan membuat lingkungan yang mirip seperti aslinya ketika terbang menggunakan *drone*. Pengujian dilakukan dengan memanfaatkan kamera untuk mengambil data berupa objek dan minikomputer raspberry pi untuk mengolah data tersebut dengan visi komputer. Tujuan pengujian tanpa *drone* adalah untuk mencegah terjadinya *crash* akibat perancangan sistem yang tidak sempurna. Apabila terdapat kesalahan pada rancangan, maka akan dilakukan perbaikan dan evaluasi pada rancangan. Selain itu akan dilakukan analisa untuk mengetahui penyebab terjadi suatu kesalahan yang timbul pada rancangan. Selanjutnya apabila pengujian tanpa *drone* dirasa sudah cukup sempurna, pengujian dilanjutkan dengan menggunakan *drone* tanpa memasang propeller. Pengujian dilakukan untuk mengetahui respon motor pada *drone*. Apabila respon motor sesuai dengan perancangan, maka selanjutnya dilakukan pengujian pada lingkungan sebenarnya. Setiap pengujian akan dievaluasi untuk selanjutnya dilakukan perbaikan demi kelancaran sistem.

6. Penyusunan Laporan

   Penyusunan laporan dilakukan seiring dengan tahapan lainnya. Laporan berisi proses pembuatan tugas akhir mulai dari awal pembuatan hingga akhir pembuatan tugas akhir.

1.6. Sistematika Penulisan

Penyusunan buku laporan tugas akhir disesuaikan dengan peraturan Buku Pedoman Tugas Akhir 2020 Departemen Teknik Elektro, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember. Laporan ini terdiri dari tiga bagian utama, yaitu bagian awal, bagian inti/pokok, dan bagian akhir. Penyusunan ini menggunakan sistematika sebagai berikut.

- Bagian Awal
  Bagian ini terdiri dari lembar pernyataan keaslian tugas akhir, halaman pengesahan, abstrak, kata pengantar, daftar isi, daftar gambar, daftar grafik, daftar diagram dan daftar tabel.
- Bagian Inti / Pokok
  Bagian ini memuat naskah utama dari tugas akhir diantaranya pendahuluan, kajian pustaka, metodologi penelitian, analisa hasil dan pembahasan, dan penutup.
- Bagian Akhir
  Bagian ini memuat bahan-bahan referensi yaitu daftar pustaka dan lampiran.

1.7. Relevansi

1. Sebagai bahan pertimbangan bagi PT PLN (Persero) dalam pemeliharaan jaringan Saluran Udara Tegangan Menengah (SUTM)
2. Sebagai bahan referensi untuk pengerjaan tugas akhir dengan topik serupa bagi mahasiswa lainnya.

[Halaman ini sengaja dikosongkan]

BAB 2
TEORI PENUNJANG DAN TINJAUAN PUSTAKA

Pada bab ini akan membahas beberapa teori dasar yang akan digunakan pada Tugas Akhir ini. Setiap teori dasar akan digunakan untuk menyelesaikan beberapa permasalahan yang akan dialami selama masa pengerjaan. Selain itu juga akan dilakukan tinjauan pustaka dimana akan membahas riset-riset sebelumnya yang hampir sama dengan tugas akhir ini.

2.1. Pengertian *Drone*

Drone atau *Unmanned Air Vehicle* (UAV) adalah pesawat udara dimana pada pengoperasiannya tidak membutuhkan adanya pilot di dalamnya. Peran pilot digantikan dengan suatu sistem komputer dan radio jarak jauh [4]. Awalnya UAV merupakan pesawat yang hanya dikendalikan jarak jauh, namun seiring berkembangnya zaman, sistem otomatis kini mulai banyak diterapkan. Perkembangan teknologi membuat *drone* juga mulai banyak diterapkan untuk kebutuhan sipil, terutama di bidang bisnis, industri dan logistik.

Berdasarkan jenisnya terdapat dua jenis *drone*, yaitu *multicopter* dan *fixed wing*. *Fixed wing* memiliki bentuk seperti pesawat terbang biasa yang dilengkapi sistem sayap. Tipe *fixed wing* memerlukan desain aerodinamika pada sayap dan badannya sehingga perancangannya cukup rumit. *Multicopter* yaitu jenis *drone* yang memanfaatkan putaran baling-baling untuk terbang. *Multicopter* dibagi menjadi dua yaitu *single-rotor* dan *multi-rotor*. Tipe *single-rotor* berbentuk seperti helikopter menggunakan baling-baling tunggal, sedangkan multi-rotor menggunakan 3 sampai 8 baling-baling [5]. *Multi-rotor* dapat dibedakan kembali menjadi beberapa jenis berdasarkan jumlah motor yang digunakan, yaitu *Single-main rotor*, *Tandem rotor*, *Coaxial rotor*, *Tri-rotor* dan *Quad-rotor*.

Quad-rotor atau *Quadcopter* adalah UAV yang memiliki konfigurasi 4 rotor. Komponen-komponen yang menyusun dari *drone* berjenis *quadcopter* adalah sebagai berikut.

1. *Flight Controller*

   Flight Controller adalah sirkuit *board* kecil yang berfungsi sebagai otak dari *drone*. Pada *flight controller* terdapat berbagai

komponen-komponen dan sensor yang gunanya untuk menjaga *drone* atau pesawat RC tetap seimbang dan terkendali. *Flight controller* akan menerima perintah dari operator yang akan dilanjutkan menuju motor untuk mengendalikan kecepatan motor sesuai dengan perintah operator. *Flight controller* memiliki sensor Gyro dan akselerometer yang berfungsi untuk menjaga keseimbangan *drone*. Beberapa *flight controller* yang terbaru bahkan sudah dilengkapi dengan sensor barometer, magnetometer dan GPS.

Salah satu *flight controller* yang paling sering digunakan adalah pixhawk. *Flight controller* pixhawk memiliki spesifikasi sebagai berikut.

- Prosesor
  - 32-bit ARM Cortex M4 *core with* FPU
  - 168 Mhz/256 KB RAM/2 MB *Flash*
  - 32-bit *failsafe co-processor*
- Sensor
  - MPU6000 *as main accel and gyro*
  - ST *Micro* 16-bit *gyroscope*
  - ST *Micro* 14-bit *accelerometer/compass* (*magnetometer*)
  - MEAS *barometer*
- Dimensi
  - Berat 38 g (1.3 oz)
  - Lebar 50 mm (2.0")
  - Tinggi 15.5 mm (.6")
  - Panjang 81.5 mm (3.2")[1]

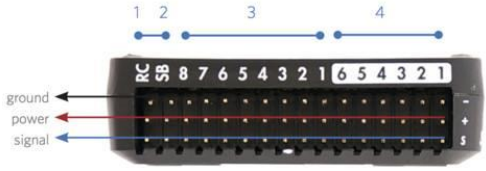
Gambar 2. 1 *Flight controller* pixhawk[1]

[1] ArduPilot Dev Team, "*Pixhawk Overview*", diakses dari https://ardupilot.org/copter/docs/common-pixhawk-overview.html, pada tanggal 01 Juli 2020, pukul 10.54 WIB.

PINOUT *flight controller* pixhawk dapat dijelaskan pada gambar 2.2 dan tabel 2.1 – tabel 2.11.

1 Spektrum DSM receiver
2 Telemetry (on-screen display)
3 Telemetry (radio telemetry)
4 USB
5 SPI (serial peripheral interface) bus
6 Power module
7 Safety switch button
8 Buzzer
9 Serial
10 GPS module
11 CAN (controller area network) bus
12 I²C splitter or compass module
13 Analog to digital converter 6.6 V
14 Analog to digital converter 3.3 V
15 LED indicator

1 Input/output reset button
2 SD card
3 Flight management reset button
4 Micro-USB port



1 Radio control receiver input
2 S.Bus output
3 Main outputs
4 Auxiliary outputs

Gambar 2. 2 Pixhawk *connector* diagram[1]

Untuk semua konektor, pin 1 berada di pin paling kiri pada gambar 2.2.

Tabel 2. 1 Port TELEM1, TELEM2[1]

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	TX (OUT)	+3.3V
3 (blk)	RX (IN)	+3.3V
4 (blk)	CTS	+3.3V
5 (blk)	RTS	+3.3V
6 (blk)	GND	GND

Tabel 2. 2 Port GPS[1]

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	TX (OUT)	+3.3V
3 (blk)	RX (IN)	+3.3V
4 (blk)	CAN2 TX	+3.3V
5 (blk)	CAN2 RX	+3.3V
6 (blk)	GND	GND

Tabel 2. 3 Port SERIAL 4/5[1]

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	TX (#4)	+3.3V
3 (blk)	RX (#4)	+3.3V
4 (blk)	TX (#5)	+3.3V
5 (blk)	RX (#5)	+3.3V
6 (blk)	GND	GND

Tabel 2. 4 POWER[1]

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	VCC	+5V
3 (blk)	CURRENT	up to +3.3V
4 (blk)	VOLTAGE	up to +3.3V
5 (blk)	GND	GND
6 (blk)	GND	GND

Tabel 2. 5 ADC 6.6V[1]

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	ADC IN	up to +6.6V
3 (blk)	GND	GND

Tabel 2. 6 ADC 3.3V[1]

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	ADC IN	up to +3.3V
3 (blk)	GND	GND
4 (blk)	ADC IN	up to +3.3V
5 (blk)	GND	GND

Tabel 2. 7 I2C[1]

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	SCL	+3.3 (pullups)
3 (blk)	SDA	+3.3 (pullups)
4 (blk)	GND	GND

Tabel 2. 8 CAN[1]

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	CAN_H	+12V
3 (blk)	CAN_L	+12V
4 (blk)	GND	GND

Tabel 2. 9 SPI[1]

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	SPI_SCK	3.3V
3 (blk)	SPI_MISO	+3.3V
4 (blk)	SPI_MOSI	+3.3V
5 (blk)	!SPI_NSS	+3.3V
6 (blk)	!GPIO	+3.3V
7 (blk)	GND	GND

Tabel 2. 10 SWITCH[1]

Pin	Signal	Volt
1 (red)	VCC	+3.3V
2 (blk)	!IO_LED_SAFETY	GND
3 (blk)	SAFETY	GND

Tabel 2. 11 Port Spektrum/DSM[1]

Pin	Signal	Volt
1 (white)	Signal	+3.3V
2 (black)	GND	GND
3 (red)	VCC	+3.3V

2. *Frame*

   Frame adalah kerangka penyusun atau bingkai dari *multirotor*, baik itu *tricopter*, *quadcopter* maupun *hexacopter*. Terdapat banyak jenis *frame* dan konfigurasi yang digunakan dalam merakit *drone*. Bahan *frame* biasanya terbuat dari plastik, serat karbon, aluminium dll. Idealnya *frame* yang dipilih untuk merakit *drone* harus keras dengan transmisi getaran seminimal mungkin. Yang perlu diperhatikan ketika memilih *frame* adalah jumlah motor yang digunakan, berat bahan *frame* dan pertimbangan desain atau bentuk *frame*.

3. Motor

   Motor merupakan penggerak utama pada *drone* untuk menghasilkan daya angkat. Motor yang telah terpasang dengan propeller dapat memberikan daya dorong pada *drone* sehingga *drone* dapat terbang. Daya dorong ini menyebabkan *drone* dapat terbang secara vertikal. Salah satu contoh motor yang digunakan pada *drone* adalah *brushless* motor RacerStar BR2212 930 kV. Motor tersebut memiliki spesifikasi sebagai berikut.

   - Nama merek: Racerstar
   - Nama barang: BR2212 *brushless* motor
   - KV: 930
   - Tegangan operasi: 2-4S
   - Berat: 52g
   - Propeller yang direkomendasikan: 1060
   - Daya angkat per-motor: 660g [2]

Gambar 2. 3 *Brushless* motor RacerStar BR2212 930kV[2]

4. *Electronic Speed Controller* (ESC)

   Electronic Speed Controller (ESC) merupakan sirkuit elektronik yang digunakan untuk mengendalikan kecepatan dari motor. Tidak hanya mengendalikan kecepatan, ESC juga dapat mengendalikan arah putaran dari motor dan pengereman motor. Selain itu, ESC juga berfungsi untuk menaikkan jumlah arus yang diperlukan oleh motor.

5. *Transmitter*

   Transmitter merupakan bagian dari sistem komunikasi *wireless*. *Transmitter* berfungsi untuk mengirimkan data ke tempat lain berupa gelombang radio. *Transmitter* pada lingkungan *drone* dapat dikatakan sebagai Radio Kontrol. Radio kontrol digunakan untuk mengendalikan pergerakan *drone* berupa *roll*, *pitch*, *throttle* dan *yaw*. Salah satu contoh dari *transmitter* adalah Taranis QX7. Taranis QX7 adalah radio kontrol yang diproduksi oleh FrSky. Berikut ini merupakan spesifikasi dari radio kontrol Taranis QX7.

   - Rentang tegangan operasi: 6~15V (2S, 3S Lipo dapat diterima)
   - Arus operasi: maksimum 210mA
   - Jumlah kanal: 16 kanal (hingga 32 kanal)
   - *Backlight* LCD *screen*: 128*64 *outdoor readable* LCD
   - Memori *flash*: 16MB [3]

---

[2] Racerstar Electronic Technology Co., Ltd., "Racerstar BR2212 930KV 2-4S *Brushless Motor For* RC *Models*", diakses dari https://www.racerstar.com/racerstar-br2212-930kv-2-4s-brushless-motor-for-rc-models-p-52.html, pada tanggal 01 Juli 2020, pukul 11.24 WIB.

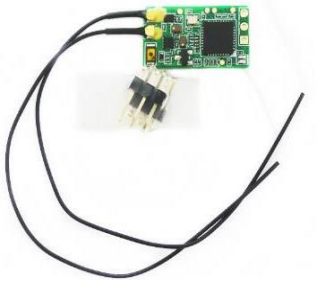
[3] FrSky Electronic Co., Ltd., "Taranis QX7", diakses dari https://www.frsky-rc.com/product/taranis-q-x7-2/, pada tanggal 01 Juli 2020, pukul 11.57 WIB.

Gambar 2. 4 FrSky Taranis QX7[3]

6. *Receiver*

Receiver merupakan bagian dari penyusun *drone* yang berfungsi untuk menerima data atau sinyal yang dikirimkan oleh *transmitter*, dalam hal ini merupakan radio kontrol. Salah satu contoh *receiver* yang sering digunakan adalah FrSky XM Plus. *Receiver* ini *compatible* dengan *transmitter* Taranis QX7 melalui mode FrSky D16. FrSky XM+ memiliki spesifikasi sebagai berikut.

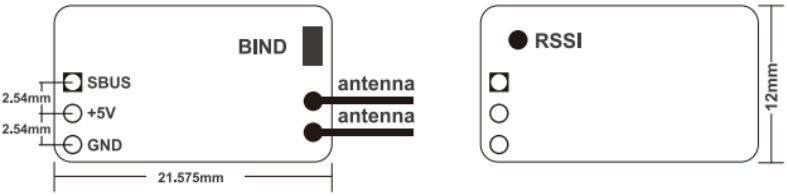
- Dimensi: 21.5*12*3.5mm(PxLxT)
- Berat: 1.6g
- Rentang tegangan operasi: 3.7V~10V
- Arus operasi: 30mA@5V
- Jumlah kanal: hingga 16CH melalui SBUS[4]



Gambar 2. 5 *Receiver* FrSky XM Plus[4]

[4] FrSky Electronic Co., Ltd., "XM PLUS", diakses dari https://www.frsky-rc.com/product/xm-plus/, pada tanggal 01 Juli 2020, pukul 12.03 WIB.

Receiver XM Plus memiliki 3 buah pin, yang masing-masing adalah SBUS, 5V dan GND. Berikut ini merupakan gambar yang menjelaskan pin-pin yang terdapat pada *receiver* XM Plus.



Gambar 2. 6 Pin-pin yang terdapat pada *receiver* FrSky XM Plus[5]

7. *Propeller*

   Propeller merupakan sebuah alat yang mengubah gerak putar menjadi daya dorong. Daya dorong ini kemudian dimanfaatkan oleh *drone* sebagai penghasil daya dorong utama. Semua *propeller* radio kontrol yang tersedia memiliki dua buah ukuran, yaitu diameter dan *pitch*. Diameter dihitung berdasarkan diameter lingkaran yang dibentuk pada saat *propeller* berputar dalam satuan *inch*. Sedangkan *pitch* merupakan jarak yang ditempuh *propeller* jika diputar dalam 1 putaran penuh dan satuan yang digunakan adalah *inch*. Semakin panjang diameter dan *pitch* dari *propeller* yang digunakan, maka semakin banyak pula udara yang disapu dan daya dorong yang dihasilkan juga semakin besar. Tapi pemilihan diameter dan *pitch* ini juga harus disesuaikan dengan motor yang digunakan. *Propeller* dengan ukuran 10x4.5" dapat dijelaskan memiliki diameter 10 *inch* dan *pitch* 4.5 *inch*. Pada *drone quadcopter*, bilah pada *propeller* berjumlah 2 bilah.

8. *Global Positioning System* (GPS)

   GPS merupakan sistem navigasi yang menggunakan satelit dengan tujuan untuk memberikan informasi posisi dari suatu objek di darat. GPS pada *quadcopter* menggunakan *receiver* untuk mengambil sinyal dari satelit untuk mendapatkan informasi posisi, kecepatan, dan waktu. GPS pada *quadcopter* dapat digunakan untuk beberapa fungsi, diantaranya *loiter*, *return to launch*, dan *auto*. *Loiter* merupakan fungsi yang digunakan oleh *quadcopter* dalam

[5] FrSky Electronic Co., Ltd., "Instruction Manual for FrSky XM+"

mempertahankan posisi dan ketinggian tertentu. *Return to launch* (RTL) merupakan fungsi dimana *quadcopter* dapat kembali ke posisi awal *take-off* secara otomatis. *Auto* merupakan fungsi *quadcopter* untuk dapat terbang secara otomatis setelah ditentukan *waypoint* yang dilalui oleh *quadcopter*. Salah satu contoh GPS yang digunakan pada *drone* adalah GPS ublox neo-m8n. GPS ini memberikan kinerja yang tinggi pada tingkat konsumsi daya yang rendah[6]. GPS ublox neo-m8n memiliki 6 buah pin yang dapat dihubungkan pada *flight controller* pixhawk, diantaranya adalah pin VCC, GND, TXD, RXD, SCL dan SDA. GPS ublox neo-m8n beserta pin-pinnya dapat dilihat pada gambar berikut.

Gambar 2. 7 GPS Ublox Neo M8N

Tabel 2. 12 *Connector pinout* GPS Ublox Neo M8N

Pin on NEO-M8N	*Color*	*Signal*	*Voltage*	*Pin on pixhawk board*
1	*Black*	GND	GND	6
2	*Red*	VCC	+5.0V	1
3	*Yellow*	UART TX	+3.3V	3
4	*Green*	UART RX	+3.3V	2
5	*White*	I2C SCL	+3.3V	4
6	*Orange*	I2C SDA	+3.3V	5

[6] U-blox, "NEO-M8 u-blox M8 concurrent GNSS modules datasheet"

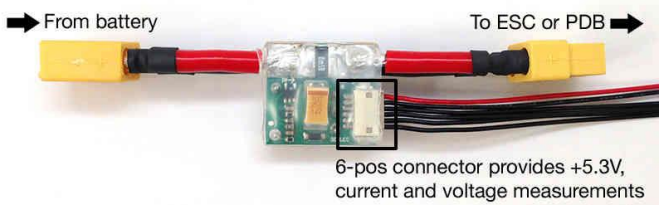
9. Baterai *Lithium Polymer* (Li-Po)

Baterai *Lithium Polymer* (Li-Po) merupakan baterai yang dapat di-*charge* ulang dan tersusun dari lapisan *polymer* yang membungkus baterai *lithium-ion*. Baterai ini dapat digunakan untuk menyuplai tegangan pada *drone*. Pada baterai Li-Po, tegangannya bergantung pada jumlah sel yang disusun secara seri. Setiap sel-nya memiliki tegangan sebesar 3,7 V. Label yang menandakan jumlah dari sel baterai Li-Po yaitu 1S, 2S, 3S, dan seterusnya. Tengangan maksimum dari satu sel baterai Li-Po adalah 4,2 V. Baterai Li-Po memiliki tegangan minimum yang dapat dikatakan sebagai tegangan *cut-off*. Tegangan *cut-off* pada baterai Li-Po yang ditetapkan dari pabrik adalah 3 V. Pada baterai LiPo terdapat nilai kapasitas yang dapat diukur dengan satuan Amp-hrs (Ah) atau milliamp-hrs (mAh). Kapasitas ini menunjukkan jumlah energi yang dapat disimpan pada baterai. Semakin besar kapasitas dari baterai makan semakin lama baterai tersebut akan bertahan.

10. *Power Module*

Power module berfungsi untuk menurunkan tegangan dari baterai LiPo sesuai dengan tegangan kerja *board flight controller*. *Power module* mendukung pengukuran tegangan baterai dan arus. Beberapa spesifikasi *power module* diantaranya sebagai berikut.

- Tegangan *input* maksimum 18V (4S LiPo)
- Mampu mengukur arus hingga 60 A
- Menyediakan catu daya 5.37V dan 2.25 A ke *board Flight Controller*



Gambar 2. 8 *Power module*[7]

[7] ArduPilot DevTeam,"*Common Power Module*", diakses dari https://ardupilot.org/copter/docs/common-3dr-power-module.html, pada tanggal 01 Juli 2020, pukul 12.24 WIB.

Tabel 2. 13 *Connector pinout power module*

Pin	*Color*	*Signal*	*Voltage*
1	*Red*	VCC	+5.3V
2	*Black*	VCC	+5.3V
3	*Black*	CURRENT SENSOR INPUT	+3.3V
4	*Black*	VOLTAGE SENSOR INPUT	+3.3V
5	*Black*	GND	GND
6	*Black*	GND	GND

11. *Buzzer*

Buzzer dapat digunakan untuk mengindikasikan perubahan status pada kendaraan. *Buzzer* dapat memainkan berbagai suara, termasuk *Arming buzz* dan *Lost Vehicle Alarms.* Menghubungkan *buzzer* dengan *flight controller* juga cukup mudah, karena hanya tinggal menghubungkan *buzzer* dengan pin VCC dan GND yang terdapat pada *flight controller* melalui port *buzzer* yang telah disediakan.

Gambar 2. 9 *Buzzer* yang dihubungkan dengan pixhawk[8]

Tabel 2. 14 *Connector pinout buzzer*

Pin	*Color*	*Signal*	*Voltage*
1	*Red*	VCC	+5.0V
2	*Black*	GND	GND

12. *Safety Switch*

Safety Switch dapat digunakan untuk meng-*enable* atau men-*disable output* ke motor dan servo. Pada *safety switch* terdapat LED

[8]ArduPilot DevTeam,"*Buzzer (aka ToneAlarm)*", diakses dari https://ardupilot.org/copter/docs/common-buzzer.html, pada tanggal 01 Juli 2020, pukul 12.30 WIB.

yang mengindikasikan keadaan dari sistem. Apabila LED berkedip secara konstan, maka mengindikasikan inisialisasi sistem. Apabila LED berkedip sesekali, maka mengindikasikan sistem sudah siap dan jika *safety switch* ditekan maka akan meng-*enable output* ke motor. Apabila LED berwarna *solid*, maka mengindikasikan bahwa *safety switch* telah ditekan.

Pada *safety switch* terdapat 3 pin yang terhubung dengan *flight controller* pixhawk, diantaranya adalah VCC, LED dan GND. Masing-masing pin ini dihubungkan ke pixhawk melalui port *safety switch* yang telah disediakan. Berikut ini merupakan *safety switch* yang telah dihubungkan dengan pixhawk.

Gambar 2. 10 *Safety switch* yang dihubungkan dengan pixhawk [9]

Tabel 2. 15 *Connector pinout safety switch*

Pin	*Color*	*Signal*	*Voltage*
1	*Red*	VCC	+3.3V
2	*Black*	SWITCH LED OUTPUT	+3.3V
3	*Black*	SWITCH INPUT	+3.3V

13. *Telemetry*

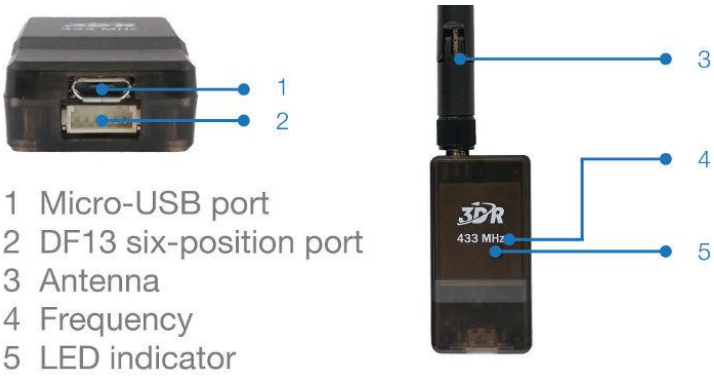
Telemetry merupakan komponen opsional yang dapat digunakan pada *drone* untuk menyediakan koneksi MAVLink nirkabel antara *Ground Control Station* (GCS) dengan *drone.* Hal ini memungkinkan *user* untuk mengetahui status *drone* ketika sedang berada di udara, seperti kecepatan *drone* saat ini, ketinggian *drone* saat ini, mode penerbangan *drone* saat ini dan lain-lain. *Telemetry* yang ada di

[9] ArduPilot Dev Team, "*Safety Switch*", diakses dari https://ardupilot.org/copter/docs/common-safety-switch-pixhawk.html, pada tanggal 01 Juli 2020, pukul 12.35 WIB

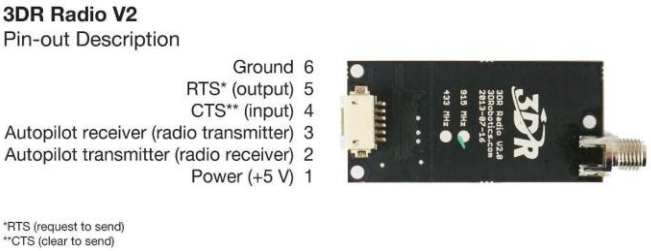
pasaran dapat berupa 915Mhz atau 433Mhz. *Telemetry* memiliki 2 warna LED yang menandakan status, satu berwarna merah dan satunya lagi berwarna hijau. Arti status dari perbedaan warna LED ini adalah:

- LED hijau berkedip-mencari pasangan *telemetry*
- LED hijau *solid- telemetry* terkoneksi dengan pasangannya
- LED merah bekedip-mentransmisikan data
- LED merah *solid*-dalam mode pembaruan *firmaware*

Telemetry memiliki modul yang digunakan pada darat dan udara. Modul pada darat, *telemetry* dihubungkan dengan GCS seperti laptop, tablet dan lain-lain. Modul pada udara, *telemetry* dihubungkan dengan *drone*. Berikut ini ditampilkan *telemetry* beserta *pinout*-nya

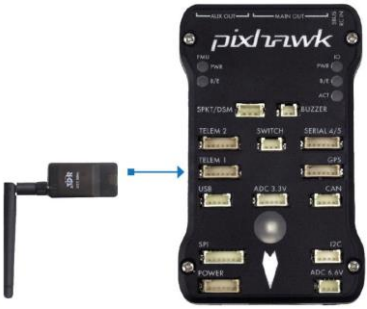


Gambar 2. 11 *Telemetry connector diagram*



Gambar 2. 12 *Pinout Telemetry*

Cara menghubungkan *telemetry* dengan *flight controller* pixhawk adalah dengan menghubungkannya melalui *port* TELEM1 atau TELEM2 yang telah tersedia pada pixhawk. Berikut ini adalah gambar yang menjelaskan cara menghubungkan *telemetry* dengan pixhawk.



Gambar 2. 13 Cara menghubungkan *telemetry* dan pixhawk[10]

2.2. *Mission Planner*

Mission planner adalah salah satu *software user interface* yang digunakan sebagai *ground control station* (GCS) untuk sebuah *drone*/UAV, *copter* dan *rover*. *Software* ini hanya cocok dijalankan untuk sistem operasi windows saja. *Software Mission planner* merupakan salah satu *software* yang dirancang untuk mempermudah *user* dalam perencanaan terbang otomatis dan biasanya digunakan untuk pemetaan. Sehingga perhitungan manual tentang skala, resolusi spasial, ketinggian, nilai *overlap* dan *sidelap*, dan lainnya bisa di dapatkan hasil perhitungan secara otomatis. Beberapa hal yang bisa dilakukan oleh *Mission Planner* antara lain seperti memuat *firmware* ke dalam *board* pixhawk, mengatur perilaku dari UAV sesuai dengan keinginan pengguna, menjalankan misi *autonomous*, dll.

2.3. MAVLink

MAVLink adalah protokol komunikasi yang bersifat *open source*. Protokol Ini dikembangkan pada tahun 2009 oleh Lorenz Meier. MAVLink merupakan salah satu protokol yang paling banyak digunakan dalam industri UAV, karena menyediakan komunikasi ringan yang

[10] ArduPilot Dev Team, "*SiK Telemetry Radio*", diakses dari https://ardupilot.org/copter/docs/common-sik-telemetry-radio.html, pada tanggal 01 Juli 2020, pukul 12.39 WIB

sederhana. Pesan atau yang paling sering disebut dengan paket, sudah mempunyai format yang telah ditentukan sebelumnya agar memiliki konsistensi antar sistem. Protokol ini bertindak dengan penyiaran paket sehingga memungkinkan penerimaan ke wahana UAV atau komponen [6].

Paket MAVLink terdiri dari *header*, *payload* dan *Cyclic Redundancy Check* (CRC). *Header* berisi informasi penting untuk identifikasi sistem yang menerima paket. Oleh karena itu, *header* disusun oleh *start sign*, *packet length*, *sequence number*, *system* ID, *component* ID dan ID dari pesan yang masuk. *Payload* berisi data yang berpura-pura untuk dikirim. *Payload* bervariasi, tergantung pada jenis pesannya. Misalnya *heartbeat*, kontrol *waypoint*, pemeriksaan parameter, dan lain-lain. CRC adalah kode pendeteksi kesalahan, digunakan untuk mengkonfirmasi integritas pesan. Informasi di atas dapat dipaparkan pada tabel 2.16 [7].

Tabel 2. 16 Komposisi pesan MAVLink[11]

Byte	Parameter	Penjelasan
0	*Start Sign*	Mengumumkan mulainya paket
1	*Payload Length*	Mengindikasikan panjang dari *payload*
2	*Sequence Number*	Digunakan untuk mendeteksi paket yang hilang
3	*System* ID	ID sistem pengiriman, untuk membedakan MAV yang berbeda
4	*Component* ID	ID komponen pengiriman, misalnya *autopilot*
5	*Message* ID	Mengidentifikasi tipe pesan
6 hingga (n+6)	Data	Data sebenarnya dari pesan, bergantung pada bidang di atas
(n+7) hingga (n+8)	CRC	Mendeteksi error pada paket

MAVLink memberikan banyak keuntungan pada *user* ketika digunakan. Salah satunya, seperti yang telah disebutkan sebelumnya bahwa protokol ini sederhana dan menyediakan komunikasi yang ringan

[11] Alexandre Valério Rodrigues, dkk, "*Unmanned Systems Interoperability in Military Maritime Operations: MAVLink to STANAG 4586 Bridge*", dalam *OCEANS 2017 - Aberdeen*, Aberdeen, United Kingdom, Jun 2017, hlm. 1–5, doi: 10.1109/OCEANSE.2017.8084866.

sebagaimana protokol ini berfokus pada kecepatan transmisi dan deteksi kesalahan. Keuntungan lainnya adalah bahwa protokol ini berlisensi gratis dan memiliki akses yang mudah ke *library* pesan.

Terlepas dari semua kelebihan yang telah disebutkan di atas, MAVLink juga memiliki beberapa kekurangan terutama di bidang militer. Sebagian besar misi angkatan bersenjata memiliki tingkat kerumitan yang cukup tinggi, sehingga UAV yang beroperasi pada lingkungan ini seharusnya tidak memiliki standar komunikasi yang sederhana [6]. Kerugian lain yang bisa disebutkan dari protokol ini adalah rentan terhadap peretasan. Tesis yang dibuat oleh Marty menunjukkan bahwa rentannya protokol MAVLink [8].

Dari penjelasan di atas dapat disimpulkan bahwa protokol MAVLink sangat baik dalam hal kecepatan komunikasi dan mendeteksi *error,* sehingga menjadi pilihan yang sangat baik untuk keperluan sipil. Namun jika digunakan untuk komunikasi militer perlu memiliki doktrin khusus dan tingkat keamanan yang tinggi [6].

2.4. *DroneKit*

DroneKit adalah sebuah *library* perangkat lunak yang bertujuan untuk memudahkan pengguna dalam memberikan perintah kepada wahana UAV atau *drone*. *DroneKit* bersifat *open source* yang berarti siapa saja bisa menggunakan *library* ini tanpa berbayar. Pada umumnya *droneKit* mengimplementasikan bahasa *python* yang biasa disebut *DroneKit-Python* (dulunya *Drone*API-*Python*). *DroneKit-Python* mampu berjalan di komputer *on-board* dan memungkinkan pengguna untuk menciptakan komunikasi antara komputer *on-board* UAV dan ArduPilot *flight controller*, semuanya menggunakan tautan latensi rendah. Pemrosesan *on-board* meningkatkan efisiensi autopilot secara signifikan, meningkatkan perilaku dari wahana UAV dan melakukan tugas-tugas yang intensif secara komputasi atau sensitif waktu, seperti visi komputer dan algoritma estimasi atau bahkan perencanaan jalur [9]. Dengan menggunakan *dronekit*, kita dapat mengontrol *drone* dan membuat misi *autonomous* melalui *script* yang telah dirancang.

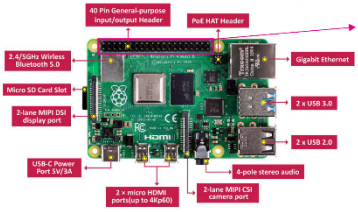
2.5. Raspberry Pi 4 Model B

Raspberry Pi 4 model B merupakan sebuah minikomputer yang dijalankan dengan menggunakan 64-bit *quadprocessor* 1.5GHz. Raspberry pi 4 model B dilengkapi dengan pilihan 1GB/2GB/4GB LPDDR4-3200 SDRAM, 2.4 GHz dan 5.0 GHz IEEE 802.11ac *wireless*,

Bluetooth 5.0, BLE, *Gigabit Ethernet*, 40 pin GPIO, 2 *port micro*-HDMI, 2 *port* USB 3.0, 2 *port* USB 2.0, *port* kamera dan *port micro*-SD. Raspberry Pi banyak diaplikasikan dalam beberapa proyek dimulai dari IoT (*Internet of Things*), robotik, otomasi rumah dan otomasi industri.

Gambar 2. 14 Raspberry Pi 4 model B

Berikut ini ditampilkan pinout dari Raspberry Pi 4 model B



Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1, I²C)		DC Power 5v	04
05	GPIO03 (SCL1, I²C)		Ground	06
07	GPIO04 (GPCLK0)		(TXD0, UART) GPIO14	08
09	Ground		(RXD0, UART) GPIO15	10
11	GPIO17		(PWM0) GPIO18	12
13	GPIO27		Ground	14
15	GPIO22		GPIO23	16
17	3.3v DC Power		GPIO24	18
19	GPIO10 (SPI0_MOSI)		Ground	20
21	GPIO09 (SPI0_MISO)		GPIO25	22
23	GPIO11 (SPI0_CLK)		(SPI0_CE0_N) GPIO08	24
25	Ground		(SPI0_CE1_N) GPIO07	26
27	GPIO00 (SDA0, I²C)		(SCL0, I²C) GPIO01	28
29	GPIO05		Ground	30
31	GPIO06		(PWM0) GPIO12	32
33	GPIO13 (PWM1)		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40

Gambar 2. 15 Pinout Raspberry Pi 4 model B [12]

2.6. Raspberry Pi *Camera Module*

Raspberry Pi *Camera Module* merupakan modul kamera yang disediakan oleh Raspberry Pi. Modul ini dihubungkan langsung ke konektor CSI pada Raspberry Pi. Kamera ini mampu menghasilkan gambar yang jernih dengan resolusi kamera 5MP, dan mampu merekam video hingga 1080p HD pada 30 fps. Modul ini terpasang pada Raspberry Pi dengan menggunakan kabel pita 15 pin ke 15 pin MIPI *Camera Serial Interface* (CSI) pada raspberry pi yang didesain khusus untuk modul

[12] Berdasarkan *user manual* Raspberry Pi

kamera. CSI bus mampu mentransfer data dengan kecepatan tinggi dan secara eksklusif membawa data piksel ke prosesor BCM2835. Modul kamera ini sendiri berukuran kecil, sekitar 25mm x 24mm x 9mm dan beratnya sekitar 3g. Modul kamera ini dapat digunakan pada seluruh model Raspberry Pi mulai dari 1, 2, 3 dan 4. Spesifikasi dari modul kamera adalah sebagai berikut.

1. Sensor : *Omnivision* OV5647
2. Piksel : 5 MP
3. Resolusi sensor : 2592 x 1944 piksel
4. Video : 1080p30, 720p60 dan 640x480p60/90
5. Besar CCD : ¼ inch
6. Panjang *Focal* : 3,6 mm *Adjustable*
7. Sudut Diagonal : 60 derajat
8. Dimensi : 25 mm x 24 mm x 9mm

Gambar 2. 16 Raspberry Pi *camera module*

2.7. Visi Komputer

Visi komputer adalah transformasi data yang diperoleh dari kamera dalam bentuk video/foto menjadi representasi baru [10]. Semua transformasi yang dilakukan bertujuan untuk mencapai tujuan tertentu. Representasi baru yang dimaksud mungkin berarti mengubah warna gambar yang semula dalam format RGB menjadi *grayscale* atau menerapkan filter-filter pada gambar seperti *thresholding* dan lain-lain. Untuk setiap video/foto yang tertangkap oleh kamera, maka akan menghasilkan sebuah gambar atau biasa dikenal dengan kata *frame*. Setiap *frame* ini memiliki deretan nilai-nilai piksel yang mewakili nilai warna yang ada pada *frame*. Kemudian, nilai-nilai pada piksel ini akan diproses sehingga dapat diambil data/informasi yang diinginkan.

2.7.1. Konversi Warna RGB menjadi *GRAY*

Warna RGB (*Red,Green,Blue*) merupakan warna yang terdiri dari 3 kanal warna, dimana masing-masing warna memiliki nilai piksel tersendiri pada suatu gambar. Namun untuk warna *GRAY* hanya memiliki

1 kanal warna. Untuk melakukan konversi dari RGB menjadi *GRAY* adalah menggunakan rumus sebagai berikut [10].

$$Gray = 0{,}299 \cdot R + 0{,}587 \cdot G + 0{,}114 \cdot B \quad (2.1)$$

2.7.2. *Adaptive Thresholding*

Thresholding adalah teknik penting dalam segmentasi gambar yang mencoba untuk mengidentifikasi dan mengekstrak objek dari latar belakang. Dibandingkan dengan *thresholding* konvensional yang menggunakan nilai global untuk semua piksel, *Adaptive Thresholding* merupakan metode di mana nilai *threshold* dihitung untuk daerah yang lebih kecil. Oleh karena itu, akan ada nilai *threshold* yang berbeda untuk daerah yang berbeda. *Adaptive Thresholding* biasanya menggunakan gambar *grayscale* atau gambar berwarna sebagai input dan dalam implementasi yang sederhana, *output* yang berupa citra biner akan mewakili segmentasi objek. Nilai *threshold* harus dihitung untuk setiap piksel dalam gambar. Jika nilai piksel di bawah nilai *threshold*, maka nilai tersebut ditetapkan sama dengan nilai latar belakang, jika tidak maka diasumsikan nilainya sama dengan nilai latar depan.

Terdapat 2 metode pada *Adaptive Thresholding*, diantaranya yaitu *Adaptive Thresholding Mean* dan *Adaptive Thresholding Gaussian*. Pada *Adaptive Thresholding Mean,* nilai *threshold*-nya adalah rata-rata dari area sekitar. Sedangkan pada *Adaptive Thresholding Gaussian,* nilai *threshold*-nya adalah jumlah *weighted* pada nilai area sekitar, dimana *weighted* adalah *Gaussian Window.* Hasil yang didapatkan dari penggunaan *threshold* adalah nilai biner berupa 1 dan 0. [10]

2.7.3. *Morphological Operation*

Morphologi adalah pengolahan citra yang memproses gambar berdasarkan *structuring element* yang telah ditentukan sebelumnya yang juga dikenal sebagai kernel. Dengan memilih ukuran dan bentuk kernel, dapat dibuat operasi morphologi yang peka terhadap bentuk-bentuk tertentu yang ingin diekstrak pada gambar input. Dua operasi morphologi yang paling dasar adalah *dilate* dan *erode*.

Dilate merupakan konvolusi gambar dengan kernel dimana nilai apapun yang diberikan akan diganti dengan nilai maksimum lokal yang terdapat pada dalam kernel tersebut. Hasil akhir dari *dilate* adalah perluasan pada area *contour* [9]. Berikut merupakan rumus dari *dilate.*

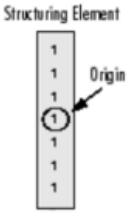
$$dilate\ (x,y) = \max_{(i,j) \in kernel} src(x + i, y + j) \quad (2.2)$$

Sedangkan *Erode* merupakan konvolusi gambar dengan kernel dimana nilai apapun yang diberikan akan diganti dengan nilai minimum lokal dari dalam kernel tersebut. Hasil akhir dari *erode* adalah pengecilan pada area yang diperoleh dari segmentasi gambar hasil *thresholding* [9]. Berikut merupakan rumus dari *erode*.

$$erode\ (x,y) = \min_{(i,j)\in kernel} src(x + i, y + j) \quad (2.3)$$

2.7.4. *Structuring Elements*

Secara umum dalam setiap operasi morphologi, *structuring element* merupakan bagian yang paling penting digunakan untuk menyelidiki gambar input. *Structuring Element* adalah matriks yang hanya terdiri dari 0 dan 1 yang dapat memiliki bentuk dan ukuran yang berbeda-beda. Biasanya bentuk dan ukurannya jauh lebih kecil dibandingkan dengan gambar yang sedang diproses. Piksel dengan nilai 1 menentukan area. Piksel tengah pada *structuring element* atau yang biasa disebut dengan *origin*, mengidentifikasi piksel yang sedang diproses.



Gambar 2. 17 Bentuk *structure element* vertikal kernel 1x7

Structuring element dapat memiliki banyak bentuk umum, seperti garis, *diamond*, *disk*, garis periodik, dan lingkaran. Biasanya *structuring element* yang dipilih memiliki ukuran dan bentuk yang sama dengan objek yang ingin diproses/ekstrak dalam gambar input. Misalnya, untuk menemukan garis dalam gambar, maka dibuat *structuring element* linier.

2.7.5. *Canny Edge Detection*

Canny Edge detector dikembangkan oleh John F. Canny pada tahun 1986 dan menggunakan algoritma multi-tahap untuk mendeteksi berbagai tepi dalam gambar. Tujuan mendeteksi tepi adalah untuk mengelompokkan objek-objek dalam citra, dan juga digunakan untuk menganalisis citra lebih lanjut [11]. Terdapat beberapa tahap yang

dilakukan untuk mengimplementasikan *canny edge detector* dalam mendeteksi tepi. Langkah – langkah tersebut adalah sebagai berikut.

1. *Noise Reduction*

Tidak dapat dipungkiri bahwa semua gambar yang diambil dari kamera akan berisi sejumlah *noise*. Untuk mencegah *noise* salah dideteksi sebagai tepian, maka *noise* harus dikurangi. Cara untuk mengurangi *noise* adalah dengan kernel 5x5 Gaussian Filter ditunjukkan pada persamaan berikut

$$K = \frac{1}{159}\begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \quad (2.4)$$

2. *Finding Gradien*

Algoritma Canny pada dasarnya menemukan titik tepi pada gambar grayscale dengan perubahan nilai intensitas yang paling besar, daerah ini ditemukan dengan menentukan gradien gambar. Gradien pada setiap piksel gambar yang telah diperhalus ditentukan dengan menerapkan operator Sobel. Langkah ini akan memperkirakan gradien pada arah x dan y. Hal ini ditunjukkan dalam persamaan berikut.

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad (2.5) \qquad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} \quad (2.6)$$

Magnitudo gradien (juga dikenal sebagai kekuatan tepi) yang diukur dengan menerapkan hukum Pythagoras dan arah tepian dapat ditunjukkan dalam persamaan berikut.

$$G = \sqrt{Gx^2 + Gy^2} \quad (2.7)$$

$$\theta = \arctan\left(\frac{Gy}{Gx}\right) \quad (2.8)$$

Dimana Gx dan Gy adalah gradien pada arah x dan y.

3. *Non-maksimum-suppression*

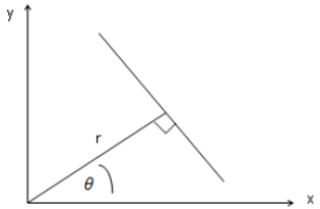
Bertujuan untuk menghilangkan piksel yang tidak diinginkan yang mungkin bukan bagian dari tepi. Sehingga pada akhirnya didapatkan garis tepi yang lebih ramping.

4. *Hysteresis*

Merupakan Langkah terakhir. Canny menggunakan dua *threshold* (*upper* dan *lower*). Jika nilai gradien piksel lebih besar daripada *upper threshold* maka piksel tersebut diterima sebagai tepi. Jika nilai gradien piksel lebih kecil daripada *lower threshold* maka piksel tersebut tidak dianggap sebagai tepi, Jika gradien piksel berada diantara dua *threshold*, maka yang hanya akan diterima jika terhubung dengan piksel yang berada di atas *upper threshold*. [12]

2.7.6. *Hough line transform*

Hough line transform adalah teknik transformasi citra yang dapat digunakan untuk mengisolasi atau dengan kata lain memperoleh fitur dari sebuah citra. Karena tujuan dari sebuah transformasi adalah mendapatkan suatu fitur yang lebih spesifik.



Gambar 2. 18 Sudut pembentuk citra garis [13]

Hough line transform merupakan teknik yang paling umum digunakan untuk mendeteksi objek berbentuk kurva seperti garis, lingkaran, elips dan parabola. Keuntungan utama dari transformasi Hough adalah dapat mendeteksi sebuah tepian dengan celah pada batas fitur dan secara relatif tidak dipengaruhi oleh noise. Transformasi Hough memiliki beberapa perbedaan rumus yang diterapkan. Semuanya tergantung pada jenis objek yang dicari, misalnya untuk mencari objek garis digunakan fungsi garis seperti berikut ini: [13]

$$x \cos\theta + y \sin\theta = r \tag{2.9}$$

Dengan x dan y merupakan titik koordinat yang menyusun objek garis tersebut, sedangkan teta adalah sudut yang dibentuk antara objek garis dengan sumbu x, dan r merupakan jarak antara garis dengan titik pusat (0,0). Dalam proses transformasi Hough, meliputi tiga bagian dasar. Bagian yang pertama adalah konversi warna dari RGB ke *gray*. Kemudian diterapkan deteksi tepi. Tujuan deteksi tepi adalah untuk menurunkan jumlah titik dalam pencarian ruang bagi objek. Ketika titik tepi ditemukan oleh detektor tepi, algoritma transformasi Hough dikerjakan hanya pada titik tersebut. Untuk deteksi tepi, digunakan detektor tepi Canny, Roberts Cross, atau Sobel yang tujuannya memaksimalkan sinyal pada rasio derau dan lokalisasi serta meminimalisasi kesalahan pada deteksi tepi. [14]

2.8. OpenCV

OpenCV adalah sebuah *library computer vision* dan *machine learning* yang bersifat *open source*. Pada tahun 1999, Gary Bradski yang bekerja pada perusahaan Intel meluncurkan OpenCV dengan harapan dapat mempercepat kinerja manusia pada bidang *computer vision* dan *artificial intelligence* dengan menyediakan infrastruktur yang solid untuk semuanya yang bekerja pada bidang tersebut [10]. OpenCV dapat digunakan dalam Bahasa pemrograman C, C++, Python, Java dan sebagainya. OpenCV dapat bekerja pada sistem operasi Windows, Linux, Android, iOS maupun Mac OS [15].

OpenCV memiliki lebih dari 2500 algoritma. Algoritma ini dapat digunakan untuk mendeteksi dan mengenali wajah, mengidentifikasi objek, mengklasifikasikan tindakan manusia dalam video, melacak objek yang bergerak, mengekstrak model objek 3D, menemukan gambar serupa dari basis data gambar, menghilangkan *red eyes* dari gambar yang diambil menggunakan *flash*, mengikuti gerakan mata, mengenali pemandangan dan lain-lain. Library ini digunakan secara luas oleh bidang akademisi, perusahaan, grup penelitian, badan pemerintahan dan lain-lain [16].

2.9. Saluran Udara Tegangan Menengah (SUTM)

Saluran Udara Tegangan Menengah (SUTM) adalah sebagai konstruksi termurah untuk penyaluran tenaga listrik pada daya yang sama. Standar tegangan menengah sebagai tegangan operasi yang digunakan di Indonesia adalah 20 kV. Konstruksi ini terbanyak digunakan untuk konsumen Jaringan Tegangan Menengah (JTM) yang digunakan di Indonesia. Ciri utama jaringan ini adalah penggunaan penghantar telanjang yang ditopang dengan isolator pada tiang besi/beton.

Penggunaan penghantar telanjang, dengan sendirinya harus diperhatikan faktor yang terkait dengan keselamatan ketenagalistrikan seperti jarak aman minimum yang harus dipenuhi penghantar bertegangan 20 kV tersebut antar Fase atau dengan bangunan atau dengan tanaman atau dengan jangkauan manusia. Termasuk dalam kelompok yang diklasifikasikan SUTM adalah juga bila penghantar yang digunakan adalah penghantar berisolasi setengah AAAC-S (*half insulated single core*). Penggunaan penghantar ini tidak menjamin keamanan terhadap tegangan sentuh yang dipersyaratkan akan tetapi untuk mengurangi resiko gangguan temporer khususnya akibat sentuhan tanaman [17].

Gambar 2. 19 Saluran Udara Tegangan Menengah (SUTM)

2.10. Tinjauan Pustaka

Berikut ini merupakan penelitian-penelitian yang pernah dilakukan terkait dengan topik tugas akhir yang diusulkan.

1. *A Power Line Detection Algorithm To Support A Fine Grain UAV Movement Guidance* [18]

   Paper yang ditulis oleh Italo De Ávila Wieczorek berfokus pada pengembangan sebuah algoritma kontrol UAV dalam memandu UAV mengikuti saluran listrik secara *autonomous*. Algoritma yang diusulkan pada paper ini didasarkan pada empat tahap: Pengambilan gambar, penyaringan gambar, deteksi garis dan kontrol penerbangan

dengan PID. Deteksi garis dilakukan dengan menggunakan transformasi Hough. Dari saluran listrik yang terdeteksi didapatkan posisi saluran listrik yang selanjutnya dimanfaatkan untuk navigasi *drone* dengan kontrol PID. Hasil dari penelitian ini mampu mendeteksi saluran listrik dengan baik dan *drone* dapat mengikuti saluran listrik secara *autonomous* dengan kontrol PID menggunakan seluruh data yang telah diperoleh sebelumnya.

2. *Drone based Power Line Tracking System* [19]

   Paper ini membahas tentang algoritma deteksi saluran listrik secara r*eal-time* dan memberikan kontrol kepada UAV berdasarkan informasi dari data yang ada. Untuk mendeteksi saluran listrik secara real-time, Region of Interest (ROI) diterapkan. Kemudian, algoritma *clustering k-means* digunakan untuk membedakan saluran listrik dari latar belakang. Terakhir, saluran listrik dideteksi dengan menggunakan transformasi Hough, dan posisi tengah dan sudut kemiringan diperkirakan dengan menggunakan filter Kalman untuk mengontrol *drone* dengan halus. Kontrol *drone* didesain berdasarkan metode kontrol proporsional-derivatif (PD). Interaksi antara kontrol posisi dan kontrol *attitude* membuat *drone* dapat mengikuti saluran listrik. Percobaan dilakukan pada lingkungan yang kondisinya mirip dengan kondisi aktual. Hasilnya, *drone* dapat megikuti saluran listrik dengan baik. Namun metode yang diusulkan masih memiliki batasan karena ketika deteksi saluran listrik gagal, sulit untuk dilakukan kontrol *drone* secara mandiri. Strategi dibuat jika deteksi saluran listrik gagal, yaitu dengan menaikkan ketinggian *drone*, melakukan pendeteksian ulang saluran listrik, atau kembali ke titik awal.

3. *Wire Recognition in Image Within Aerial Inspection Application* [20]

   Paper ini membahas tentang pendeteksian saluran listrik secara otomatis dari gambar yang diperoleh oleh UAV. Metode utama yang diusulkan pada paper ini adalah dengan mengetahui posisi saluran listrik untuk memperkirakan orientasi kabel dalam gambar, dengan menggunakan fungsi *elliptical* Gaussian. Tahap ini diikuti oleh deteksi tepi canny. Hasil simulasi menunjukkan bahwa kisaran estimasi kesalahan sudut yang diijinkan sekitar $\pm 12^{\circ}$. Hasil simulasi juga menunjukkan bahwa metode yang diusulkan memiliki akurasi lebih besar dalam mengenali kabel melengkung daripada metode [22], [24] dan [25].

4. *Robust Real-Time UAV Based Power Line Detection and Tracking* [21]

Paper yang ditulis oleh Guang Zhou, Jinwei Yuan, I-Ling Yen, Farokh Bastan membahas tentang pendeteksian dan pelacakan saluran listrik secara *real-time* dengan menggunakan wahana UAV. *Paper* ini juga menyebutkan bahwa solusi yang ditawarkan jauh lebih baik daripada yang sudah ada. *Paper* ini mengusulkan metode baru yang dapat secara efektif menyesuaikan *threshold* dalam algoritma deteksi. Pertama, *Region of Interest* (RoI) dipilih untuk analisa lebih lanjut dan melacak saluran listrik. Kemudian ditingkatkan kernel pada deteksi tepi dan *hysteresis thresholding* dalam pendekatan Canny. Kemudian dipetakan setiap titik dalam gambar ke ruang fisik yang sesuai. Melalui pemetaan ini, UAV dapat memperoleh koordinat terkait untuk semua *patch.* Selanjutnya Hough *Transform* digunakan untuk menyaring struktur linier yang tidak relevan. Kemudian, model saluran listrik baru direkonstruksi berdasarkan pada saluran listrik dan tiang listrik yang terdeteksi di semua *patch*. Terakhir, pindah ke *frame* berikutnya dan mengulang proses yang sama dari awal. Dari hasil menunjukkan bahwa metode pada *paper* ini mempunyai kinerja yang lebih baik daripada hasil transformasi Canny and Hough secara manual terutama di latar belakang yang kompleks.

5. *High speed Automatic Power Line Detection and Tracking for a UAV-Based Inspection* [22]

   Paper yang ditulis oleh Jingjing Zhang, Liang Liu, Binhai Wang, Xiguang Chen, Qian Wang dan Tianru Zheng menyajikan metode baru dalam mendeteksi dan melacak saluran listrik. Pertama, transformasi Hough digunakan untuk mengekstrak segmen garis dari saluran listrik. Kedua, sesuai dengan karakteristik saluran listrik dari gambar yang tertangkap di udara, mereka menggunakan K-means di Hough *space* untuk mengelompokkan dan menyaring garis lurus, dan kemudian mendeteksi saluran listrik. Terakhir, filter Kalman digunakan untuk melacak saluran listrik di ruang Hough, berdasarkan kontinuitas urutan video. Hasil percobaan menunjukkan bahwa pendekatan yang diusulkan efektif dalam mendeteksi dan melakukan pelacakan saluran listrik secara otomatis di lingkungan yang kompleks. Metode yang diusulkan juga dibandingkan dengan Canny dan PCNF. Hasilnya menunjukkan bahwa metode yang diusulkan mengungguli kedua metode di atas.

6. *Automatic Power Line Detection for a UAV System [23]*

Paper ini mengusulkan algoritma baru untuk sistem deteksi saluran listrik, yaitu dengan menggunakan Gabor filter untuk mengekstrak dan meningkatkan fitur dan tekstur saluran listrik serta Hough Transform untuk mendeteksi saluran listrik. Percobaan dilakukan dengan membuat miniatur saluran listrik sendiri dan pada lingkungan yang tidak berantakan di mana mengandung fitur bergaris yang minimal. Hasil yang ditemukan terbukti efektif dalam mendeteksi saluran listrik. Percobaan dilakukan sebanyak 10 kali pada ketinggian yang berbeda-beda untuk mengukur keandalan sistem. Hasilnya, semakin tinggi percobaan dilakukan, kesuksesan sistem dalam mendeteksi saluran listrik berkurang.

7. *Automatic System for Overhead Power Line Inspection using an Unmanned Aerial Vehicle* [24]

   Dalam *paper* ini, sistem otomatis baru untuk inspeksi saluran listrik oleh *Unmanned Aerial Vehicle* (UAV) disajikan. Metode yang disajikan berfokus pada pencegahan anomali dan pemeliharaan pada saluran listrik. Kontribusi utama dalam *paper* ini adalah desain dan pengembangan sistem inspeksi di udara secara otomatis sepenuhnya tanpa memerlukan intervensi manual pada *ground station* atau pusat kontrol, untuk mendeteksi kemungkinan anomali saluran listrik. Sistem yang diusulkan pada *paper* ini diberi nama dengan RELIFO *project*. Pertama-tama, rencana penerbangan dibuat dan diedit dengan perangkat lunak *ground station*. Penerbangan dilakukan secara otomatis tanpa intervensi manual dan UAV terbang dengan ketinggian 10 meter di atas kabel listrik. Kemudian dilakukan inspeksi dengan mengkalkulasikan jarak dari konduktor saluran listrik ke benda-benda sekitarnya seperti pohon, bangunan dan lain-lain dengan menerapkan metode *stereoscopic vision* menggunakan *frame* gambar yang berurutan. Pada pengaplikasiannya UAV mampu mendeteksi hampir secara *real-time* ketika jarak saluran listrik dengan benda-benda di sekitarnya kurang dari 5 meter. UAV juga mampu mendeteksi titik panas pada saluran listrik dengan menggunakan kamera thermal. Ketika UAV mendeteksi jarak kurang dari 5 meter, sistem yang diusulkan membuat alarm dengan gambar dan data telemetri yang diterima. Alarm menampilkan informasi berikut: ID gambar, gambar, posisi GPS, ketinggian UAV di permukaan laut, dan lain-lain.

8. *Fast Power Line Detection and Localization Using Steerable Filter for Active UAV Guidance* [25]

Dalam *paper* ini membahas tentang deteksi saluran listrik secara cepat serta mengusulkan arsitektur untuk navigasi Unmanned Aerial Vehicle (UAV) berbasis visi aktif. Tahap pendeteksian didasarkan pada *steerable filter* untuk *ridge detection*. Sebagian besar metode deteksi saluran listrik sebelumnya menggunakan metode berbasis deteksi tepi canny, yang dipercaya oleh mereka kurang tepat karena banyak kesalahan yang terdeteksi ketika diterapkan pada latar belakang yang kompleks. Dalam paper ini diusulkan menggunakan *ridge* dalam mendeteksi garis. Sebagai perbandingan, dilakukan pengujian terhadap dua metode deteksi garis yang telah ada, yaitu LSD dan EDLines sebagai dasar untuk mengevaluasi algoritma yang diusulkan. Secara keseluruhan, algoritma yang diusulkan memperoleh hasil deteksi yang lebih baik dengan kesalahan deteksi garis lebih sedikit daripada EDLines dan LSD. Dari segi waktu komputasi, algoritma yang diusulkan juga jauh lebih cepat.

9. *Power Line Detection Via Background Noise Removal* [26]

   Dalam *paper* ini, diusulkan metode deteksi saluran listrik yang akurat, di mana *noise* pada latar belakang dikurangi dengan pengklasifikasian Convolution Neural Network (CNN) sebelum dilakukan ekstraksi saluran listrik. Metode yang diusulkan beroperasi dalam tiga langkah: 1) mengekstrak fitur tepi saluran listrik dari gambar, 2) menggunakan pengklasifikasian CNN untuk menghilangkan *noise* pada latar belakang, 3) menggunakan Hough-Transform (HT) untuk mencari saluran listrik. Pada *paper* ini, pengujian dilakukan dengan membandingkan metode yang diusulkan dengan 2 metode lain yang telah digunakan secara luas, yaitu *Gaussian Model* (GM) dan *Spatial Contexts* (SC). Pengujian melibatkan 3 indeks kinerja, yaitu *Precision Rate* (PR), *Missing Rate* (MR) dan *False Alarm Rate* (FAR). Hasilnya, metode yang diusulkan lebih unggul pada indeks *precision rate* dan *false alarm rate* dibandingkan dengan metode GM dan SC.

10. *Knowledge-based Power Line Detection for UAV Surveillance and Inspection Systems* [27]

    Paper yang ditulis oleh Zhengrong Li, Yuee Liu, Ross Hayward, Jinglan Zhang dan Jinhai Cai dari Univertas Queensland membahas tentang pendeteksian saluran listrik untuk pengawasan dan inspeksi dengan UAV. Dalam *paper* ini, diusulkan metode deteksi saluran listrik dengan menggunakan filter PCNN. Filter PCNN digunakan untuk menghilangkan *noise* di latar belakang pada gambar sebelum

transformasi Hough digunakan untuk mendeteksi garis lurus. Terakhir, pengelompokan K-means digunakan untuk membedakan saluran listrik dengan objek linear lainnya. Metode ini diuji pada data gambar nyata yang telah diambil dari UAV sebelumnya di daerah pedesaan Queensland, karena latar belakang saluran listrik di daerah tersebut berantakan dengan berbagai objek (misalnya rumput, pohon, jalan, dan bangunan). Hasilnya, pendekatan yang diusulkan efektif untuk deteksi saluran listrik secara otomatis. Meskipun masih terdapat kekurangan, yaitu garis linier yang paralel dengan saluran listrik (seperti pagar dan tepi jalan) juga ikut terdeteksi.

11. *Overhead Power Line Detection from UAV Video Images* [28]

    Algoritma real-time dipelajari dalam *paper* ini untuk mendeteksi saluran listrik dari gambar video yang ditangkap dengan menggunakan UAV. Metode yang diusulkan melalui beberapa tahap: Pertama, gambar video dikonversi menjadi gambar biner melalui pendekatan *thresholding* adaptif. Kemudian, Hough Transform digunakan untuk mendeteksi kandidat garis dalam gambar biner. Terakhir, algoritma pengelompokan fuzzy Cmeans (FCM) digunakan untuk membedakan saluran listrik dari kandidat garis yang terdeteksi. Hasil pengujian menunjukkan bahwa algoritma yang diajukan efektif dan mampu mentolerir *noise* dari latar belakang yang rumit dan berbagai iluminasi.

12. *Detection of Thin Lines using Low-Quality Video from Low-Altitude Aircraft in Urban Settings* [29]

    Paper yang ditulis oleh Joshua Candamo, Rangachar Kasturi, Dmitry Goldgof dan Sudeep Sarkar membahas tentang algoritma pendeteksian garis tipis yang mampu memberikan kinerja deteksi tingkat tinggi pada video dengan ketinggian yang rendah di lingkungan perkotaan. Sistem ini dirancang untuk mendukung misi *Urban Search and Rescue* (USAR), yang sering melibatkan pemandangan kota yang sangat berantakan dan kondisi cuaca yang tidak terkendali. Proses deteksi menggunakan estimasi gerakan pada tingkat piksel, dikombinasikan dengan deteksi tepi dan diikuti oleh *windowed hough transform*. Hasilnya, garis terdeteksi dengan benar jika dalam toleransi sudut dan jarak *threshold* yang ditentukan. Sudut *threshold* adalah 40 ±, dan jarak *threshold* adalah 40 px. Algoritma yang diusulkan mampu memberikan performa mendeteksi garis tipis dengan kesalahan kurang dari 25%.

13. *Towards Automatic Power Line Detection for a UAV Surveillance System Using Pulse Coupled Neural Filter and an Improved Hough Transform* [30]

    Dalam *paper* ini, metode baru diusulkan untuk pendeteksian saluran listrik dari gambar yang diperoleh di udara. Tujuannya adalah untuk sistem inspeksi saluran listrik berbasis visi secara otomatis. Beberapa tahapannya adalah sebagai berikut: Pertama, Pulse Coupled Neural Filter (PCNF) diaplikasikan untuk menghilangkan *noise* pada latar belakang dan menghasilkan peta tepi. Kemudian transformasi Hough diterapkan untuk mendeteksi garis lurus. Pada *paper* ini dilakukan perbandingan hasil dengan menggunakan filter yang berbeda-beda antara filter canny, filter sobel dan filter yang diusulkan yaitu PCNF. Hasilnya, filter canny dan sobel mencoba untuk mendeteksi setiap tepi pada gambar dan sangat sensitif terhadap *noise* pada gambar. Sedangkan filter PCNF yang diusulkan lebih fleksibel karena bisa digunakan untuk mendeteksi tepi yang diinginkan daripada mendeteksi seluruh tepi pada gambar. Dalam paper ini menunjukkan bahwa pendekatan yang diusulkan secara signifikan meningkatkan akurasi deteksi saluran listrik di lingkungan yang kompleks.

14. *Automatic Extraction of Power Lines from Aerial Images* [31]

    Paper ini membahas tentang metode dalam ekstraksi saluran listrik dari gambar yang diperoleh oleh kamera digital yang terpasang pada helikopter. Algoritma ini pertama-tama menggunakan transformasi Radon untuk mengekstrak segmen garis dari saluran listrik. Kemudian digunakan metode pengelompokan untuk menghubungkan setiap segmen. Terakhir, diterapkan filter Kalman untuk menghubungkan *gap* antar garis. *Paper* ini membandingkan algoritmanya dengan metode *line mask detector* dan *ratio line detector*, dan mengevaluasi kinerjanya. Hasil percobaan menunjukkan bahwa algoritma pada *paper* ini berhasil mengekstrak saluran listrik dari gambar terlepas dari kompleksitas latar belakang. Hasil penelitian menunjukkan bahwa lebih dari 90% total panjang saluran listrik dapat secara otomatis diekstraksi dan kurang dari 10% dari total panjang saluran listrik diekstraksi secara manual.

15. *Power Line Recognition and Tracking Method for UAVs Inspection* [32]

    Paper yang ditulis oleh Feng Tian, Yaping Wang dan Linlin Zhu dari Universitas Shenyang Aerospace diusulkan algoritma

pengenalan saluran listrik secara otomatis dalam inspeksi menggunakan UAV. Beberapa tahapannya sebagai berikut. Pertama, saluran listrik ditingkatkan dengan *bilateral filtering linear*. Kemudian, transformasi Hough digunakan untuk mendeteksi saluran listrik. Terakhir, pada gambar yang berurut, keadaan saluran listrik diestimasi, dan saluran listrik dilacak dengan estimasi tersebut. Pelacakan dapat mengurangi waktu komputasi. Untuk memverifikasi kinerja metode yang diusulkan, dilakukan percobaan pada data gambar nyata yang diambil dari UAV. Hasilnya menunjukkan metode yang diusulkan pada *paper* ini secara waktu komputasi lebih efisien namun memiliki tingkat kesuksesan pengenalan saluran listrik yang rendah.

16. *Performance Evaluation of Electrical Transmission Line Detection and Tracking Algorithms Based on Image Processing Using UAV* [33]

    Paper yang ditulis oleh Ebru Karakose dari Universitas Firat di Turki menyajikan evaluasi metode yang diperlukan ketika kesalahan mendeteksi saluran transmisi listrik dengan gambar yang diambil dari kamera pada UAV yang bergerak secara otonom. Metode yang diusulkan melalui beberapa tahapan sebagai berikut. Pertama dilakukan akuisisi gambar pada gambar yang diperoleh oleh UAV. Kemudian warna RGB pada gambar tersebut dikonversi menjadi *gray*. Tahap selanjutnya diterapkan morfologi filter pada gambar. Pada gambar yang diperoleh dari langkah pemrosesan morfologi, tepi pada gambar ditentukan dengan menggunakan operator *Canny edge*. Terakhir, diterapkan Hough *transform* untuk mendeteksi garis. Hasilnya menunjukkan bahwa metode ini masih memiliki kekurangan yang signifikan. Sehingga perlu dilakukan analisa dari metode yang telah ada dalam literatur dan mengembangkan pendekatan baru yang sesuai.

17. *Power Line Detection and Tracking Using Hough Transform and Particle Filter* [34]

    Dalam *paper* ini dibahas mengenai metode baru berdasarkan pendekatan filter partikel untuk mendeteksi dan melacak saluran listrik menggunakan video yang diambil dari udara. Dalam algoritma yang diusulkan, gradien gambar dihitung menggunakan operator Sobel. Kemudian metode *threshold* adaptif digunakan untuk mengkonversi gambar menjadi hitam dan putih. Tujuannya untuk mengurangi waktu komputasi. Untuk mendeteksi dan melacak

saluran listrik, ide utamanya adalah menggabungkan Hough Transform dan filter partikel. Partikel-partikel ini didefinisikan dalam Ruang Hough (ρ, θ). Untuk mengevaluasi kinerja metode yang diusulkan, dilakukan pengujian secara *offline* terhadap 100 frame saluran listrik yang ditangkap oleh UAV. Hasilnya, algoritma dapat mendeteksi saluran listrik secara kontinyu setelah beberapa *frame*. Selanjutnya algoritma ini dapat digunakan untuk memberikan perintah yang sesuai pada UAV dalam inspeksi saluran listrik secara otomatis.

18. *Extra Matters Recognition of Transmission System Based on Hough Transform* [35]

    Paper yang ditulis oleh Shujia Yan dan Lijun Jin ini membahas tentang pengenalan materi tambahan dari sistem transmisi berdasarkan hough *transform*. Sistem pengenalan yang diusulkan untuk menguji tingkat kesalahan identifikasi atau *False Idetification Rate* (FIR). Pada *paper* ini, sistem pendeteksian saluran transmisi pada gambar berdasarkan pada Hough Transform dan pada pendeteksian tersebut dianalisa jika ada masalah tambahan pada saluran tersebut. Sebelum pendeteksian saluran transmisi, terdapat beberapa tahap pengolahan citra. Pertama, warna gambar RGB dikonversi menjadi *gray*. Setelah itu, filter *smoothing* spasial digunakan untuk meningkatkan kualitas gambar. Pengujian dilakukan pada 150 sampel. Dan hasilnya, tingkat kesalahan identifikasi kurang dari 15%.

19. *A Method For Detecting Power Lines In UAV Aerial Images* [36]

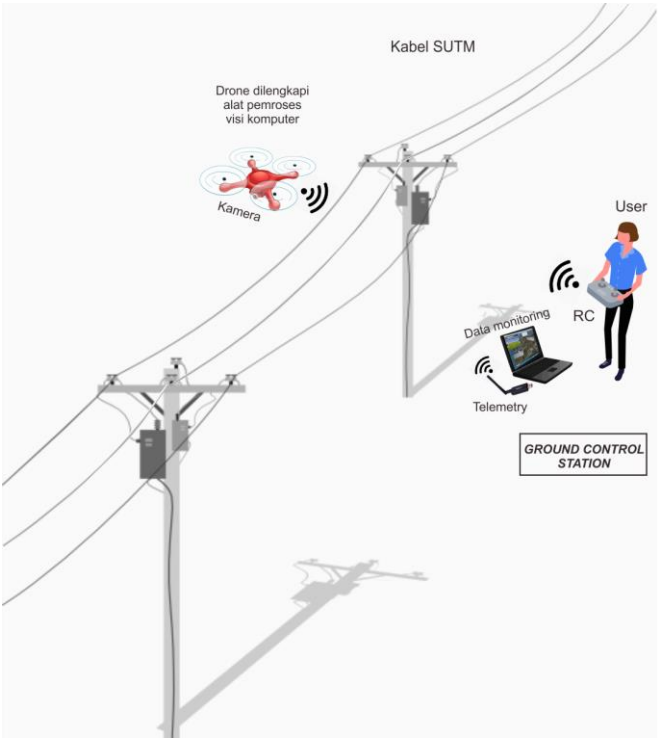
    Dalam tulisan ini diusulkan pendekatan baru untuk deteksi saluran listrik dari gambar yang ditangkap oleh *Unmanned Aerial Vehicle* (UAV). Pendekatan yang diusulkan yaitu dengan menggunakan *Randomized Hough Transform* (RHT), atau disebut dengan *Area Divide Randomized Hough Transform* (ADRHT) untuk mendeteksi fitur linear dan menghindari kesalahan deteksi di lingkungan yang kompleks. Pengujian dilakukan dengan membandingkan antara algoritma yang sudah terkenal, yaitu RHT dan LSD. Hasilmya, dibandingkan dengan algoritma *Line Segment Detector* (LSD) dan algoritma RHT, pendekatan pada *paper* ini lebih cepat dalam proses komputasi dan dari hasil pengujian lebih efisien terhadap lingkungan yang kompleks. Metode LSD dan RHT sensitif terhadap kondisi latar belakang yang kompleks, banyak kesalahan yang terdeteksi selama pengujian.

20. *The Design and Implementation of a Hotline Tracking UAV* [37]

    Dalam *paper* ini dibahas mengenai desain dan pengembangan sistem pelacakan saluran listrik berbasis visi untuk inspeksi yang digunakan oleh *quad-rotor*. *Quad-rotor* diterbangkan di atas saluran listrik, kemudian kombinasi dari *Steerable filter* dan *Probabilistic Hough Line transform* digunakan untuk mendeteksi saluran listrik. *Unscented Kalman filter* digunakan untuk melacak posisi saluran listrik terhadap robot. Sistem kontrol didesain berdasarkan dengan metode *backstepping*. Kemudian pendekatan tersebut diuji dengan *Virtual Robot Experimentation Platform* (V-REP) berdasarkan lingkungan simulasi fisik sebenarnya. Hasilnya, filter yang diusulkan telah berhasil diuji dalam lingkungan simulasi fisik. *Paper* ini menunjukkan bahwa pendekatan yang diusulkan pragmatis dalam melacak saluran listrik secara mandiri.

Dalam tugas akhir ini diusulkan metode pendeteksian saluran listrik secara *real time* dengan mengekstrak garis vertikal pada gambar yang diperoleh dari UAV menggunakan *morphological operations*. Informasi posisi saluran listrik yang terdeteksi pada *frame* akan digunakan sebagai navigasi *drone* untuk menjalankan misi *autonomous* mengikuti saluran listrik dengan kontrol PID.

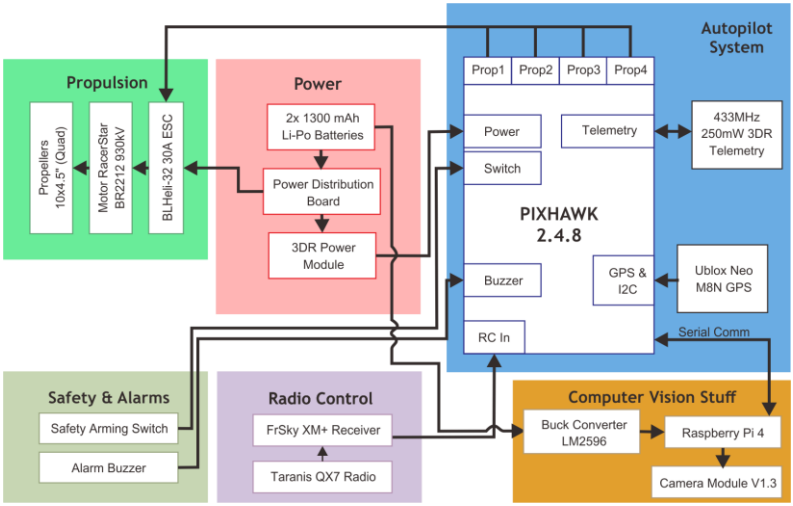
BAB 3
PERANCANGAN SISTEM



Gambar 3. 1 Ilustrasi rancangan sistem

Alat yang akan dibuat pada tugas akhir ini adalah *drone* yang dapat menjejaki kabel Saluran Udara Tegangan Menengah (SUTM) secara mandiri dengan memanfaatkan visi komputer. Perancangan sistem akan dibagi menjadi tiga bagian, yaitu perancangan *drone*, perancangan visi komputer dan perancangan kontrol *drone* dengan memanfaatkan data yang diperoleh dari visi komputer. Nantinya *drone* akan diterbangkan secara manual oleh pilot ke atas kabel SUTM dan pilot mengarahkan posisi depan *drone* sejajar dengan kabel SUTM. Kemudian ketika posisi *drone* sudah tepat di atas kabel SUTM dan sejajar dengan kabel SUTM, maka mode penerbangan akan diubah menjadi *guided* untuk menjalankan misi penjejakan kabel SUTM secara mandiri. *Drone* akan

mempertahankan ketinggiannya di udara ketika melakukan penjejakan kabel SUTM. *Drone* akan dilengkapi dengan kamera dan raspberry pi sehingga *drone* dapat memproses gambar yang tertangkap oleh kamera dengan visi komputer. Berikut ini ditampilkan blok diagram keseluruhan perangkat keras sistem penjejakan kabel SUTM.

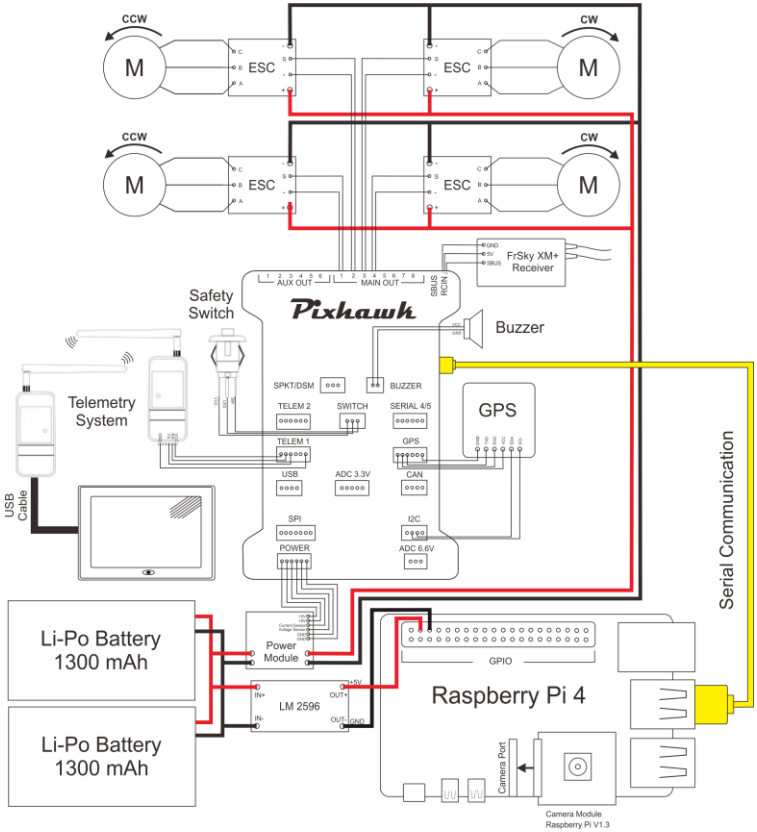


Gambar 3. 2 Blok diagram perangkat keras sistem penjejakan kabel SUTM

Pada gambar blok diagram di atas dapat dilihat bahwa *flight controller* pixhawk yang merupakan otak dari *drone* dalam mengatur penerbangan ketika di udara dihubungkan dengan berbagai komponen-komponen pendukungnya seperti GPS, *telemetry*, *safety switch*, *buzzer*, *receiver*, *power module*, ESC, motor dan lain-lain. GPS, *telemetry*, *safety switch*, *buzzer*, *power module*, *receiver* dan ESC dihubungkan melalui port yang telah disediakan pada pixhawk, yaitu masing-masing secara berurutan seperti yang dijelaskan pada tabel 3.1. Supplai tegangan pada *drone* ini menggunakan 2 buah baterai lipo 4S berukuran 1300mAh yang dihubungkan secara paralel untuk mendapatkan jumlah arus yang ganda. Selain itu, *drone* juga dihubungkan dengan raspberry pi dan kamera sebagai tempat pemrosesan visi komputer ketika *drone* terbang di udara. Skema rangkaian elektronika dari komponen-komponen yang terdapat pada blok diagram di atas dapat dilihat pada gambar 3.3.

Tabel 3. 1 Hubungan antara komponen-komponen dengan pixhawk

Nama Komponen	Dihubungkan ke Pixhawk melalui port
GPS Ublox M8N	GPS & I2C
433 MHz 250 mW 3DR Telemetry	TELEM1
Safety Arming Switch	SWITCH
Alarm Buzzer	BUZZER
Power Module	POWER
FrSky XM+ Receiver	RC IN
BLHeli-32 30A ESC	MAIN OUT 1,2,3,4



Gambar 3. 3 Skema rangkaian elektronika sistem

3.1. Perancangan *Drone*

Drone merupakan alat utama yang digunakan pada tugas akhir ini. *Drone* bertugas untuk membawa alat pemroses visi komputer berupa kamera dan raspberry pi ke udara. Perancangan *drone* perlu dilakukan terlebih dahulu agar *drone* dapat terbang dengan stabil dan dapat melakukan penjejakan kabel SUTM secara mandiri dengan optimal.

Pada tahap ini akan dilakukan pertimbangan terhadap penyusunan serta pemasangan komponen-komponen pada *frame* agar daya angkat yang dihasilkan oleh *drone* sesuai dengan kemampuan dari motor yang digunakan. Komponen-komponen yang akan digunakan pada *drone* ini beserta nama komponennya dapat dilihat pada tabel berikut.

Tabel 3. 2 Komponen-komponen penyusun *drone*

Penyusun	Komponen	Nama Komponen
Drone	*Frame*	DJI *Flamewheel* F450
	Flight controller	Pixhawk 2.4.8
	Motor	*Brushless* Motor *RacerStar* BR2212 930 kV
	ESC	FuriBee H32 BLHeli-32 30A *Brushless*
	Transmitter	Taranis QX7
	Receiver	FrSky XM+
	GPS	Ublox NEO M8N GPS
	Propeller	*Propeller* 10x4.5”
	Baterai	2x Baterai LiPo 4S 1300 mAh

Seluruh komponen yang disebutkan di atas akan diletakkan pada *frame drone*. Berikut ini merupakan penjelasan mengenai seluruh komponen yang disebutkan di atas.

1. DJI *Flamewheel* F450

   Drone akan menggunakan *frame quadcopter* dengan tipe x. *Frame* yang akan digunakan adalah DJI *Flamewheel* F450. *Frame* F450 dilengkapi dengan *top plate*, *bottom plate* dan 4 buah *frame arm*. Pada *frame* terdapat *power distribution board* (PDB) yang berfungsi mendistribusikan suplai dari baterai Li-Po ke 4 buah ESC.
2. Pixhawk 2.4.8

   Drone akan menggunakan *flight controller* Pixhawk 2.4.8. *Flight controller* akan dihubungkan dengan komponen-komponen

yang lain seperti ESC, GPS, *buzzer*, *safety switch*, *power module*, *receiver*, *telemetry*, dan Raspberry Pi. Pixhawk dapat dihubungkan dengan raspberry pi melalui komunikasi serial antara *port* usb yang terdapat pada Raspberry Pi dengan *port micro* usb yang terdapat pada pixhawk.

3. *Brushless* Motor *RacerStar* BR2212 930 kV

   Drone akan menggunakan 4 buah motor DC *brushless* dengan merek *RacerStar* BR2212 930 kV. Motor ini dipilih karena mempunya daya angkat/*pull* yang cukup besar tiap motornya, yaitu sebesar 660 gram. Sehingga bila dijumlahkan, *drone* mampu mengangkat beban hingga 2640 gram.

4. *Propeller* 10x4.5”

   Drone akan menggunakan *propeller* dengan diameter 10 inci dan *pitch* 4.5 inci. *Propeller* yang digunakan terbuat dari material *Acrylonitrile Butadiene Styrene* (ABS). *Drone* akan menggunakan *propeller* dengan jumlah dua buah *propeller* CCW (*Counter Clock Wise*) dan dua buah *propeller* CW (*Clock Wise*). Pemilihan *propeller* ini disesuaikan dengan motor yang digunakan.

5. FuriBee H32 BLHeli-32 30A *Brushless*

   Drone akan menggunakan *Electronic Speed Controller* (ESC) FuriBee H32 BLHeli-32 30A *Brushless*. ESC ini memiliki nilai rating arus sebesar 30 A sesuai dengan namanya. ESC ini dapat beroperasi pada baterai 2-4 sel.

6. GPS Ublox NEO M8N

   Drone akan menggunakan GPS Ublox NEO M8N. GPS digunakan untuk memberikan informasi terkait ketinggian dan posisi *drone* ketika terbang dengan memanfaatkan GPS. Suplai tegangan pada GPS ini adalah 3-5V.

7. FrSky XM+ dan Taranis QX7

   Receiver dan *transmitter* yang akan digunakan pada *drone* ini adalah FrSky XM+ dan Taranis QX7. *Receiver* dan *transmitter* digunakan untuk menerbangkan *drone* dengan kontrol manual. Receiver FrSky XM+ menggunakan protokol SBUS dan beroperasi pada tegangan 3,7-10V.

8. Baterai Li-Po 4S 1300mAh.

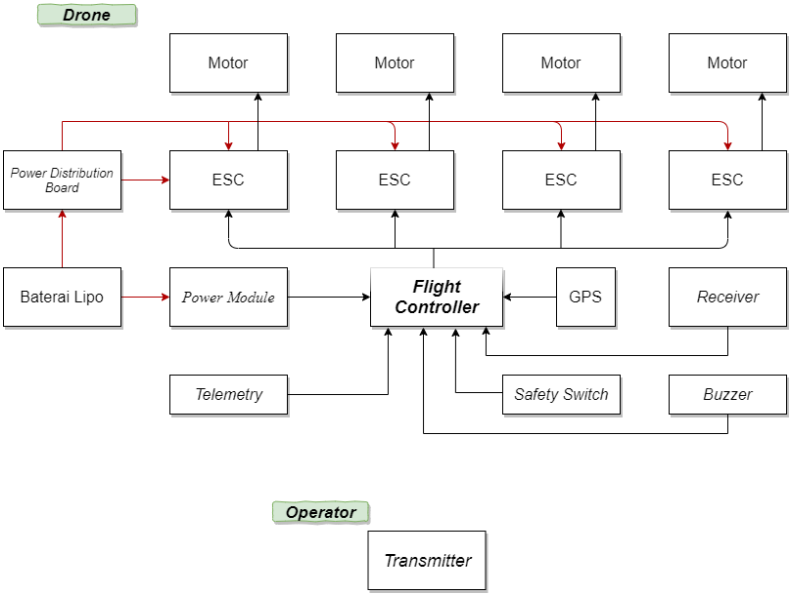
   Suplai tegangan pada *drone* ini berasal dari baterai Li-Po 4S 1300mAh. Baterai yang digunakan pada *drone* ini berjumlah 2 buah dan dihubungkan secara seri untuk mendapatkan jumlah arus yang

ganda. Total jumlah arus yang diperoleh dengan menghubungkan baterai ini secara seri adalah 1300mAh x 2 = 2600mAh.

Komponen-komponen yang disebutkan tadi kemudian dihubungkan satu sama lain berdasarkan fungsinya. Berikut ini merupakan tabel dan gambar yang menjelaskan sistem wiring yang menghubungkan antar komponen pada *drone*.

Tabel 3. 3 Sistem *wiring* pada *drone*

Komponen	Bagian Komponen	Port
ESC	VCC	+ Baterai Li-Po
	GND	- Baterai Li-Po
	Signal	*Main Out* 1 s/d 4 *Flight Controller*
Receiver	SBUS	SBUS *Flight Controller*
	VCC	+5 V *Flight Controller*
	GND	GND *Flight Controller*
GPS	VCC	+5 V *Flight Controller*
	GND	GND *Flight Controller*
	RX	TX *Flight Controller*
	TX	RX *Flight Controller*
	SDA	SDA *Flight Controller*
	SCL	SCL *Flight Controller*
Buzzer	VCC	+5 V *Flight Controller*
	GND	GND *Flight Controller*
Telemetry	VCC	+5 V *Flight Controller*
	GND	GND *Flight Controller*
	RX	TX *Flight Controller*
	TX	RX *Flight Controller*
Safety Switch	VCC	+3.3 V *Flight Controller*
	GND	GND *Flight Controller*
	LED *Safety*	GND *Flight Controller*
Power Module	VCC	+5 V *Flight Controller*
	VCC	+5V *Flight Controller*
	Current Sensor	V*Current Flight Controller*
	Voltage Sensor	VBatt *Flight Controller*
	GND	GND *Flight Controller*
	GND	GND *Flight Controller*



Gambar 3. 4 Sistem *wiring* pada *drone*

Flight controller akan mengendalikan komponen-komponen yang terhubung dengannya. *Flight controller* dihubungan dengan 4 *electronic speed controller* (ESC) yang masing-masing ESC akan mengendalikan kecepatan putaran dari motor. *Flight controller* juga dihubungkan dengan *receiver* dimana fungsinya menerima sinyal dari *transmitter* yang digunakan oleh operator untuk mengirimkan perintah berupa *throttle*, *pitch*, *roll*, *yaw* dan lain-lain. Selain itu terdapat GPS yang dihubungkan pada *flight controller* yang berfungsi untuk mendapatkan posisi koordinat dari *drone*. *Flight controller* juga terhubung dengan *telemetry* yang berfungsi menghubungkan *drone* dengan *ground control station* (GCS) untuk menerima informasi data berupa *attitude* dan kondisi *drone* ketika mengudara, dan GCS juga dapat mengirimkan perintah ke *drone* melalui *telemetry*. *Flight controller* juga dilengkapi dengan *safety switch* dan *buzzer* yang masing-masing gunanya untuk mengaktifkan atau menonaktifkan output ke motor agar bisa di*arming* dan untuk mengindikasikan perubahan status pada *drone*. Baterai Li-Po digunakan sebagai suplai tegangan pada komponen-komponen yang disebutkan di atas.

3.2. Perancangan Visi Komputer

Pada tahap ini akan dijelaskan mengenai perancangan alat yang memproses visi komputer dan perancangan visi komputer dalam mendeteksi kabel SUTM. Pada perancangan alat yang memproses visi komputer akan dibahas mengenai komponen-komponen penyusun alat yang memproses visi komputer. Sedangkan pada perancangan visi komputer akan dibahas mengenai tahapan visi komputer dari data *input* berupa gambar kabel SUTM yang diperoleh dari kamera yang terpasang pada *drone* menjadi informasi yang dibutuhkan oleh *drone* untuk terbang menjejaki kabel SUTM secara mandiri.

3.2.1. Perancangan Alat yang Memproses Visi Komputer

Agar *drone* dapat melakukan pemrosesan visi komputer pada saat di udara, maka perlu dilakukan perancangan alat yang memproses visi komputer. Komponen utama yang digunakan dalam melakukan pemrosesan visi komputer adalah minikomputer raspberry pi. Tabel berikut ini akan menjelaskan komponen-komponen yang digunakan pada alat yang memproses visi komputer.

Tabel 3. 4 Komponen penyusun yang memproses visi komputer

Penyusun	Komponen	Nama Komponen
Alat yang memproses visi komputer	Minikomputer	Raspberry Pi 4 model B
	Kamera	Modul Kamera Raspberry Pi V1.3
	Buck converter	LM 2596
	Baterai	2x Baterai LiPo 4S 1300 mAh

Seluruh komponen yang disebutkan di atas akan diletakkan pada *frame drone*. Berikut ini merupakan penjelasan dari seluruh komponen yang disebutkan di atas.

1. Raspberry Pi 4 Model B

   Drone ini akan dilengkapi dengan minikomputer, yaitu raspberry pi 4 model B. Raspberry pi 4 bisa dikatakan merupakan versi terbaru dari produk mereka. Beberapa fitur pada Raspberry Pi 4 juga lebih diunggulkan dibandingkan dengan versi sebelumnya, baik itu dari segi memori, kecepatan mentransfer data, kecepatan prosessor dan lain-lain. Pada tugas akhir ini akan digunakan

raspberry pi 4 model B dengan RAM 2 GB. Minikomputer raspberry pi 4 akan digunakan untuk mengolah data berupa gambar yang ditangkap oleh kamera menjadi informasi yang dibutuhkan oleh *drone* untuk mengendalikan perilaku *drone* sehingga dapat terbang menjejaki kabel SUTM secara mandiri.

2. Modul Kamera Raspberry Pi V1.3

   Drone ini akan dilengkapi dengan modul kamera raspberry pi V1.3. Modul ini dapat dihubungkan langsung dengan raspberry pi melalui konektor CSI yang telah disediakan. Kamera ini beresolusi 5MP dan mampu merekam video hingga 1080p HD pada 30 fps. Kamera ini akan dipasang pada sisi bawah *drone* dan selalu menghadap ke bawah..

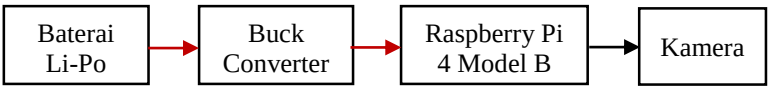
3. *Buck Converter* LM2596

   Komponen ini berfungsi untuk menurunkan tegangan dari tegangan suplai. Tegangan pada baterai yang semula bernilai 14.8 V akan diturunkan menjadi 5V sehingga dapat digunakan sebagai tegangan masukan pada minikomputer raspberry pi.

4. Baterai Li-Po 4S 1300 mAh

   Suplai tegangan yang diperoleh dari raspberry pi akan dihasilkan dari dua buah baterai Li-Po dengan kapasitas 1300 mAh 4S. Baterai ini akan dihubungkan secara seri sehingga didapatkan jumlah arus yang ganda.

Komponen-komponen yang disebutkan di atas kemudian dihubungkan satu sama lain. Berikut ini merupakan diagram dari sistem *wiring* alat yang memproses visi komputer.



Gambar 3. 5 Sistem *wiring* alat yang memproses visi komputer

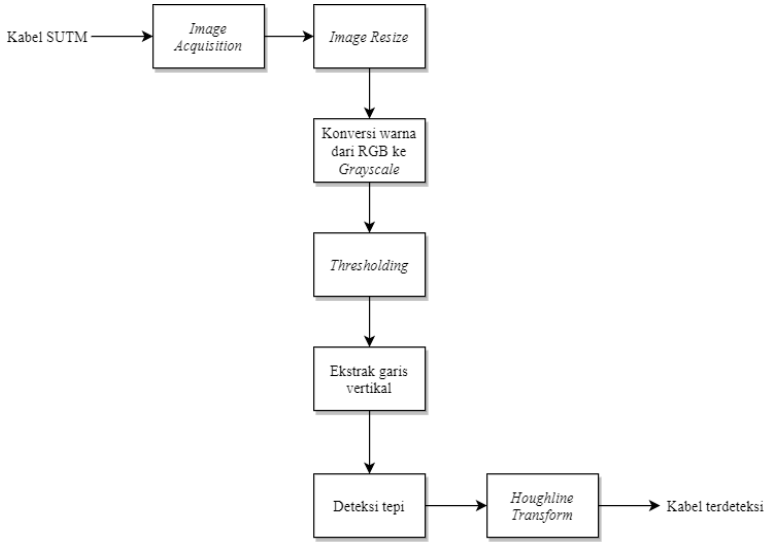
Tabel 3. 5 Sistem *wiring* alat yang memproses visi komputer

Komponen	Bagian Komponen	Port
Buck Converter LM2596	VCC	+ Baterai Li-Po
	GND	- Baterai Li-Po
Raspberry Pi 4 Model B	VCC (Pin 4)	+ *Buck Converter*
	GND (Pin 6)	- *Buck Converter*
Kamera	*Camera Port*	Raspberry Pi *Camera Module*

Suplai tegangan yang dihasilkan oleh baterai Li-Po akan diturunkan terlebih dahulu dengan *buck converter* LM2596 sehingga menghasilkan tegangan 5V. Tegangan ini kemudian diaplikasikan pada raspberry pi melalui pin 4 yang merupakan 5V dan pin 6 yang merupakan GND pada GPIO raspberry pi.

3.2.2. Visi Komputer

Perancangan visi komputer dalam mendeteksi kabel SUTM akan menggunakan *library* OpenCV. Gambar kabel SUTM yang diperoleh dari kamera akan melewati beberapa tahapan hingga pada akhirnya didapatkan informasi hanya berupa objek yang diinginkan yaitu kabel SUTM dan memblok *noise* disekitar gambar. Kabel SUTM yang terdeteksi akan memberikan informasi berupa posisi kabel SUTM pada koordinat x dan y pada *frame*. Berikut ini merupakan blok diagram visi komputer dalam mendeteksi kabel SUTM.



Gambar 3. 6 Blok diagram visi komputer dalam mendeteksi kabel SUTM

3.2.2.1 *Image Resize*

Kamera yang akan digunakan pada tugas akhir ini adalah raspberry pi *camera module*. Resolusi *default* pada kamera ini adalah 640

x 480. Kemudian resolusi ini akan diperkecil ukurannya menjadi 320 x 240. Tujuan pengecilan resolusi ini adalah agar mendapatkan *framerate per second* lebih besar sehingga proses visi komputer yang dilakukan oleh minikomputer raspberry pi dapat berjalan lebih cepat.

3.2.2.2 *Konversi Warna dari RGB ke Grayscale*

Gambar yang diperoleh pertama kali mempunyai format dengan warna RGB yang mempunyai 3 kanal yaitu R (*Red*), G (*Green*) dan B (*Blue*). Gambar ini tentu saja mempunyai informasi yang sangat banyak untuk dilakukan proses visi komputer dibandingkan dengan gambar *grayscale* yang hanya mempunyai 1 kanal. Selain itu minikomputer tempat pengolahan gambar juga membutuhkan kinerja yang cukup ekstra dalam mengolah gambar RGB. Sehingga konversi warna perlu dilakukan untuk mengurangi informasi yang ada agar proses visi komputer dapat berjalan lebih singkat dan mudah. Rumus yang digunakan dalam mengkonversi warna RGB menjadi *grayscale* adalah sebagai berikut. [35]

$$GRAY = 0.299 \times R + 0.587 \times G + 0.114 \times B \quad (3.1)$$

Dimana R, G dan B adalah komponen merah, hijau dan biru dari setiap piksel. *GRAY* adalah nilai piksel dari hasil konversi. Konversi warna RGB menjadi *grayscale* dapat ditulis dengan fungsi berikut pada openCV: cv2.COLOR_BGR2GRAY. Berikut program konversi warna dari RGB ke *grayscale* beserta hasilnya ditampilkan pada gambar di bawah.

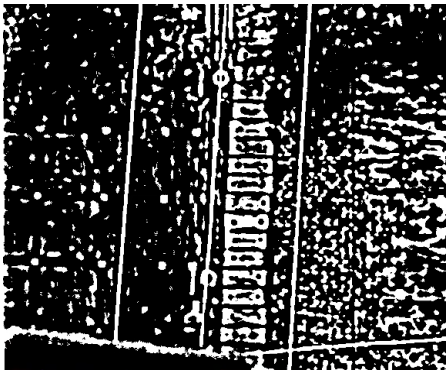
```
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

Gambar 3. 7 Konversi warna dari RGB ke *Grayscale*

3.2.2.3 *Thresholding*

Setelah mengkonversi warna dari RGB ke *grayscale*, selanjutnya akan dilakukan *thresholding*. *Thresholding* akan menghasilkan keluaran 0 atau 1 pada tiap piksel sehingga dapat mempercepat waktu komputasi dan proses visi komputer. *Adaptive threshold* metode ADAPTIVE_THRESH_MEAN_C akan digunakan pada tugas akhir ini dengan tipe *threshold* THRESH_BINARY atau *threshold binary.* ADAPTIVE_THRESH_MEAN_C merupakan metode *threshold* dimana nilai *threshold*-nya adalah rata-rata dari area sekitar. Dan THRESH_BINARY atau *threshold binary* merupakan tipe *threshold* dimana nilai piksel yang memiliki nilai di bawah nilai *threshold* akan diubah menjadi nilai minimum yaitu 0 dan nilai piksel yang memiliki nilai piksel di atas *threshold* akan menjadi nilai maksimum yaitu 1. Berikut program *adaptive thresholding* yang diaplikasikan pada *frame* dan hasilnya.

```
Thresholding = cv2.adaptiveThreshold (gray, 255,
cv2.ADAPTIVE_THRESH_MEAN_C,  cv2.THRESH_BINARY, 15,
-2)
```

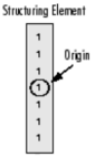


Gambar 3. 8 *Frame* hasil *thresholding*

3.2.2.4 *Ekstrak Garis Vertikal*

Agar diperoleh hanya informasi yang diinginkan, yaitu berupa kabel Saluran Udara Tegangan Menengah (SUTM), maka pada *frame* perlu dilakukan ekstrak garis vertikal sehingga informasi yang dilewatkan hanya berupa garis vertikal yang terdapat pada gambar, sedangkan garis

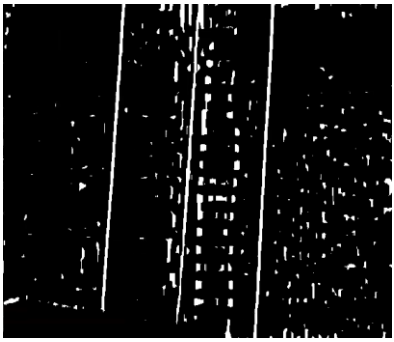
horizontal maupun *noise* seperti pohon, rumah, kendaraan dan lain-lain tidak dilewatkan. Cara ini bisa dilakukan dengan menggunakan operasi morfologi. Operasi morfologi merupakan tahapan visi komputer yang memproses gambar berdasarkan *structuring element* yang telah ditentukan sebelumnya, atau lebih dikenal sebagai kernel. Karena informasi yang ingin diekstrak adalah garis vertikal, maka *structure element* yang sesuai untuk itu akan memiliki bentuk sebagai berikut.



Gambar 3. 9 *Structure element* garis vertikal

Selain itu, filter *erode* perlu diterapkan setelah menentukan *structure element* untuk menghilangkan *noise-noise* kecil setelah esktraksi garis vertikal dilakukan. Berikut merupakan program untuk mengekstrak garis vertikal dan hasilnya.

```
vertical = np.copy(thresholding)
rows = vertical.shape[0]
verticalsize = rows // 44
verticalStructure = cv2.getStructuringElement
(cv2.MORPH_RECT, (1, verticalsize))
vertical = cv2.erode(vertical, verticalStructure)
```

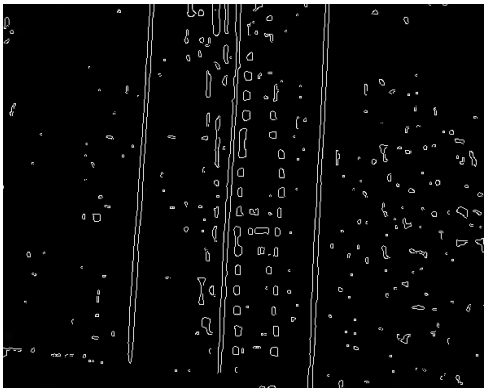


Gambar 3. 10 Hasil ekstraksi garis vertikal

3.2.2.5 *Deteksi Tepi*

Setelah mengekstrak garis vertikal pada *frame*, selanjutnya dilakukan deteksi tepi. Terdapat banyak metode dan operator yang dapat digunakan dalam mendeteksi tepi. Pada tugas akhir ini digunakan deteksi tepi Canny. Deteksi tepi Canny diterapkan untuk mendapatkan nilai yang selanjutnya akan diberikan sebagai parameter untuk transformasi Hough. Dua nilai *threshold* digunakan ketika ekstraksi tepi dilakukan. Dalam metode yang diusulkan, nilai *threshold* yang pertama adalah 20 dan yang kedua adalah 255. Kedua nilai ini diperoleh setelah melakukan kalibrasi manual dengan menggunakan *trackbar* pada kondisi aktual di lapangan. Berikut merupakan program deteksi tepi Canny dan hasilnya.

```
edges = cv2.Canny(vertical, 20,255,apertureSize=3)
```



Gambar 3. 11 Deteksi tepi Canny

3.2.2.6 *Transformasi Houghline*

Transformasi *Houghline* merupakan salah satu teknik yang paling umum digunakan untuk mendeteksi garis.Transformasi *Houghline* akan digunakan dalam mendeteksi kabel Saluran Udara Tegangan Menengah (SUTM). Transformasi *Houghline* dalam mendeteksi garis lurus dapat direpresentasikan dalam persamaan berikut.

$$x \cos\theta + y \sin\theta = \rho \tag{3.2}$$

Dimana x and y merupakan koordinat horizontal dan vertikal dari gambar dan ρ dan θ adalah parameter garis. Ketika membedakan

garis lurus dalam sebuah gambar, transformasi hough memetakan garis dalam koordinat Cartesian menjadi sebuah titik di koordinat kutub (θ, ρ).

Pada tugas akhir ini, transformasi *houghline* yang digunakan dalam mendeteksi garis adalah transformasi *houghline probabilistic*. Transformasi *houghline probabilistic* digunakan karena lebih efisien dan secara komputasi lebih cepat dibandingkan dengan transformasi *houghline* biasa. Pada *library* openCV, transformasi *houghline probabilistic* dapat digunakan dengan memanggil fungsi cv2.HoughLinesP(). .

Pada *houghline probabilistic*, nilai *minLineLength* diset pada angka 400, sehingga apabila ada segmen garis yang panjangnya kurang dari 400 maka akan ditolak. Dan maxLineGap diset pada angka 225, sehingga apabila ada gap antara segmen melebihi 225 maka tidak dianggap sebagai satu garis. Berikut ini merupakan program deteksi kabel Saluran Udara Tegangan Menengah (SUTM) dengan menggunakan transformasi *houghline probabilistic* dan hasilnya.

```
lines = cv2.HoughLinesP(edges, 1, np.pi / 180, 100,
minLineLength=400, maxLineGap=225)

if lines is not None:
    for line in lines:
        x1, y1, x2, y2 = line[0]
        cv2.line(frame50, (x1, y1), (x2, y2), (0,
        255, 0), 2)
```

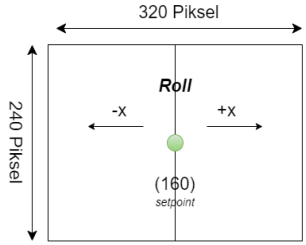
Gambar 3. 12 Hasil deteksi garis menggunakan transformasi *Houghline*

3.3. Perancangan Kontrol *Drone*

Setelah melewati beberapa tahap visi komputer, data yang diperoleh akan digunakan untuk mengontrol perilaku *drone* dalam menjalankan misi *autonomous* menjejaki kabel SUTM. Perilaku *drone* akan dikendalikan dengan menggunakan kontrol PID. Pergerakan yang akan dikontrol pada *drone* adalah pergerakan *roll*. Sedangkan pergerakan *pitch* dan *yaw* akan dibuat konstan. *Roll* akan membuat *drone* terbang ke kiri dan ke kanan, sehingga perlu dikontrol agar *drone* dapat mengoreksi pergerakannya terhadap *setpoint* yang telah ditentukan ketika melakukan penjejakan kabel SUTM.

Gambar 3. 13 Prinsip kerja *drone*

Pergerakan *roll* pada *drone* menggunakan referensi *frame* kamera yang beresolusi 320x240 piksel. Koordinat x positif akan membuat *drone* melakukan *roll* ke kiri dan koordinat x negatif akan membuat *drone* melakukan *roll* ke kanan. Berikut ini merupakan ilustrasi pergerakan *drone* dengan menggunakan referensi *frame* kamera.



Gambar 3. 14 Pergerakan *drone* dengan menggunakan referensi *frame*

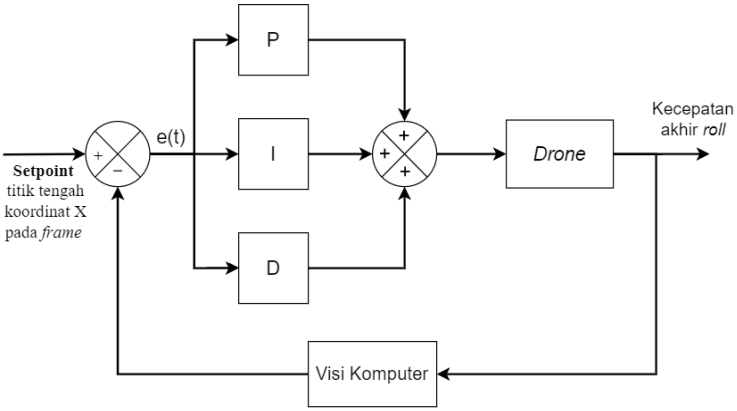
Kontrol PID pada *drone* akan memanfaatkan data yang telah diperoleh dari hasil pemrosesan visi komputer, yaitu kabel SUTM yang terdeteksi. Karena kabel SUTM yang terdeteksi terdapat 3 kabel, maka diperlukan adanya garis kontrol. Garis kontrol merupakan rata-rata dari semua koordinat x kabel SUTM yang terdeteksi saat ini pada *frame*. Berikut ini merupakan kode program untuk mendapatkan garis kontrol beserta hasilnya.

```
lines = cv2.HoughLinesP(edges, 1, np.pi / 180, 100,
minLineLength=400, maxLineGap=225)
xCable = 0

   if lines is not None:
      for line in lines:
          x1, y1, x2, y2 = line[0]
          cv2.line(gambar, (x1, y1), (x2, y2), (0,
          255, 0), 2)
          xCable = xCable + x1 + x2

      xCable /= len(lines) * 2
      cv2.line(gambar, (xCable, 0), (xCable, 240),
      (255, 0, 0), 3)
```

Gambar 3. 15 Garis kontrol pada *frame*

Garis kontrol ini akan dibandingkan nilainya dengan *setpoint* yang telah ditentukan sehingga akan menghasilkan nilai *error* yang

selanjutnya akan diteruskan sebagai masukan PID. *Setpoint* akan diambil nilai setengah dari panjang sumbu x pada *frame*. Karena *frame* beresolusi 320x240 maka nilai setengahnya adalah x = 160. Keseluruhan sistem kontrol dapat dilihat pada blok diagram berikut.



Gambar 3. 16 Blok diagram kontrol PID *drone*

Gambar 3.15 merupakan blok diagram kontrol PID *drone* yang mampu menjelaskan keseluruhan sistem kontrol. Kecepatan akhir *drone* dapat ditentukan dengan menggunakan rumus PID sebagai berikut.

$$V = Kp\ x\ error + Ki\ x\ (error + Previous\ error) + Kd\ x\ (error - Previous\ error) \quad (3.3)$$

Dengan Kp, Ki dan Kd merupakan konstanta PID yang akan diperoleh dengan cara tuning manual, yaitu dengan menerbangkan *drone* sambil mengubah-ubah nilai Kp, Ki dan Kd hingga didapatkan nilai konstanta sesuai dengan pergerakan *drone* yang diharapkan. Setelah kecepatan *roll* akhir diperoleh melalui kontrol PID, kemudian Raspberry Pi akan mengirimkan pesan MAVLink ke *flight controller* dengan menggunakan *library dronekit* ketika *drone* berada pada mode penerbangan *guided*. Sehingga pada akhirnya *drone* dapat melakukan penjejakan kabel Saluran Udara Tegangan Menengah (SUTM) secara mandiri.

BAB 4
HASIL DAN ANALISA DATA

4.1. Realisasi Alat

Gambar 4. 1 Realisasi alat terlihat dari atas

Gambar 4. 2 Realisasi alat terlihat dari depan

Berikut merupakan spesifikasi dari *drone* tersebut.

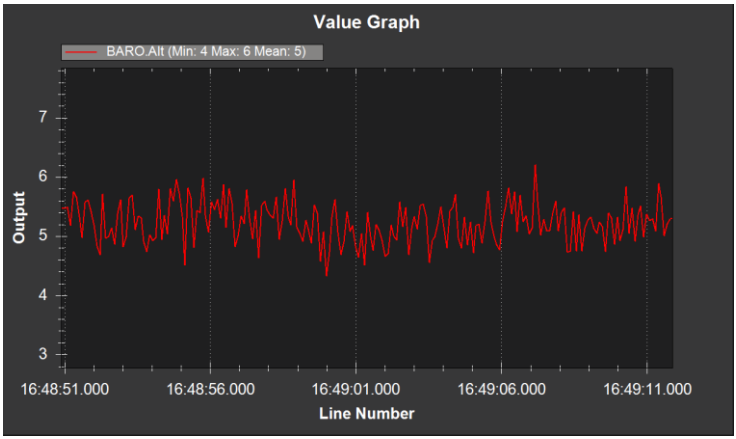
Tabel 4. 1 Spesifikasi alat

Berat	1057 gram
Panjang	32 cm
Lebar	33 cm
Tinggi	26,5 cm
Lama Terbang	$\pm$ 5 menit

4.2. Pengujian Tingkat Kestabilan *Drone*

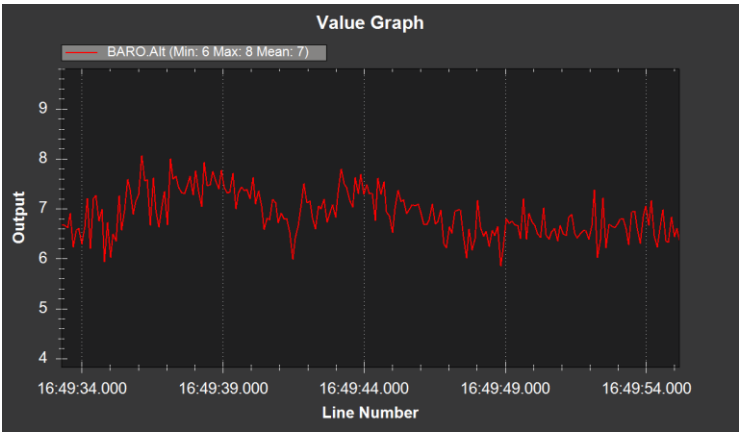
Pengujian ini bertujuan untuk menguji tingkat kestabilan *drone* dalam mempertahankan ketinggian serta posisi *drone* pada saat terbang di udara. Pengujian dilakukan dengan menerbangkan *drone* pada mode penerbangan *loiter*. Mode *loiter* merupakan mode penerbangan yang secara otomatis mencoba untuk mempertahankan lokasi saat ini, ketinggian dan arah *drone*. Setelah *drone* diterbangkan hingga mencapai ketinggian tertentu, kemudian pilot akan melepas kendali *drone*. Selanjutnya dilihat perilaku *drone* ketika di udara pada saat mode *loiter*. Saat *drone* terbang di udara akan dilakukan pengambilan nilai *altitude* dari *drone* yang berasal dari pembacaan sensor barometer yang terdapat pada *flight controller* pixhawk. Berikut ini merupakan hasil dari beberapa kali pengujian tingkat kestabilan *drone*.

Gambar 4. 3 Pengujian tingkat kestabilan *drone*

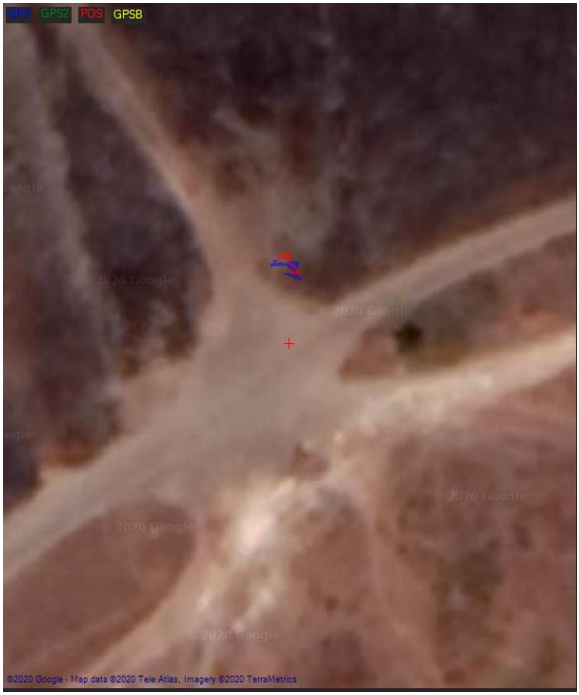


Gambar 4. 4 Ketinggian pada pengujian ke-1 tingkat kestabilan *drone*

Gambar 4. 5 Posisi pada pengujian ke-1 tingkat kestabilan *drone*



Gambar 4. 6 Ketinggian pada pengujian ke-2 tingkat kestabilan *drone*



Gambar 4. 7 Posisi pada pengujian ke-2 tingkat kestabilan *drone*

Dari dua pengujian yang dilakukan, pilot mencoba menerbangkan *drone* pada ketinggian 5 meter dan 7 meter. Pembacaan ketinggian *drone* dapat diketahui dengan memanfaatkan *telemetry* yang masing-masing terpasang pada *drone* dan *ground control station* (GCS). Pengujian dilakukan dengan mengambil data ketinggian dan posisi *drone* selama 20 detik. Ketika *drone* telah mencapai ketinggian yang diharapkan, pilot melepas kendali *radio control* seperti yang terlihat pada gambar 4.3 dan melakukan pengujian kestabilan *drone* dalam mempertahankan ketinggian dan posisinya. Dapat dilihat data grafik pada gambar 4.4 dan gambar 4.6, *drone* dapat mempertahankan ketinggiannya meskipun terdapat *error* ketinggian ± 1 meter dari ketinggian yang diharapkan. Kemudian dari gambar 4.5 dan gambar 4.7 terlihat bahwa *drone* mampu mempertahankan posisinya pada saat di udara.

4.3. Pengujian Pendeteksian Kabel SUTM

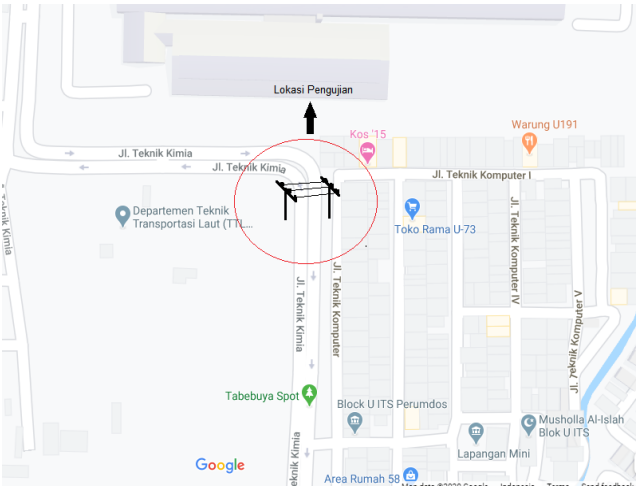
Pengujian ini dilakukan untuk mengetahui keandalan metode yang diusulkan dalam mendeteksi kabel Saluran Udara Tegangan Menengah (SUTM) dilingkungan yang terdapat banyak *noise* seperti tepi jalan, pohon, rerumputan dan objek lainnya. Pengujian dilakukan secara *online*, maksudnya yaitu *drone* diterbangkan secara manual ke atas kabel SUTM kemudian dilakukan pengujian program pendeteksian kabel yang telah dibuat secara langsung. Lokasi pengujian dilakukan pada dua tempat yang berbeda, yaitu di Jl. Teknik Kimia depan perumahan dosen blok U ITS dan di Jl. Raya Keputih Permai depan masjid Thaybah perumahan Sukolilo Park Regency.

Di Jl. Teknik Kimia ITS, lingkungan pengujian minim akan garis vertikal selain kabel SUTM, sedangkan di Jl. Raya Keputih Permai terdapat banyak garis vertikal selain kabel SUTM, seperti kabel Saluran Udara Tegangan Rendah (SUTR) yang terdapat di bawah kabel SUTM, kabel telkom dan tepi jalan yang sejajar dengan kabel SUTM. Lokasi pengujian ditunjukkan pada gambar 4.8 dan 4.9.

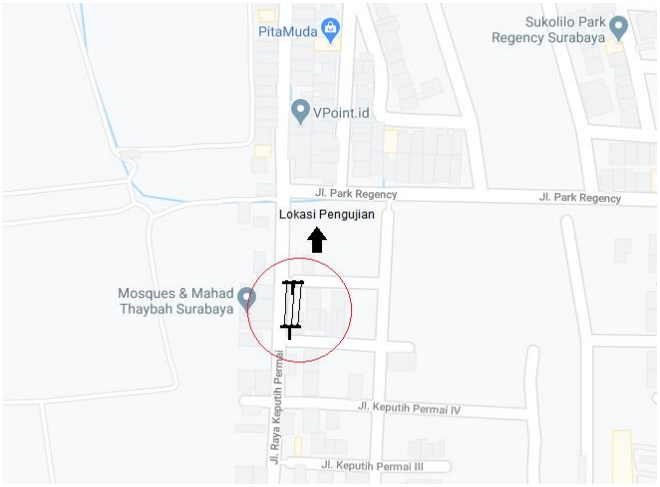
Sebelum dilakukan pengujian, perlu dilakukan penentuan nilai dari ukuran kernel pada filter yang digunakan pada bagian ekstrak garis vertikal, sehingga informasi yang didapatkan hanya berupa kabel Saluran Udara Tegangan Menengah (SUTM). Ukuran kernel yang digunakan adalah sebagai berikut.

Structuring Element (kernel 1x10) → *Erode (Kernel 1x10)*

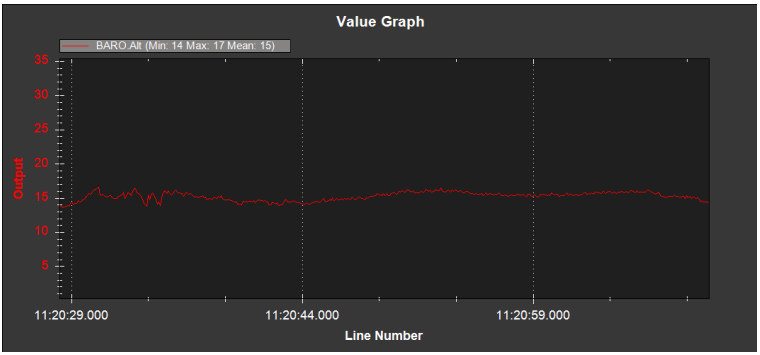
Untuk nilai *structuring element* dan *erode* masing-masing ditentukan nilai kernel berukuran 1x10, sehingga informasi yang diekstrak dari *frame* hanya objek yang berbentuk vertikal. Sedangkan objek yang berbentuk horizontal seperti tepi jalan tidak dilewatkan.



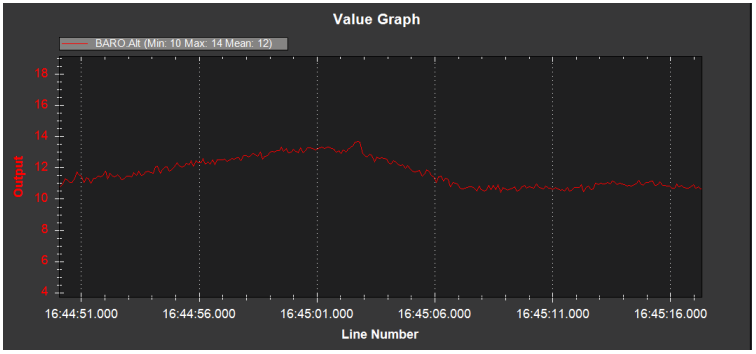
Gambar 4. 8 Lokasi pengujian 1 di Jl. Teknik Kimia ITS



Gambar 4. 9 Lokasi pengujian 2 di Jl. Raya Keputih Permai



Gambar 4. 10 Ketinggian pengujian deteksi kabel di Jl. Teknik Kimia ITS



Gambar 4. 11 Ketinggian pengujian deteksi kabel di Jl. Raya Keputih Permai

Pada pengujian pertama di Jl. Teknik Kimia ITS, kabel Saluran Udara Tegangan Menengah (SUTM) memiliki ketinggian 8 m di atas tanah. Pengukuran ketinggian kabel SUTM dilakukan dengan memanfaatkan sensor barometer yang terdapat pada *flight controller* pixhawk. Pengukuran dilakukan dengan menerbangkan *drone* pada ketinggian yang sama dengan kabel SUTM, kemudian pembacaan ketinggian *drone* dilihat pada *ground control station* (GCS).

Pengujian pendeteksian kabel SUTM di Jl. Teknik Kimia ITS dilakukan dengan menerbangkan *drone* di atas kabel SUTM dengan ketinggian $\pm$ 16 m di atas permukaan tanah. Pada pengujian tersebut,

pendeteksian kabel dapat dilakukan dengan baik, dimana objek yang terdeteksi hanya berupa kabel SUTM. Sedangkan *noise* disekitar gambar, seperti tepi jalan, pohon, rerumputan dan lain-lain tidak terdeteksi. Hasil pengujian deteksi kabel SUTM dapat dilihat pada gambar 4.12.

Gambar 4. 12 Hasil deteksi kabel SUTM di Jl. Teknik Kimia ITS

Pada pengujian kedua di Jl. Raya Keputih Permai, pengujian dilakukan dengan dengan menerbangkan *drone* di atas kabel Saluran Udara Tegangan Menengah (SUTM) dengan ketinggian 12 m di atas permukaan tanah. Pada pegujian ini, pendeteksian kabel dapat dilakukan dengan baik, karena ketiga kabel SUTM dapat terdeteksi semuanya. Berikut merupakan hasil pendeteksian kabel pada pengujian kedua.

Gambar 4. 13 Hasil deteksi kabel SUTM di Jl. Raya Keputih Permai

Namun pada pengujian kedua terdapat beberapa kesalahan yang dilakukan ketika pengujian pendeteksian kabel dilakukan, dimana beberapa objek seperti kabel Saluran Udara Tegangan Rendah (SUTR) yang berada di bawah kabel SUTM dan tepi jalan sesekali ikut terdeteksi. Hal ini disebabkan karena objek tersebut memiliki bentuk vertikal, sehingga informasinya dilewatkan dan ikut terdeteksi. Berikut ini merupakan kesalahan deteksi ketika pengujian dilakukan.

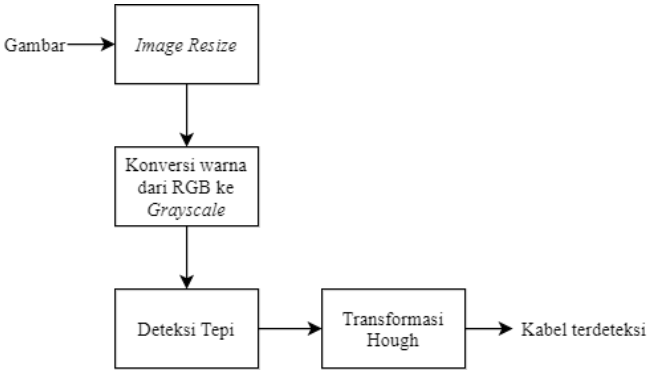
Gambar 4. 14 Kesalahan deteksi pertama ketika pengujian dilakukan

Gambar 4. 15 Kesalahan deteksi kedua ketika pengujian dilakukan

Dari hasil pengujian dapat dikatakan bahwa metode yang diusulkan cukup baik dalam mendeteksi kabel Saluran Udara Tegangan Menengah (SUTM) karena mampu mendeteksi tiga kabel yang ada. Hanya saja metode ini tidak efektif ketika pengujian dilakukan pada lingkungan yang terdapat banyak objek vertikal.

4.3.1. Perbandingan dengan Metode Konvensional

Berdasarkan hasil pendeteksian kabel Saluran Udara Tegangan Menengah (SUTM) yang dilakukan secara *online* dengan menggunakan metode yang diusulkan, akan dilakukan perbandingan dengan metode konvensional. Metode konvensional adalah metode yang paling umum digunakan dalam mendeteksi saluran listrik dimana tahapannya terdiri dari *edge detection* dan transformasi Hough [21,32]. Perbandingan dilakukan secara *offline*, dimana video gambar kabel Saluran Udara Tegangan Menengah (SUTM) yang telah diambil dari *drone* disimulasikan menggunakan python pada *software* pycharm. Perbandingan dilakukan terhadap 2 pengujian yang telah dilakukan sebelumnya secara *online*. Tujuan dilakukan perbandingan metode adalah untuk mengetahui keandalan metode yang diusulkan dalam mendeteksi kabel SUTM. Berikut ini merupakan blok diagram dari metode deteksi garis secara konvensional.



Gambar 4. 16 Blok diagram metode deteksi garis konvensional

Perbedaan antara metode konvensional dengan metode yang diusulkan adalah pada metode konvensional tidak terdapat tahap ekstraksi garis vertikal, sehingga semua informasi yang terdapat dalam gambar

dilewatkan. Sedangkan pada metode yang diusulkan, informasi yang diizinkan untuk lewat hanya berupa objek vertikal.

Dari hasil simulasi yang dilakukan secara *offline*, metode konvensional sering melakukan kesalahan deteksi. Pada pengujian ke-1 yang dilakukan di Jl. Teknik Kimia ITS, metode konvensional ikut mendeteksi tepi jalan yang posisinya horizontal di *frame*, dapat dilihat pada gambar 4.17. Pada pengujian ke-2 yang dilakukan di Jl. Raya Keputih Permai, metode konvensional juga ikut mendeteksi tepi jalan yang posisinya horizontal dan vertikal di *frame* seperti yang terlihat pada gambar 4.18. Selain itu, kabel SUTM yang terdapat pada *frame* juga sesekali tidak terdeteksi dengan menggunakan metode konvensional.

Kemudian dilakukan perhitungan akurasi dari masing-masing metode. Kesalahan objek yang terdeteksi seperti tepi jalan, kabel telkom dan kabel Saluran Udara Tegangan Rendah (SUTR) yang terletak di bawah kabel SUTM dihitung jumlahnya. Dan juga kabel SUTM yang tidak terdeteksi pada *frame* dihitung jumlahnya, baik metode konvensional maupun metode yang diusulkan untuk mengetahui akurasi dari masing-masing metode. Perhitungan akurasi dilakukan dengan menggunakan rumus sebagai berikut.

Akurasi = (jumlah *frame* – (jumlah *frame* saat kabel tidak terdeteksi+kesalahan deteksi)) ÷ jumlah *frame* (4.1)

Jumlah *frame* pada video yang diambil oleh kamera dikurangi dengan jumlah *frame* saat kabel tidak terdeteksi ditambah dengan jumlah *frame* saat kesalahan deteksi dilakukan memberikan informasi kabel yang berhasil dideteksi oleh sistem. Kemudian kabel yang berhasil dideteksi tersebut dibandingkan nilainya dengan jumlah *frame* sehingga memberikan nilai akurasi sistem dalam mendeteksi kabel SUTM. Perbandingan akurasi antara metode yang diusulkan dengan metode konvensional dapat dilihat pada tabel 4.2. Dari tabel 4.2, pada pengujian ke-2 di Jl. Raya Keputih Permai terdapat banyak kesalahan deteksi dikarenakan pada lokasi pengujian tersebut terdapat banyak garis vertikal selain kabel SUTM, seperti kabel Saluran Udara Tegangan Rendah (SUTR), kabel Telkom dan tepi jalan yang vertikal. Sehingga yang terdeteksi ketika pengujian dilakukan yaitu bukan hanya kabel SUTM, tetapi kabel SUTR, kabel Telkom dan tepi jalan sesekali juga ikut terdeteksi. Sedangkan pada lokasi pengujian ke-1 minim akan garis vertikal selain kabel SUTM.

Gambar 4. 17. Kesalahan deteksi pengujian ke-1 di Jl. Teknik Kimia ITS menggunakan metode konvensional

Gambar 4. 18. Kesalahan deteksi pengujian ke-1 di Jl. Teknik Kimia ITS menggunakan metode konvensional

Tabel 4. 2 Hasil perbandingan metode yang diusulkan dengan metode konvensional

Pengujian secara *offline*	Metode	Jumlah *frame*	Jumlah *frame* saat kabel tidak terdeteksi + kesalahan deteksi	Akurasi
Ke-1 di Jl. Teknik Kimia ITS (sedikit garis vertikal)	konvensional	940	93 + 125	76,80%
	yang diusulkan	940	8 + 0	99,14%
Ke-2 di Jl. Raya Keputih (banyak garis vertikal)	konvensional	535	12 + 258	49,53%
	yang diusulkan	535	0 + 227	57,57%
Rata-rata metode konvensional				62,97%
Rata-rata metode yang diusulkan				78,35%

Dapat dilihat pada tabel 4.2 bahwa metode yang diusulkan lebih unggul dalam mendeteksi kabel SUTM dengan akurasi sebesar 78,35% dibandingkan dengan metode konvensional dengan akurasi sebesar 62,97%.

4.4. Pengujian Keseluruhan Sistem

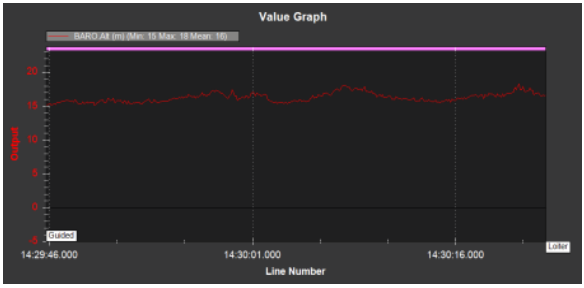
Pengujian ini dilakukan untuk mengetahui keseluruhan sistem yang telah dibuat. Informasi posisi kabel Saluran Udara Tegangan Menengah (SUTM) pada *frame* yang diperoleh dari hasil pendeteksian kabel menggunakan visi komputer akan digunakan oleh kontrol PID *drone* sebagai navigasi *drone* dalam terbang menjejaki kabel SUTM secara mandiri. Pengujian dilakukan di dua lokasi yang berbeda, yaitu di Jl. Teknik Kimia depan perumahan dosen blok U ITS dan di Jl. Raya Keputih Permai depan masjid Thaybah perumahan Sukolilo Park Regency. Lokasi pengujian dapat dilihat pada gambar 4.8 dan gambar 4.9.

Pengujian dilakukan dengan menerbangkan *drone* secara manual pada mode penerbangan *loiter* ke atas kabel SUTM dan mengarahkan posisi depan *drone* sejajar dengan kabel SUTM. Kemudian ketika posisi

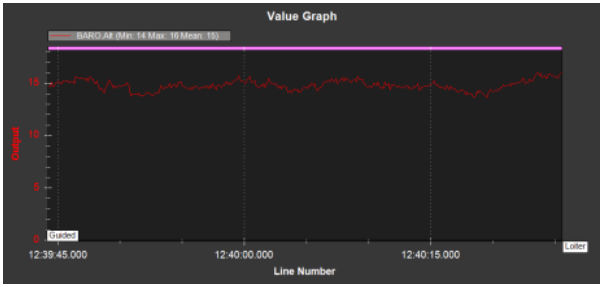
drone sudah tepat berada di atas kabel SUTM dan sejajar dengan kabel SURM, mode penerbangan akan diubah menjadi *guided* sehingga *drone* dapat melakukan misi penjejakan kabel yang telah dirancang. Pengujian dilakukan sebanyak 4 kali, dengan masing-masing lokasi dilakukan pengujian sebanyak 2 kali.

Pengujian dilakukan pada ketinggian 15 m di atas permukaan tanah. Grafik ketinggian pada pengujian ke-1, 2, 3 dan 4 dapat dilihat pada gambar 4.19, 4.20, 4.21 dan 4.22. Dapat dilihat pada grafik bahwa *drone* dapat mempertahankan ketinggiannya.

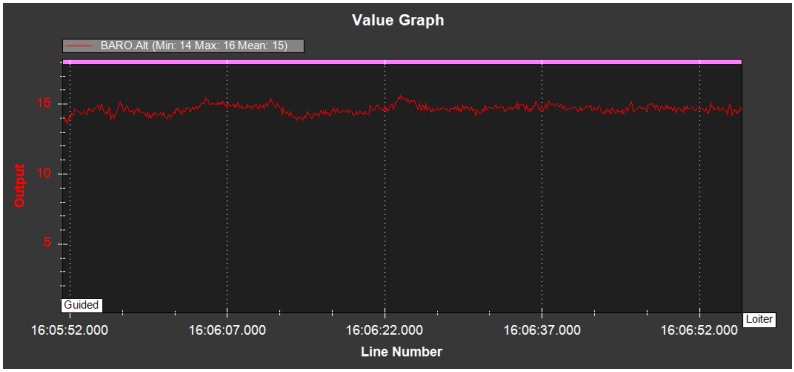
Pada gambar 4.19, 4.20, 4.21 dan 4.22 dapat dilihat bahwa ketika *drone* telah mencapai ketinggian 15 m, mode penerbangan yang semula adalah *loiter* diubah menjadi *guided*. Pada saat ini, *drone* menjalankan misi mengikuti kabel SUTM secara mandiri. *Drone* mengikuti kabel SUTM dengan jarak $\pm$ 15 m. Navigasi *drone* memanfaatkan informasi dari kabel yang terdeteksi.



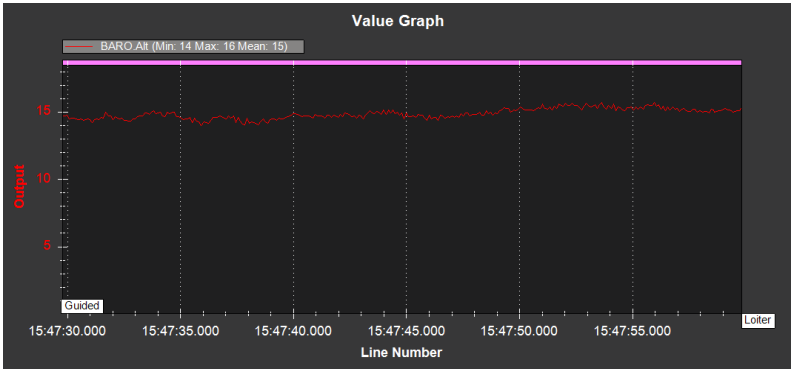
Gambar 4. 19 Ketinggian *drone* pada pengujian ke-1 di Jl. Teknik Kimia ITS



Gambar 4. 20 Ketinggian *drone* pada pengujian ke-2 di Jl. Teknik Kimia ITS



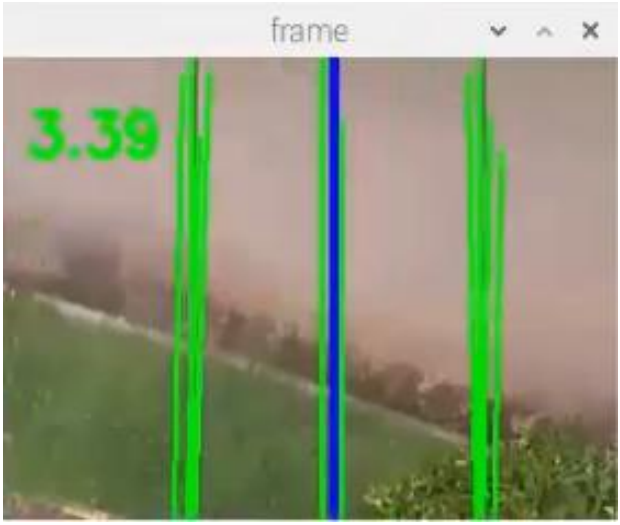
Gambar 4. 21 Ketinggian *drone* pada pengujian ke-3 di Jl. Raya Keputih Permai



Gambar 4. 22 Ketinggian *drone* pada pengujian ke-4 di Jl. Raya Keputih Permai

Drone memanfaatkan informasi dari kabel yang terdeteksi untuk menjalankan misi terbang secara mandiri menjejaki kabel SUTM. Pada gambar 4.23, 4.24, 4.25 dan 4.26 terdapat garis berwarna biru yang merupakan garis kontrol *drone* terhadap *setpoint* yang telah ditentukan, yaitu x=160. Garis kontrol ini akan dimasukkan sebagai *feedback* pada kontrol PID untuk dibandingkan nilainya dengan *setpoint* sehingga menghasilkan kecepatan akhir *roll*. Hasil pendeteksian kabel dapat dilihat pada gambar 4.23, 4.24, 4.25 dan 4.26.

Gambar 4. 23 Hasil deteksi kabel pada pengujian ke-1 di Jl. Teknik Kimia ITS



Gambar 4. 24 Hasil deteksi kabel pada pengujian ke-2 di Jl. Teknik Kimia ITS

Gambar 4. 25 Hasil deteksi kabel pada pengujian ke-3 di Jl. Raya Keputih Permai.

Gambar 4. 26 Hasil deteksi kabel pada pengujian ke-4 di Jl. Raya Keputih Permai.

Pada gambar 4.23, 4.24, 4.25 dan 4.26 dapat dilihat bahwa kabel SUTM terdeteksi dengan baik, dimana ketiga kabel SUTM terdeteksi pada *frame*. Namun ketika pengujian dilakukan, tidak selamanya pendeteksian kabel berjalan dengan baik. Terkadang, objek vertikal di sekitar lingkungan pengujian seperti tepi jalan dan kabel Saluran Udara Tegangan Rendah (SUTR) sesekali juga ikut terdeteksi. Dan juga terkadang kabel tidak terdeteksi sama sekali. Apabila kabel tidak terdeteksi sama sekali, maka antisipasi yang dilakukan adalah dengan mengirimkan pesan MAVLink ke *flight controller* untuk kecepatan sumbu-z atau vz sebesar -0,05 m/s sehingga *drone* menambah ketinggiannya 5 cm/s hingga sistem dapat mendeteksi kabel kembali.

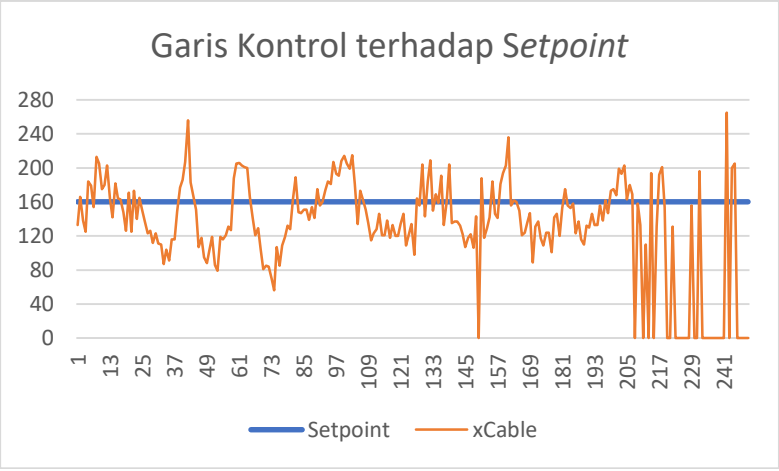
Perhitungan akurasi sistem dalam mendeteksi kabel SUTM dilakukan dengan menggunakan persamaan 4.1. Akurasi pendeteksian kabel SUTM dihitung dengan cara menjumlahkan *frame* saat kabel tidak terdeteksi dan saat kesalahan deteksi dilakukan. Kesalahan deteksi dihitung ketika garis yang terdeteksi adalah garis selain kabel SUTM. Meskipun ketika pengujian, kabel SUTM terdeteksi, namun jika ada garis lain yang ikut terdeteksi maka bisa dikatakan sistem melakukan kesalahan deteksi. Garis yang bisa saja ikut terdeteksi ketika pengujian dilakukan adalah kabel Saluran Udara Tegangan Rendah (SUTR), kabel Telkom dan tepi jalan yang vertikal di *frame*. Berikut ini merupakan tabel akurasi pendeteksian kabel SUTM.

Tabel 4. 3 Akurasi pendeteksian kabel SUTM

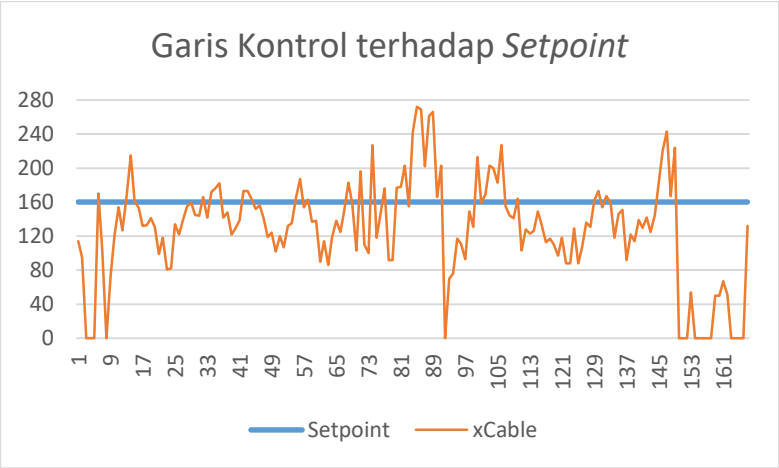
Pengujian	Jumlah Frame	Jumlah frame saat kabel tidak terdeteksi + kesalahan deteksi	Akurasi
Ke-1 di Jl. Teknik Kimia ITS	250	30 + 3	86,8%
Ke-2 di Jl. Teknik Kimia ITS	167	17 + 0	89,82%
Ke-3 di Jl. Raya Keputih Permai	272	0 + 105	61,39%
Ke-4 di Jl. Raya Keputih Permai	236	0 + 67	71,61%
Rata-rata akurasi			77,40%

Pada pengujian di Jl. Teknik Kimia ITS, dari dua pengujian yang dilakukan, terdapat kabel yang tidak terdeteksi dan kesalahan deteksi masing-masing sebanyak 33 dan 17 *frame.* Sedangkan pada pengujian di Jl. Raya Keputih Permai, dari dua pengujian yang dilakukan, semua kabel dapat terdeteksi, namun terdapat kesalahan deteksi, yaitu kabel Saluran Udara Tegangan Rendah (SUTR), kabel Telkom dan tepi jalan sesekali ikut terdeteksi. Dari dua pengujian tersebut, masing-masing kesalahan deteksi berjumlah 105 dan 67 *frame*. Sehingga hasil akhirnya dapat diperoleh nilai rata-rata dari akurasi pendeteksian kabel SUTM dengan metode yang diusulkan sebesar 77,40%.

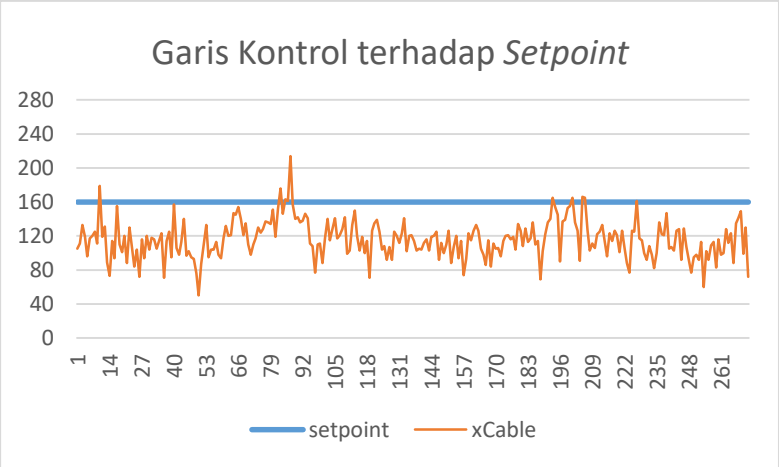
Pada gambar 4.23, 4.24, 4.25, 4.26 terdapat garis kontrol yang diberi warna biru. Garis kontrol ini merupakan nilai rata-rata koordinat x dari semua garis yang terdeteksi pada *frame*. Pada pengujian yang dilakukan di Jl. Teknik Kimia ITS, garis kontrol sesekali tidak terdeteksi karena tidak adanya garis yang terdeteksi pada *frame*. Sedangkan pada pengujian yang dilakukan di Jl. Raya Keputih Permai, garis kontrol selalu ada karena kabel selalu terdeteksi pada *frame*. Pada gambar 4.27, 4.28, 4.29 dan 4.30 merupakan grafik garis kontrol terhadap *setpoint* pada *frame* x=160. Dari gambar dapat dilihat bahwa grafik naik turun diantara *setpoint* yang telah ditentukan. Sehingga dapat dikatakan bahwa *drone* mengoreksi pergerakannya terhadap *setpoint* yang telah ditentukan pada *frame*.



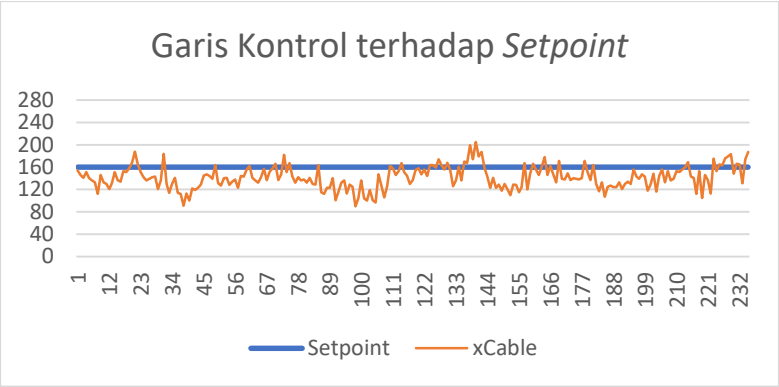
Gambar 4. 27 Grafik garis kontrol terhadap *setpoint* pengujian ke-1



Gambar 4. 28 Grafik garis kontrol terhadap *setpoint* pengujian ke-2

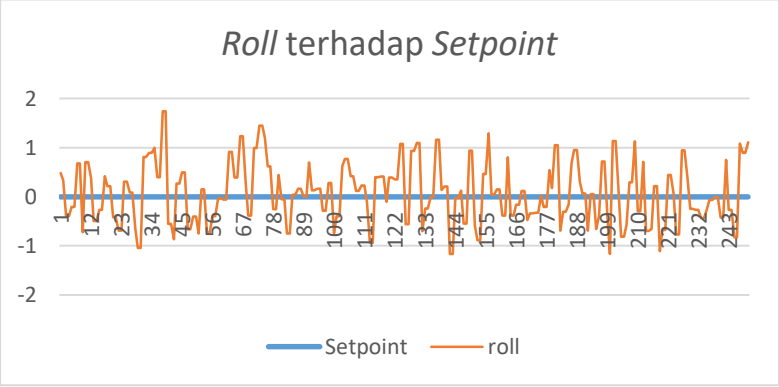


Gambar 4. 29 Grafik garis kontrol terhadap *setpoint* pengujian ke-3

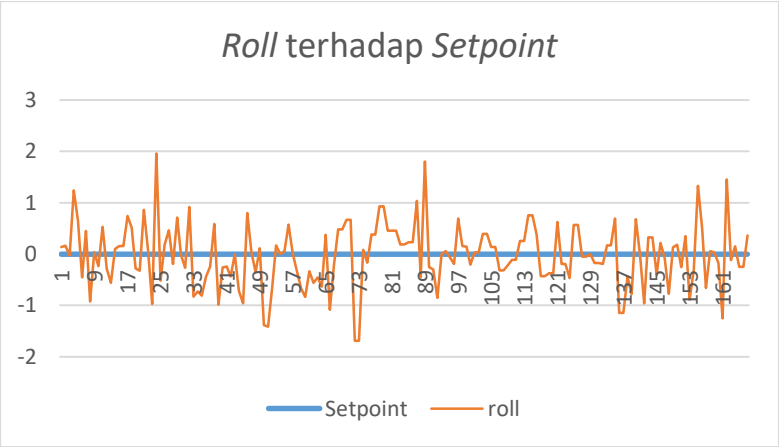


Gambar 4. 30 Grafik garis kontrol terhadap *setpoint* pengujian ke-4

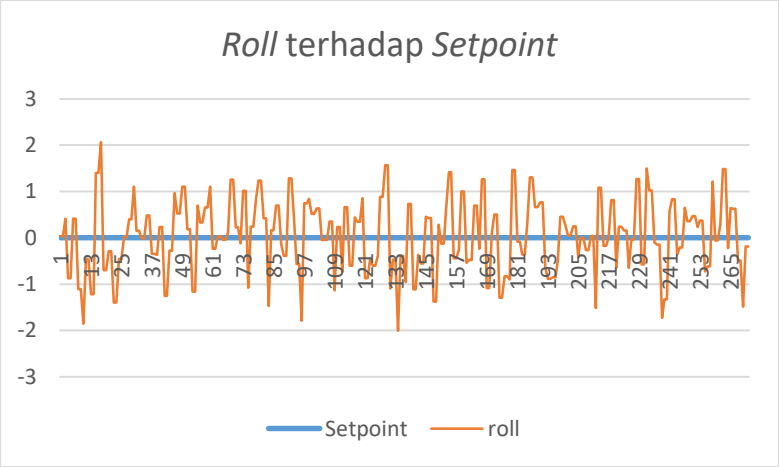
Garis kontrol yang terdeteksi akan digunakan sebagai *feedback* pada kontrol PID. Nilainya akan dibandingkan dengan nilai *setpoint* yang telah ditentukan pada *frame*. Perbandingan antara garis kontrol dengan *setpoint* akan menghasilkan nilai *error* untuk selanjutnya diberikan sebagai masukan pada P, I dan D. Nilai konstanta P, I dan D diperoleh melalui hasil *trial and error*. Dari hasil kalkulasi P, I, dan D akan diteruskan ke motor sehingga menghasilkan kecepatan *roll* akhir. Apabila kecepatan *roll* bernilai 0, maka bisa dikatakan *drone* berada tepat pada tengah *frame*, yaitu x = 160. Grafik *roll* terhadap *setpoint* dapat ditunjukkan pada gambar berikut ini.



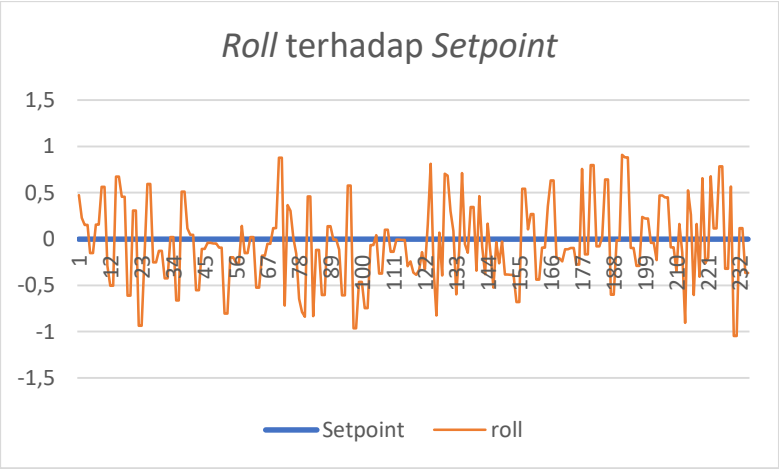
Gambar 4. 31 Grafik *roll* terhadap *setpoint* pada pengujian ke-1



Gambar 4. 32 Grafik *roll* terhadap *setpoint* pada pengujian ke-2



Gambar 4. 33 Grafik *roll* terhadap *setpoint* pada pengujian ke-3



Gambar 4. 34 Grafik *roll* terhadap *setpoint* pada pengujian ke-4

Pada grafik yang ditampilkan di atas dapat dilihat bahwa kecepatan *roll* naik turun diantara *setpoint*. Sehingga dapat dikatakan bahwa *drone* melakukan koreksi pergerakan terhadap *setpoint* yang telah ditentukan pada *frame*.

4.4.1. Pengujian Pengaruh Kecepatan

Pengujian ini dilakukan untuk mengetahui pengaruh kecepatan terhadap pendeteksian kabel SUTM ketika *drone* sedang melakukan penjejakan kabel SUTM. Pengujian dilakukan secara *online*, maksudnya adalah *drone* diterbangkan oleh pilot secara manual ke atas kabel SUTM pada mode *loiter* dan pilot mengarahkan posisi depan *drone* sejajar dengan kabel SUTM. Kemudian apabila posisi *drone* sudah tepat di atas kabel SUTM dan sejajar dengan kabel SUTM, mode penerbangan akan diubah menjadi *guided*. Pada mode penerbangan tersebut *drone* akan melakukan penjejakan kabel SUTM sesuai dengan sistem yang telah dirancang.

Pengujian dilakukan di 2 lokasi yang berbeda, yaitu di Jl. Teknik Kimia depan perumahan dosen blok U ITS dimana lingkungannya terdapat sedikit garis vertikal dan di Jl. Raya Keputih Permai depan masjid Thaybah perumahan Sukolilo Park Regency dimana . Lokasi pengujian dapat dilihat pada gambar 4.8 dan 4.9.

Kecepatan yang akan diubah-ubah nilainya adalah kecepatan sumbu-x, yaitu kecepatan vx. Sedangkan nilai vz dibuat konstan, yaitu sama dengan 0 sehingga *drone* mempertahankan ketinggiannya. Dan kecepatan vy akan diperoleh dari kontrol PID *drone*.

Kecepatan vx yang akan diatur dimulai dari 0,2 m/s, 0,4 m/s, 0,6 m/s dan 0,8 m/s. Perhitungan akurasi sistem dalam mendeteksi kabel SUTM menggunakan persamaan 4.1. Dari hasil pengujian diperoleh hasilnya seperti yang terlihat pada tabel

Tabel 4. 4 Hasil pengujian pengaruh kecepatan di Jl. Teknik Kimia ITS (lingkungan yang sedikit garis vertikal)

Kecepatan	Jumlah *frame*	Jumlah frame saat kabel tidak terdeteksi + kesalahan deteksi	Akurasi
vx = 0,2 m/s	321	31 + 0	90,34%
vx = 0,4 m/s	141	12 + 0	90,08%
vx = 0,6 m/s	73	8 + 0	89,04%
vx = 0,8 m/s	60	7 + 0	88,33%

Tabel 4. 5 Hasil pengujian pengaruh kecepatan di Jl. Raya Keputih Permai (lingkungan yang terdapat banyak garis vertikal)

Kecepatan	Jumlah *frame*	Jumlah *frame* saat kabel tidak terdeteksi + kesalahan deteksi	Akurasi
vx = 0,2 m/s	412	0 + 107	74,03%
vx = 0,4 m/s	272	0 + 71	73,89%
vx = 0,6 m/s	84	0 + 22	73,80%
vx = 0,8 m/s	76	0 + 20	73,68%

Dari tabel 4.4 dan 4.5 yang merupakan hasil dari pengujian penjejakan kabel SUTM yang dilakukan di dua lokasi yang berbeda dapat dilihat bahwa akurasi sistem dalam mendeteksi kabel SUTM menurun ketika kecepatan drone ditingkatkan. Pada tabel 4.5 dapat dilihat bahwa sistem sering melakukan kesalahan deteksi karena lingkungan pengujian terdapat banyak garis vertikal seperti kabel Saluran Udara Tegangan Rendah (SUTR), kabel Telkom dan tepi jalan. Sehingga dari dua

pengujian yang dilakukan dapat dikatakan bahwa kecepatan mempengaruhi akurasi sistem dalam mendeteksi kabel SUTM ketika *drone* sedang melakukan penjejakan kabel SUTM.

[Halaman ini sengaja dikosongkan]

BAB 5
KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil dan analisa data yang diperoleh, dapat disimpulkan bahwa:

1. Metode yang diusulkan yaitu dengan mengekstrak garis vertikal pada *frame* dapat dengan baik melakukan deteksi kabel SUTM, dimana ketiga kabel SUTM terdeteksi pada *frame*
2. *Noise* selain bentuk vertikal seperti pohon, rerumputan, kendaraan dan objek lainnya dapat dihilangkan dengan baik menggunakan metode ekstraksi garis vertikal
3. Metode ekstraksi garis vertikal mempunyai akurasi lebih tinggi dalam mendeteksi kabel SUTM dengan akurasi sebesar 78,35% dibandingkan dengan metode konvensional dengan akurasi sebesar 62,97%
4. Rata-rata akurasi pendeteksian kabel SUTM yang diperoleh ketika *drone* melakukan penjejakan kabel SUTM secara mandiri adalah sebesar 77,40%.
5. Kecepatan mempengaruhi sistem dalam mendeteksi kabel SUTM, dimana seiring meningkatknya kecepatan drone membuat akurasi sistem dalam mendeteksi kabel SUTM semakin menurun.

5.2. Saran

Perlu dilakukan pengembangan lebih lanjut terhadap sistem pendeteksian kabel SUTM agar objek yang terdeteksi pada *frame* hanya berupa kabel SUTM, sedangkan objek linier lainnya tidak ikut terdeteksi. Dalam membedakan kabel SUTM dengan objek linier lainnya dapat menggunakan metode *clustering K-means* setelah tahap deteksi garis menggunakan transformasi Hough [27]. Sehingga kendali *drone* dapat lebih baik dalam menjejaki kabel SUTM .

[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

[1] SK DIR 0017.E/DIR/2014, "Metode Pemeliharaan Trafo Distribusi Berbasis Kaidah Manajemen Aset", PT PLN (Persero), 2014

[2] SK DIR 0018.E/DIR/2014, "Metode Pemeliharaan Saluran Udara Tegangan Menengah Berbasis Kaidah Manajemen Aset", PT PLN (Persero), 2014

[3] N, Wahyudi Sarimun. 2014. Buku Saku Pelayanan Teknik. Jakarta: Garamond

[4] R. Austin, Unmanned Aircraft Systems: UAVS Design, Development and Deployment. Chichester, UK: John Wiley & Sons, Ltd, 2010.

[5] Suroso, Indreswari, "Peran *Drone*/*Unmanned Aerial Vehicle* (UAV) Buatan STTKD Dalam Dunia Penerbangan", 2016, hlm.104-112.

[6] A. V. Rodrigues, R. S. Carapau, M. M. Marques, V. Lobo, dan F. Coito, "*Unmanned systems interoperability in military maritime operations: MAVLink to STANAG 4586 bridge*," dalam *OCEANS 2017 - Aberdeen*, Aberdeen, United Kingdom, Jun 2017, hlm. 1–5, doi: 10.1109/OCEANSE.2017.8084866.

[7] N. Reker, D. Troxell, D. Troy,"Universal UAV Payload Interface" 2015.

[8] J. a Marty, "*Vulnerability Analysis of the MAVLink Protocol for Command and Control of Unmanned Aircraft*" 2013.

[9] D. Robotics. (2015) Guiding and controlling copter. [Online]. Available: http://python.*drone*kit.io/guide/copter/guided mode.html

[10] A. Kaehler dan G. R. Bradski, *Learning OpenCV 3: computer vision in C++ with the OpenCV library*, First edition, Second release. Sebastopol, CA: O'Reilly Media, 2017.

[11] Danil, Christopher, "Edge Detection dengan Algoritma Canny," Medan, STMIK IBBI, pp. 197-203

[12] Kurniawati, I. D., & Kusumawardhani, A. (2017). Implementasi Algoritma Canny dalam Pengenalan Wajah menggunakan Antarmuka GUI Matlab. *Institution of Engineering and Technology*, 1-3.

[13] Syahri Muharom. (2017). Penerapan Metode Hough Line Transform untuk Mendeteksi Pintu Ruangan Menggunakan Kamera. Jurnal IPTEK. 21(2): 79-86.

[14] Usman Ahmad, "*Pengolahan Citra Digital & Teknik Pemrogramannya*", Edisi Pertama, Graha Ilmu, 2005.

[15] I. Corporation, “http://opencv.org/platforms/,” 2000. [Online]. Available: http://opencv.org/platforms/.
[16] I. Corporation, “OpenCV,” Itseez, 2000. [Online]. Available: http://opencv.org.
[17] SK DIR 606.K/DIR/2010, “Buku 5 Standar Konstruksi Jaringan Tegangan Menengah Tenaga Listrik”, PT PLN (Persero), 2010
[18] Wieczorek, I. D. A. (2017). *A power line detection algorithm to support a fine grain UAV movement guidance*.
[19] Jeong, J., Kim, J., Yoon, T. S., & Park, J. B. (2018). Drone-based power-line tracking system. *Transactions of the Korean Institute of Electrical Engineers*, *67*(6), 773-781.
[20] Koshelev, V. I., & Kozlov, D. N. (2015, June). Wire recognition in image within aerial inspection application. In *2015 4th Mediterranean Conference on Embedded Computing (MECO)* (pp. 159-162). IEEE.
[21] Zhou, G., Yuan, J., Yen, I. L., & Bastani, F. (2016, September). Robust real-time UAV based power line detection and tracking. In *2016 IEEE International Conference on Image Processing (ICIP)* (pp. 744-748). IEEE.
[22] Zhang, J., Liu, L., Wang, B., Chen, X., Wang, Q., & Zheng, T. (2012, August). High speed automatic power line detection and tracking for a UAV-based inspection. In *2012 International Conference on Industrial Control and Electronics Engineering* (pp. 266-269). IEEE.
[23] Manlangit, C., & Green, R. Automatic Power Line Detection for a UAV system.
[24] Larrauri, J. I., Sorrosal, G., & González, M. (2013, May). Automatic system for overhead power line inspection using an Unmanned Aerial Vehicle—RELIFO project. In *2013 International Conference on Unmanned Aircraft Systems (ICUAS)* (pp. 244-252). IEEE.
[25] Liu, Y., Mejias, L., & Li, Z. (2012). Fast power line detection and localization using steerable filter for active UAV guidance.
[26] Pan, C., Cao, X., & Wu, D. (2016, December). Power line detection via background noise removal. In *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)* (pp. 871-875). IEEE.
[27] Li, Z., Liu, Y., Hayward, R., Zhang, J., & Cai, J. (2008, November). Knowledge-based power line detection for UAV surveillance and

inspection systems. In *2008 23rd International Conference Image and Vision Computing New Zealand* (pp. 1-6). IEEE.

[28] Yang, T. W., Yin, H., Ruan, Q. Q., Da Han, J., Qi, J. T., Yong, Q., ... & Sun, Z. Q. (2012, November). Overhead power line detection from UAV video images. In *2012 19th International Conference on Mechatronics and Machine Vision in Practice (M2VIP)* (pp. 74-79). IEEE.

[29] Candamo, J., Kasturi, R., Goldgof, D., & Sarkar, S. (2009). Detection of thin lines using low-quality video from low-altitude aircraft in urban settings. *IEEE Transactions on aerospace and electronic systems*, *45*(3), 937-949.

[30] Li, Z., Liu, Y., Walker, R., Hayward, R., & Zhang, J. (2010). Towards automatic power line detection for a UAV surveillance system using pulse coupled neural filter and an improved Hough transform. *Machine Vision and Applications*, *21*(5), 677-686.

[31] Yan, G., Li, C., Zhou, G., Zhang, W., & Li, X. (2007). Automatic extraction of power lines from aerial images. *IEEE Geoscience and Remote Sensing Letters*, *4*(3), 387-391.

[32] Tian, F., Wang, Y., & Zhu, L. (2015, August). Power line recognition and tracking method for UAVs inspection. In *2015 IEEE International Conference on Information and Automation* (pp. 2136-2141). IEEE.

[33] Karakose, E. (2017, September). Performance evaluation of electrical transmission line detection and tracking algorithms based on image processing using UAV. In *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)* (pp. 1-5). IEEE.

[34] Nasseri, M. H., Moradi, H., Nasiri, S. M., & Hosseini, R. (2018, October). Power Line Detection and Tracking Using Hough Transform and Particle Filter. In *2018 6th RSI International Conference on Robotics and Mechatronics (IcRoM)* (pp. 130-134). IEEE.

[35] Yan, S., & Jin, L. (2011, March). Extra Matters Recognition of Transmission System Based on Hough Transform. In *2011 Asia-Pacific Power and Energy Engineering Conference* (pp. 1-3). IEEE.

[36] Liu, X., Hou, L., & Ju, X. (2017, December). A method for detecting power lines in UAV aerial images. In *2017 3rd IEEE International Conference on Computer and Communications (ICCC)* (pp. 2132-2136). IEEE.

[37] Valipour, S., Khandani, H., & Moradi, H. (2015, October). The design and implementation of a hotline tracking UAV. In *2015 3rd RSI International Conference on Robotics and Mechatronics (ICROM)* (pp. 252-258). IEEE.

LAMPIRAN

A. *Datasheet* Komponen

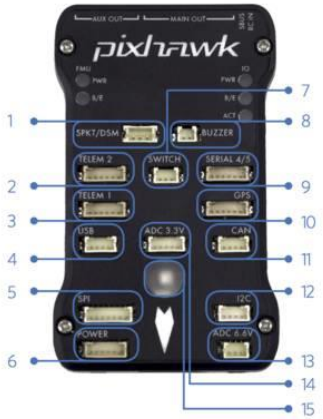
1. *Flight controller* pixhawk 2.4.8

- *Specifications*

  - **Processor**
    - 32-bit ARM Cortex M4 core with FPU
    - 168 Mhz/256 KB RAM/2 MB Flash
    - 32-bit failsafe co-processor
  - **Sensors**
    - MPU6000 as main accel and gyro
    - ST Micro 16-bit gyroscope
    - ST Micro 14-bit accelerometer/compass (magnetometer)
    - MEAS barometer
  - **Power**
    - Ideal diode controller with automatic failover
    - Servo rail high-power (7 V) and high-current ready
    - All peripheral outputs over-current protected, all inputs ESD protected
  - **Interfaces**
    - 5x UART serial ports, 1 high-power capable, 2 with HW flow control
    - Spektrum DSM/DSM2/DSM-X Satellite input
    - Futaba S.BUS input (output not yet implemented)
    - PPM sum signal
    - RSSI (PWM or voltage) input
    - I2C, SPI, 2x CAN, USB
    - 3.3V and 6.6V ADC inputs
  - **Dimensions**
    - Weight 38 g (1.3 oz)
    - Width 50 mm (2.0")
    - Height 15.5 mm (.6")
    - Length 81.5 mm (3.2")

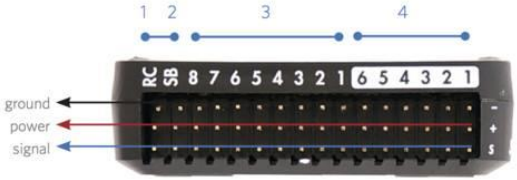
- Pixhawk *connector assignments*

1 Flight management unit (FMU) power
2 FMU bootloader mode (flashing) or error (solid)
3 Input/output unit power
4 I/O bootloader mode (flashing) or error (solid)
5 Activity (flashing indicates all units responsive)



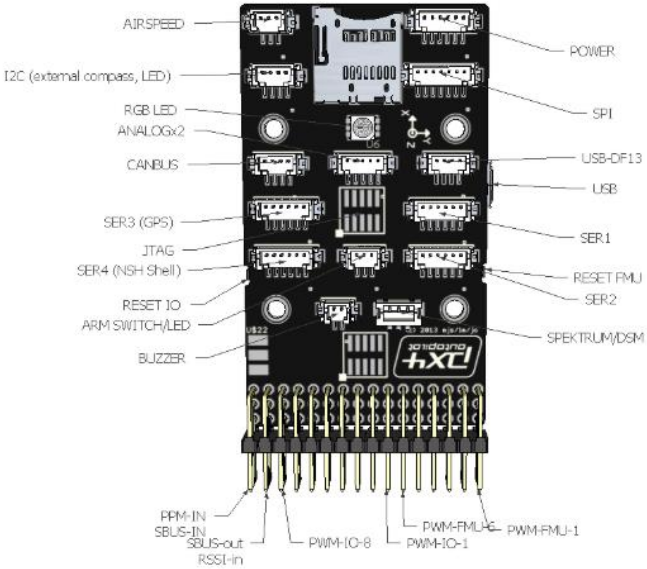
1 Spektrum DSM receiver
2 Telemetry (on-screen display)
3 Telemetry (radio telemetry)
4 USB
5 SPI (serial peripheral interface) bus
6 Power module
7 Safety switch button
8 Buzzer
9 Serial
10 GPS module
11 CAN (controller area network) bus
12 I²C splitter or compass module
13 Analog to digital converter 6.6 V
14 Analog to digital converter 3.3 V
15 LED indicator

1 Input/output reset button
2 SD card
3 Flight management reset button
4 Micro-USB port



1 Radio control receiver input
2 S.Bus output
3 Main outputs
4 Auxiliary outputs

- Pixhawk *connector diagram*



For all connectors pin 1 is on the right in the above image

Serial 1 (Telem 1) and Serial 2 (Telem 2) Pins: 6 = GND, 5 = RTS, 4 = CTS, 3 = RX, 2 = TX, 1 = 5V.

- Pixhawk *connector pin assignments*

TELEM1, TELEM2 ports

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	TX (OUT)	+3.3V
3 (blk)	RX (IN)	+3.3V
4 (blk)	CTS	+3.3V
5 (blk)	RTS	+3.3V
6 (blk)	GND	GND

GPS port

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	TX (OUT)	+3.3V
3 (blk)	RX (IN)	+3.3V
4 (blk)	CAN2 TX	+3.3V
5 (blk)	CAN2 RX	+3.3V
6 (blk)	GND	GND

SERIAL 4/5 port

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	TX (#4)	+3.3V
3 (blk)	RX (#4)	+3.3V
4 (blk)	TX (#5)	+3.3V
5 (blk)	RX (#5)	+3.3V
6 (blk)	GND	GND

POWER

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	VCC	+5V
3 (blk)	CURRENT	up to +3.3V
4 (blk)	VOLTAGE	up to +3.3V
5 (blk)	GND	GND
6 (blk)	GND	GND

I2C

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	SCL	+3.3 (pullups)
3 (blk)	SDA	+3.3 (pullups)
4 (blk)	GND	GND

CAN

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	CAN_H	+12V
3 (blk)	CAN_L	+12V
4 (blk)	GND	GND

ADC 6.6V

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	ADC IN	up to +6.6V
3 (blk)	GND	GND

SWITCH

Pin	Signal	Volt
1 (red)	VCC	+3.3V
2 (blk)	!IO_LED_SAFETY	GND
3 (blk)	SAFETY	GND

Spektrum/DSM Port

The Spektrum/DSM port is for connecting Spektrum DSM-2/DSMX receiver modules.

Pin	Signal	Volt
1 (white)	Signal	+3.3V
2 (black)	GND	GND
3 (red)	VCC	+3.3V

SPI

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	SPI_SCK	3.3V
3 (blk)	SPI_MISO	+3.3V
4 (blk)	SPI_MOSI	+3.3V
5 (blk)	!SPI_NSS	+3.3V
6 (blk)	!GPIO	+3.3V
7 (blk)	GND	GND

ADC 3.3V

Pin	Signal	Volt
1 (red)	VCC	+5V
2 (blk)	ADC IN	up to +3.3V
3 (blk)	GND	GND
4 (blk)	ADC IN	up to +3.3V
5 (blk)	GND	GND

2. Raspberry Pi 4 Model B

2 Features

2.1 Hardware

- Quad core 64-bit ARM-Cortex A72 running at 1.5GHz
- 1, 2 and 4 Gigabyte LPDDR4 RAM options
- H.265 (HEVC) hardware decode (up to 4Kp60)
- H.264 hardware decode (up to 1080p60)
- VideoCore VI 3D Graphics
- Supports dual HDMI display output up to 4Kp60

2.2 Interfaces

- 802.11 b/g/n/ac Wireless LAN
- Bluetooth 5.0 with BLE
- 1x SD Card
- 2x micro-HDMI ports supporting dual displays up to 4Kp60 resolution
- 2x USB2 ports
- 2x USB3 ports
- 1x Gigabit Ethernet port (supports PoE with add-on PoE HAT)
- 1x Raspberry Pi camera port (2-lane MIPI CSI)
- 1x Raspberry Pi display port (2-lane MIPI DSI)
- 28x user GPIO supporting various interface options:
  - Up to 6x UART
  - Up to 6x I2C
  - Up to 5x SPI

- 1x SDIO interface
- 1x DPI (Parallel RGB Display)
- 1x PCM
- Up to 2x PWM channels
- Up to 3x GPCLK outputs

4 Electrical Specification

Caution! Stresses above those listed in Table 2 may cause permanent damage to the device. This is a stress rating only; functional operation of the device under these or any other conditions above those listed in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Copyright Raspberry Pi (Trading) Ltd. 2019

Symbol	Parameter	Minimum	Maximum	Unit
VIN	5V Input Voltage	-0.5	6.0	V

Table 2: Absolute Maximum Ratings

Please note that VDD_IO is the GPIO bank voltage which is tied to the on-board 3.3V supply rail.

Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Unit
V_{IL}	Input low voltage[a]	VDD_IO = 3.3V	-	-	TBD	V
V_{IH}	Input high voltage[a]	VDD_IO = 3.3V	TBD	-	-	V
I_{IL}	Input leakage current	TA = +85°C	-	-	TBD	µA
C_{IN}	Input capacitance	-	-	TBD	-	pF
V_{OL}	Output low voltage[b]	VDD_IO = 3.3V, IOL = -2mA	-	-	TBD	V
V_{OH}	Output high voltage[b]	VDD_IO = 3.3V, IOH = 2mA	TBD	-	-	V
I_{OL}	Output low current[c]	VDD_IO = 3.3V, VO = 0.4V	TBD	-	-	mA
I_{OH}	Output high current[c]	VDD_IO = 3.3V, VO = 2.3V	TBD	-	-	mA
R_{PU}	Pullup resistor	-	TBD	-	TBD	kΩ
R_{PD}	Pulldown resistor	-	TBD	-	TBD	kΩ

[a] Hysteresis enabled
[b] Default drive strength (8mA)
[c] Maximum drive strength (16mA)

Table 3: DC Characteristics

Pin Name	Symbol	Parameter	Minimum	Typical	Maximum	Unit
Digital outputs	t_{rise}	10-90% rise time[a]	-	TBD	-	ns
Digital outputs	t_{fall}	90-10% fall time[a]	-	TBD	-	ns

[a] Default drive strength, CL = 5pF, VDD_IO = 3.3V

Table 4: Digital I/O Pin AC Characteristics

Figure 2: Digital IO Characteristics

4.1 Power Requirements

The Pi4B requires a good quality USB-C power supply capable of delivering 5V at 3A. If attached downstream USB devices consume less than 500mA, a 5V, 2.5A supply may be used.

5 Peripherals

5.1 GPIO Interface

The Pi4B makes 28 BCM2711 GPIOs available via a standard Raspberry Pi 40-pin header. This header is backwards compatible with all previous Raspberry Pi boards with a 40-way header.

5.1.1 GPIO Pin Assignments

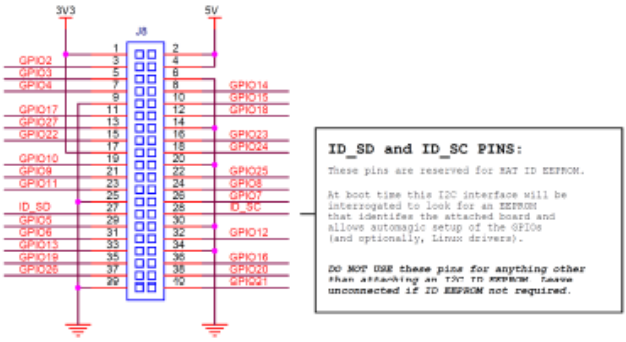


Figure 3: GPIO Connector Pinout

5.1.2 GPIO Alternate Functions

GPIO	Default Pull	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5
0	High	SDA0	SA5	PCLK	SPI3_CE0_N	TXD2	SDA6
1	High	SCL0	SA4	DE	SPI3_MISO	RXD2	SCL6
2	High	SDA1	SA3	LCD_VSYNC	SPI3_MOSI	CTS2	SDA3
3	High	SCL1	SA2	LCD_HSYNC	SPI3_SCLK	RTS2	SCL3
4	High	GPCLK0	SA1	DPI_D0	SPI4_CE0_N	TXD3	SDA3
5	High	GPCLK1	SA0	DPI_D1	SPI4_MISO	RXD3	SCL3
6	High	GPCLK2	SOE_N	DPI_D2	SPI4_MOSI	CTS3	SDA4
7	High	SPI0_CE1_N	SWE_N	DPI_D3	SPI4_SCLK	RTS3	SCL4
8	High	SPI0_CE0_N	SD0	DPI_D4	-	TXD4	SDA4
9	Low	SPI0_MISO	SD1	DPI_D5	-	RXD4	SCL4
10	Low	SPI0_MOSI	SD2	DPI_D6	-	CTS4	SDA5
11	Low	SPI0_SCLK	SD3	DPI_D7	-	RTS4	SCL5
12	Low	PWM0	SD4	DPI_D8	SPI5_CE0_N	TXD5	SDA5
13	Low	PWM1	SD5	DPI_D9	SPI5_MISO	RXD5	SCL5
14	Low	TXD0	SD6	DPI_D10	SPI5_MOSI	CTS5	TXD1
15	Low	RXD0	SD7	DPI_D11	SPI5_SCLK	RTS5	RXD1
16	Low	FL0	SD8	DPI_D12	CTS0	SPI1_CE2_N	CTS1
17	Low	FL1	SD9	DPI_D13	RTS0	SPI1_CE1_N	RTS1
18	Low	PCM_CLK	SD10	DPI_D14	SPI6_CE0_N	SPI1_CE0_N	PWM0
19	Low	PCM_FS	SD11	DPI_D15	SPI6_MISO	SPI1_MISO	PWM1
20	Low	PCM_DIN	SD12	DPI_D16	SPI6_MOSI	SPI1_MOSI	GPCLK0
21	Low	PCM_DOUT	SD13	DPI_D17	SPI6_SCLK	SPI1_SCLK	GPCLK1
22	Low	SD0_CLK	SD14	DPI_D18	SD1_CLK	ARM_TRST	SDA6
23	Low	SD0_CMD	SD15	DPI_D19	SD1_CMD	ARM_RTCK	SCL6
24	Low	SD0_DAT0	SD16	DPI_D20	SD1_DAT0	ARM_TDO	SPI3_CE1_N
25	Low	SD0_DAT1	SD17	DPI_D21	SD1_DAT1	ARM_TCK	SPI4_CE1_N
26	Low	SD0_DAT2	TE0	DPI_D22	SD1_DAT2	ARM_TDI	SPI5_CE1_N
27	Low	SD0_DAT3	TE1	DPI_D23	SD1_DAT3	ARM_TMS	SPI6_CE1_N

Table 5: Raspberry Pi 4 GPIO Alternate Functions

Table 5 details the default pin pull state and available alternate GPIO functions. Most of these alternate peripheral functions are described in detail in the BCM2711 Peripherals Specification document which can be downloaded from the hardware documentation section of the website.

3. Pi Camera V1.3

Hardware specification

	Camera Module v1	Camera Module v2	HQ Camera
Net price	$25	$25	$50
Size	Around 25 × 24 × 9 mm		38 x 38 x 18.4mm (excluding lens)
Weight	3g	3g	
Still resolution	5 Megapixels	8 Megapixels	12.3 Megapixels
Video modes	1080p30, 720p60 and 640 × 480p60/90	1080p30, 720p60 and 640 × 480p60/90	1080p30, 720p60 and 640 × 480p60/90
Linux integration	V4L2 driver available	V4L2 driver available	V4L2 driver available
C programming API	OpenMAX IL and others available	OpenMAX IL and others available	
Sensor	OmniVision OV5647	Sony IMX219	Sony IMX477
Sensor resolution	2592 × 1944 pixels	3280 × 2464 pixels	4056 x 3040 pixels
Sensor image area	3.76 × 2.74 mm	3.68 x 2.76 mm (4.6 mm diagonal)	6.287mm x 4.712 mm (7.9mm diagonal)
Pixel size	1.4 µm × 1.4 µm	1.12 µm x 1.12 µm	1.55 µm x 1.55 µm
Optical size	1/4"	1/4"	
Full-frame SLR lens equivalent	35 mm		
S/N ratio	36 dB		
Dynamic range	67 dB @ 8x gain		
Sensitivity	680 mV/lux-sec		
Dark current	16 mV/sec @ 60 C		
Well capacity	4.3 Ke-		
Fixed focus	1 m to infinity		N/A
Focal length	3.60 mm +/- 0.01	3.04 mm	Depends on lens
Horizontal field of view	53.50 +/- 0.13 degrees	62.2 degrees	Depends on lens
Vertical field of view	41.41 +/- 0.11 degrees	48.8 degrees	Depends on lens
Focal ratio (F-Stop)	2.9	2.0	Depends on lens

Hardware features

Available	Implemented
Chief ray angle correction	Yes
Global and rolling shutter	Rolling shutter
Automatic exposure control (AEC)	No - done by ISP instead
Automatic white balance (AWB)	No - done by ISP instead
Automatic black level calibration (ABLC)	No - done by ISP instead
Automatic 50/60 Hz luminance detection	No - done by ISP instead
Frame rate up to 120 fps	Max 90fps. Limitations on frame size for the higher frame rates (VGA only for above 47fps)
AEC/AGC 16-zone size/position /weight control	No - done by ISP instead
Mirror and flip	Yes
Cropping	No - done by ISP instead (except 1080p mode)
Lens correction	No - done by ISP instead
Defective pixel cancelling	No - done by ISP instead
10-bit RAW RGB data	Yes - format conversions available via GPU
Support for LED and flash strobe mode	LED flash
Support for internal and external frame synchronisation for frame exposure mode	No
Support for 2 × 2 binning for better SNR in low light conditions	Anything output res below 1296 x 976 will use the 2 x 2 binned mode
Support for horizontal and vertical sub-sampling	Yes, via binning and skipping
On-chip phase lock loop (PLL)	Yes
Standard serial SCCB interface	Yes
Digital video port (DVP) parallel output interface	No
MIPI interface (two lanes)	Yes
32 bytes of embedded one-time programmable (OTP) memory	No
Embedded 1.5V regulator for core power	Yes

4. GPS Ublox M8N

Deskripsi :
- Receiver type 72-channel M8 engine
- GPS/QZSS L1 C/A, GLONASS L10F, BeiDou B1
- SBAS L1 C/A: WAAS, EGNOS, MSAS
- Galileo-ready E1B/C (M8N)
- Nav. update rate1 Single GNSS: up to 18 HZ
- Concurrent GNSS: up to 10 Hz
- Position accuracy2 2.0 m CEP
- Acquisition2 Cold starts: 26 s
- Aided starts: 2 s
- Reacquisition: 1.5 s
- Sensitivity2 Tracking & Nav: 167 dBm
- Cold starts: 148 dBm
- Hot starts: 156 dBm
- Assistance AssistNow GNSS Online
- AssistNow GNSS Offline (up to 35 days)3
- AssistNow Autonomous (up to 6 days)
- OMA SUPL & 3GPP compliant
- Oscillator TCXO (M8N/Q),
- Crystal (M8M)
- RTC crystal Built-In
- Noise figure On-chip LNA (M8M). Extra LNA for
- lowest noise figure (M8N/Q)
- Anti jamming Active CW detection and removal. Extra
- onboard SAW band pass filter (M8N/Q)
- Supported antennas Active and passive
- Odometer Travelled distance
- Data-logger For position, velocity, and time (M8N)

Connector pinout

Pin	Signal	Voltage
1	VCC	+5.0V
2	UART TX	+3.3V
3	UART RX	+3.3V
4	I2C SCL	+3.3V
5	I2C SDA	+3.3V
6	GND	GND

5. Motor RacerStar BR2212 930 KV 2-4S *Brushless Motor For* RC *Models*

Product Specification | Product Description | Reviews

Recommended Prop : 1060 | Weight : 52g | Operating Voltage : 2-4S

RPM/V : 930KV

Racerstar BR2212 930KV 2-4S Brushless Motor For RC Models

Description:
Brand Name: Racerstar
Item Name: BR2212 brushless motor
KV: 930
Operating Voltage: 2-4S
Weight: 52g
Recommended Prop: 1060

6. 3DR Radio *Telemetry* 433 Mhz

Description

The 3DR Radio telemetry system is designed as an open source Xbee replacement radio set, offering a lower price, longer range (approx 1 mile) and superior performance to Xbee radios. The system provides a full-duplex link using HopeRF's HM-TRP modules running custom, open source firmware. Interface to the module is via standard 5V-tolerant TTL serial / FTDI USB serial. The SiK firmware includes a bootloader that permits radio firmware upgrades over the serial interface, and radio firmware with configurable parameters. Firmware upgrades and configuration are fully supported in the APM Mission Planner. Configuration is also possible through the 3DR Radio Configurator or AT commands.

Hardware features and specifications

- Very small size
- Light weight (under 4 grams without antenna)
- Available in 900MHz or 433MHz variants
- Receiver sensitivity to -121 dBm
- Transmit power up to 20dBm (100mW)
- Transparent serial link
- Air data rates up to 250kbps
- Range of approx 1 mile
- MAVLink protocol framing and status reporting
- Frequency hopping spread spectrum (FHSS)
- Adaptive time division multiplexing (TDM)
- Support for LBT and AFA
- Configurable duty cycle

- Built in error correcting code (can correct up to 25% data bit errors)
- Demonstrated range of several kilometers with a small omni antenna
- Can be used with a bi-directional amplifier for even more range
- Open source firmware
- AT commands for radio configuration
- RT commands for remote radio configuration
- Adaptive flow control when used with APM
- Based on the HopeRF HM-TRP radio module, featuring an SiLabs Si1000 RF microcontroller.

7. *Transmitter* FrSky Taranis QX-7

FrSky 2.4GHz ACCST Taranis Q X7 Manual

Introduction

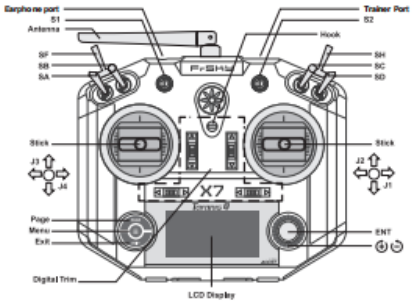
Thank you for purchasing the FrSky 2.4GHz ACCST Taranis Q X7 digital telemetry radio system. In order to make the best use of your system and to fly safely, please read this manual carefully. If you have any difficulties while using your system, please consult the manual, your hobby dealer, or FrSky technical support.

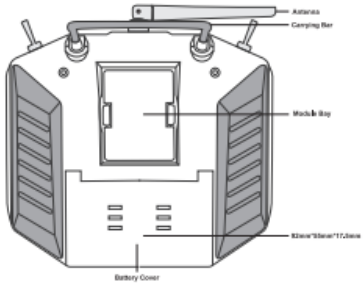
Due to unforeseen changes in production, the information contained in this manual is subject to change without notice.

Meanings of Special Markings

Pay special attention to safety where indicated by the following marks:

- DANGER - Procedures which may lead to dangerous conditions and cause death/serious injury if not carried out properly.
- WARNING - Procedures which may lead to a dangerous condition or cause death or serious injury to the user if not carried out properly or procedures where the probability of superficial injury or physical damage is high.
- CAUTION - Procedures where the possibility of serious injury to the user is small, but there is a danger of injury, or physical damage, if not carried out properly.
- Note - Steps, Tips or information.
- WARNING - Always keep electrical components away from children.





Overview

(Switch Default Settings)
- SA: 3 positions; Short Lever
- SB: 3 positions; Long Lever
- SC: 3 positions; Long Lever
- SD: 3 positions; Short Lever
- SF: 2 positions; Short Lever
- SH: 2 positions; Momentary; Long Lever

You can choose the Switch and define its positions in the Mixer menu.

Adjust sticks of Taranis Q X7

Taranis Q X7 has 4 centred sticks and will not distinguish between the throttle stick and other sticks.
You can change the stick mode and feeling according to your need.

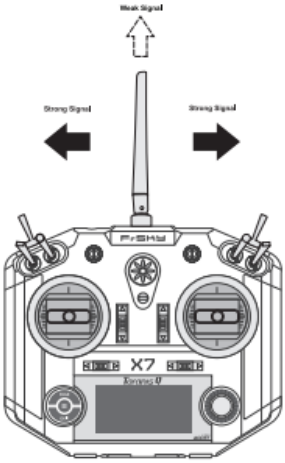
Battery Connector Polarity

Ensure that the battery connector polarity is correct when connecting batteries into the battery compartment, otherwise the Taranis Q X7 might not be powered on.

FrSky Electronic Co., Ltd www.frsky-rc.com Contact us : frsky@frsky-rc.com
Add:F-4,Building C, Zhongxiu Technology Park, No.3 Yuanxi Road, Wuxi, 214125, Jiangsu, China Technical Support: sales4tech@gmail.com

⚠ Cautions on handling antenna

- Do not touch the antenna during operation. Doing so could interfere with transmission, causing a crash.
- Do not carry the transmitter by the antenna. The antenna wire could break and prevent transmission.
- Do not pull the antenna forcefully. The antenna wire could break and prevent transmission.



Rotating Antenna

The antenna can be rotated 180 degrees and angled 90 degrees. Forcing the antenna further than this can cause damage to the antenna. The antenna is not removable.

Angle adjustment of the antenna

The antenna rotation and angle can be adjusted. The antenna features week radio signal in the forward direction and strong radio signal in the sideways directions. Adjust the antenna angle to match your flying style.

Specifications

Model Name: Taranis Q X7
Number of channels: up to 16 channels
Operating Voltage Range: 6~15V (2S, 3S Lipos are acceptable)
Operating Current: 210mA maximum (both RF module and backlit are on)
Operating Temperature: -10~45℃
Backlight LCD Screen: 128*64 outdoor readable LCD
Model Memories: 60 (extendable by MicroSD (TF) card)
Compatibility: FrSky X series, D series and V8-II series receivers (plus other receivers if an external module is used)

Taranis Q X7-EU version is only compatible with X/LR series receivers EU version.

Features

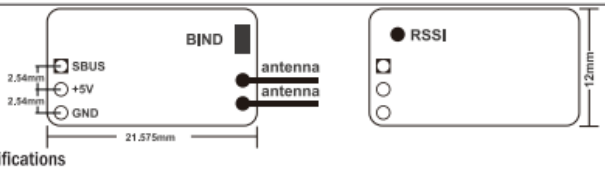
- Quad Ball Bearing Gimbals
- Receiver Match
- Audio Speech Outputs (values, alarms, settings, etc.)
- Real-time Flight Data Logging
- Receiver Signal Strength Indicator (RSSI) Alerts
- Super Low Latency
- Vibration Alerts
- Model files are compatible with TARANIS X9D/X9D Plus/X9E.
- Open source firmware OpenTx installed.

FrSky Electronic Co., Ltd www.frsky-rc.com Contact us : frsky@frsky-rc.com
Add:F-4,Building C, Zhongxiu Technology Park, No.3 Yuanxi Road, Wuxi, 214125, Jiangsu, China Technical Support: sales4tech@gmail.com

8. *Receiver* XM Plus

Instruction Manual for FrSky XM+

Overview



Specifications

Dimension: 21.5*12*3.5mm (L x W x H)
Weight: 1.6g
Number of Channels: Up to 16CH from SBUS（CH1~CH15 for PWM, CH16 RSSI for FC)
Operating Voltage Range: 3.7~10V
Operating Current: 30mA@5V
Operating Range: Full range
With RSSI output on board: Analog 0～3.3V
Firmware Upgradeable
Compatibility: FrSky D16 mode

Bindng Procedure

Binding is the process of uniquely associating a particular receiver to a transmitter module. A transmitter module can be bound to multiple receivers (not to be used simultaneously). A receiver can only be bound to one transmitter module.

Follow the steps below to finish the binding procedure.

1.Turn on the transmitter while holding the F/S button on the module (please refer to the module instruction manual for switch positions). Release the button. The RED LED on the Module will flash, indicating the transmitter is ready to bind to the receiver.

2.Connect battery to the XM+ receiver while holding the F/S button on the receiver. The LED on the receiver will flash, indicating the binding process is completed.

3.Turn off both the transmitter and the receiver.

4.Turn on the transmitter and connect the battery. The GREEN LED on the receiver indicates the receiver is receiving commands from the transmitter. The receiver/transmitter module binding will not have to be repeated, unless one of the two is replaced.

Failsafe

Failsafe is a useful feature in which all controls move to a preset position whenever the control signal is lost for a period of time. XM+ supports failsafe function for all channels. Follow the steps below to set failsafe positions for each channel:

1. Bind the receiver first and turn on both the transmitter and the receiver;
2. Move the controls to the desired failsafe position for all channels;
3. Press briefly the F/S button on the receiver (less than 1 second). The Green LED will flash twice, indicating the failsafe position has been set in the receiver.

To disable the failsafe function, re-bind the receiver.

Failsafe is recommended to set when system is firstly used, or receiver has been re-bound. Follow steps below to set failsafe.

Option-1: How to set failsafe to a user-determined state on lost signal:

1) Bind the receiver to the transmitter module first and turn on both the transmitter and the receiver;
2) Move the controls to desired failsafe position for all channels;
3) Press the F/S button on the receiver (less than 1 second). The Green LED will flash twice, indicating the failsafe position has been set in the receiver.

Option-2: How to set failsafe for no pulses on lost signal (needed for some flight contollers):

1. Turn off the transmitter, power on the receiver, and then press the F/S button on the receiver (less than 1 second). The Green LED will flash twice, indicating the failsafe position has been set in the receiver.

Note: If failsafe is not set, failsafe default will hold last position before signal was lost. In this mode your model may fly away or cause injury.

FrSky Electronic Co., Ltd
Website:www.frsky-rc.com E-mail:frsky@frsky-rc.com Technical Support: sales4tech@gmail.com

B. Dokumentasi

C. Source Code

- **Program Utama . Nama file : main.py**

```
import cv2
import numpy as np
from dronekit import connect, VehicleMode, LocationGlobal,
LocationGlobalRelative, APIException
from timeit import default_timer as timer
from queue import Queue
from threading import Thread
from time import sleep
from control import Drone, PID
import argparse
import csv

# Connect the Vehicle
parser = argparse.ArgumentParser(description='commands')
parser.add_argument('--connect',
                    help="Vehicle connection target string. If not specified,
SITL automatically started and used.")

args = parser.parse_args()

connection_string = "/dev/ttyACM0"
sitl = None

class Camera:
    def __init__(self, camNumber):
        self.camNumber = camNumber

        # Connect to Camera
        self.cap = cv2.VideoCapture(camNumber)
        self.cap.set(cv2.CAP_PROP_FRAME_WIDTH, 640)
        self.cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 480)

        # Camera Threading
        self.qCamera = Queue()
        self.tCamera = Thread(target=self._reader)
        self.tCamera.daemon = True
```

```
        self.tCamera.start()

    def _reader(self):
        while True:
            ret, frame = self.cap.read()
            if not ret:
                break
            if not self.qCamera.empty():
                try:
                    self.qCamera.get_nowait()
                except:
                    pass
            self.qCamera.put((ret, frame))
            sleep(0.001)

    def read(self):
        return self.qCamera.get()

    def release(self):
        self.cap.release()
        cv2.destroyAllWindows()


drone = Drone(connection_string)
cap = Camera(0)
pidX = PID(P=0.002, I=0, D=0.0001)
pidX.setPoint=160
pitchList = []
yawList = []
rollList= []
kabel = []
velocity_y = []

while (True):
    ret, frame = cap.read()
    gambar = frame[0:240, 160:480]
    roi = frame[0:480, 160:480]
    start = timer()
    if ret:
```

```
gray = cv2.cvtColor(roi, cv2.COLOR_BGR2GRAY)
thr = cv2.adaptiveThreshold(gray, 255,
cv2.ADAPTIVE_THRESH_MEAN_C, cv2.THRESH_BINARY,
15, -2)
# Create the images that will use to extract the vertical lines
vertical = np.copy(thr)
# [vert]
# Specify size on vertical axis
rows = vertical.shape[0]
verticalsize = rows // 44
# Create structure element for extracting vertical lines through
  morphology operations
verticalStructure =
cv2.getStructuringElement(cv2.MORPH_RECT, (1, verticalsize))
 # Apply morphology operations
vertical = cv2.erode(vertical, verticalStructure)
edges = cv2.Canny(vertical, 20, 255, apertureSize=3)
#cv2.imshow('edges', edges)
lines = cv2.HoughLinesP(edges, 1, np.pi / 180, 100,
minLineLength=400, maxLineGap=225)
xCable = 0
if lines is not None:
    for line in lines:
        x1, y1, x2, y2 = line[0]
        cv2.line(gambar, (x1, y1), (x2, y2), (0, 255, 0), 2)
        xCable = xCable + x1 + x2
    xCable /= len(lines) * 2
    cv2.line(gambar, (xCable, 0), (xCable, 240), (255, 0, 0), 3)

fps = str(round(1 / (timer() - start), 2))
cv2.putText(gambar, fps, (12, 50),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 3)

cv2.imshow('frame', gambar)

if drone.checkMode() is "GUIDED":
            print("Drone in GUIDED mode")
            print('xCable = ',xCable)
            pidX.update(xCable)
```

```
            if xCable==0:
                        vx, vy, vz = 0, 0, -0.05
    else:
                        vy = pidX.output * (-1)
                        vx = +0.5
                        vz = 0

    if vy >= 0.3:
                        vy = 0.3
    elif vy <= -0.3:
                        vy = -0.3
    else:
                        vy = vy

    drone.set_velocity_body(vx, vy, vz)
                print('vx, vy, vz = ',vx,vy,vz)
                pitchRead = drone.vehicle._pitchspeed
                yawRead = drone.vehicle._yawspeed
                rollRead = drone.vehicle._rollspeed
                print('pitch = ',pitchRead)
                print('roll = ',rollRead)
                print('yaw = ',yawRead)

                pitchList.append(pitchRead)
                yawList.append(yawRead)
                rollList.append(rollRead)
                kabel.append(xCable)
                velocity_y.append(vy)
                with open('data2.csv', 'w') as write:
                        theWriter = csv.writer(write)

                theWriter.writerow(['pitch' ,pitchList])
                        theWriter.writerow(['yaw', yawList])
                        theWriter.writerow(['roll', rollList])
                        theWriter.writerow(['xCable', kabel])
                        theWriter.writerow(['vy', velocity_y])


if cv2.waitKey(1) & 0xFF == ord('q'):
```

```
            break
    else:
        break

cap.release()
cv2.destroyAllWindows()
```

- **Program kontrol *drone*. Nama file : control.py**

```
from __future__ import print_function
from droneкit import connect, VehicleMode, LocationGlobal,
LocationGlobalRelative, APIException
from pymavlink import mavutil  # needed for command message
definitions
import time

class Drone:
    def __init__(self, port):
        """
        Drone initialize
        """
        self.port = port
        print('Vehicle : Connecting to vehicle on: %s' % port)
        self.vehicle = connect(port, baud=115200, wait_ready=True)

        # Time
        self.time_now = None
        self.time_past = None
        self.time_out = None

    def sendMavlink(self, vX, vY, vZ):
        self.set_velocity_body(vX, vY, vZ)

    def set_velocity_body(self, vX, vY, vZ):
        """ Remember: vz is positive downward!!!
        http://ardupilot.org/dev/docs/copter-commands-in-guided-
mode.html
```

```
        Bitmask to indicate which dimensions should be ignored by the
vehicle
        (a value of 0b0000000000000000 or 0b0000001000000000
indicates that
        none of the setpoint dimensions should be ignored). Mapping:
        bit 1: x,  bit 2: y,  bit 3: z,
        bit 4: vx, bit 5: vy, bit 6: vz,
        bit 7: ax, bit 8: ay, bit 9:
        """
        msg =
self.vehicle.message_factory.set_position_target_local_ned_encode(
            0,
            0, 0,
            mavutil.mavlink.MAV_FRAME_BODY_NED,
            0b0000111111000111,  # -- BITMASK -> Consider only the
velocities
            0, 0, 0,  # -- POSITION
            vX, vY, vZ,  # -- VELOCITY
            0, 0, 0,  # -- ACCELERATIONS
            0, 0)
        self.vehicle.send_mavlink(msg)

    def checkMode(self, compare=None):
        """
        :return: current vehicle mode
        """
        if compare is None:
            return self.vehicle.mode.name
        else:
            return self.vehicle.mode == VehicleMode(compare)

class PID:
    def __init__(self, P=0.2, I=0.0, D=0.0, speedCutoff=0.1,
current_time=None):
        self.Kp = P
        self.Ki = I
        self.Kd = D
        self.setPoint = 0
        self.speedCutoff = speedCutoff
```

```
        self.sample_time = 0.00
        self.current_time = current_time if current_time is not None else
time.time()
        self.last_time = self.current_time

        self.PTerm = 0.0
        self.ITerm = 0.0
        self.DTerm = 0.0
        self.last_error = 0.0

        self.int_error = 0.0
        self.windup_guard = 20.0

        self.output = 0.0

    def clearPID(self):
        self.PTerm = 0.0
        self.ITerm = 0.0
        self.DTerm = 0.0
        self.last_error = 0.0

        # Integral protector
        self.int_error = 0.0
        self.windup_guard = 20.0

        self.output = 0.0

    def update(self, value, current_time=None):
        error = self.setPoint - value
        self.current_time = current_time if current_time is not None else
time.time()
        delta_time = self.current_time - self.last_time
        delta_error = error - self.last_error

        if delta_time >= self.sample_time:
            self.PTerm = self.Kp * error
            self.ITerm += error * delta_time
```

```
            if self.ITerm < - self.windup_guard:
                self.ITerm = - self.windup_guard
            elif self.ITerm > self.windup_guard:
                self.ITerm = self.windup_guard

            self.DTerm = 0.0
            if delta_time > 0:
                self.DTerm = delta_error / delta_time

            self.last_time = self.current_time
            self.last_error = error

            self.output = self.PTerm + (self.Ki * self.ITerm) + (self.Kd *
self.DTerm)

            if self.speedCutoff is not None:
                if abs(self.output) > self.speedCutoff:
                    self.output = (self.output / abs(self.output)) /
self.speedCutoff
                    return self.output
                else:
                    return self.output

    def setKp(self, proportional_gain):
        self.Kp = proportional_gain

    def setKi(self, integral_gain):
        self.Ki = integral_gain

    def setKd(self, derivative_gain):
        self.Kd = derivative_gain

    def setWindup(self, windup):
        """
        Maximum Integrator
        :param windup:
        :return:
        """
        self.windup_guard = windup
```

```
def setSampleTime(self, sample_time):
    self.sample_time = sample_time
```

[Halaman ini sengaja dikosongkan]

BIODATA PENULIS

Penulis merupakan kelahiran dari kota Lhokseumawe, Aceh, tanggal 28 April 1998. Penulis merupakan anak kedua dari lima bersaudara. Penulis menjalankan studi Strata-1 di Departemen Teknik Elektro, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember, Surabaya pada tahun 2016 setelah lulus dari SMAN Modal Bangsa Aceh. Penulis mengambil bidang studi elektronika sebagai bidang studi pilihan. Selama kuliah, penulis aktif sebagai asisten laboratorium B202 dan laboratorium A206.

Email : muhammad.marie@gmail.com
HP : 0822-7213-1748