

DAFTAR TABEL

Tabel 2.1 Spesifikasi produk.....	8
Tabel 2.2 Verifikasi 1.....	8
Tabel 2.3 Verifikasi 2.....	8
Tabel 2.4 Verifikasi 3.....	9
Tabel 3.1 Analisis Konsep.....	12
Tabel 3.2 Kalibrasi perangkat pengambil gambar.....	22
Tabel 3.3 Perbandingan performance YOLOv8 model.....	23
Tabel 3.4 Jadwal pengerjaan.....	23
Tabel 4.1 Perbandingan panjang bounding box.....	29
Tabel 4.2 Indeks tinggi pohon.....	30
Tabel 5.1 Partisi data.....	34
Tabel 5.2 Pengujian partisi 1.....	35
Tabel 5.3 Pengujian partisi 2.....	36
Tabel 5.4 Pengujian partisi 3.....	37
Tabel 5.5 Pengujian partisi 4.....	38
Tabel 5.6 Pengujian batch size 4.....	39
Tabel 5.7 Pengujian batch size 8.....	40
Tabel 5.8 Pengujian batch size 16.....	41
Tabel 5.9 Pengujian batch size 32.....	42
Tabel 5.10 Pengujian batch size 64.....	43
Tabel 5.11 Pengujian learning rate 0.01.....	44
Tabel 5.12 Pengujian learning rate 0.001.....	45
Tabel 5.13 Pengujian learning rate 0.0001.....	46
Tabel 5.14 Pengujian learning rate 0.00001.....	47
Tabel 5.15 Pengujian epoch 50.....	48
Tabel 5.16 Pengujian epoch 100.....	49
Tabel 5.17 Pengujian epoch 150.....	50
Tabel 5.18 Pengujian epoch 200.....	51
Tabel 5.19 Pengujian confidence threshold.....	53
Tabel 5.20 Pengujian jarak dan ketinggian drone kelas pisang.....	54
Tabel 5.21 Pengujian jarak dan ketinggian drone kelas pinus.....	54
Tabel 5.22 Pengujian jarak dan ketinggian drone kelas palm.....	55

Tabel 5.23 Pengujian intensitas cahaya siang hari.....	57
Tabel 5.24 Pengujian intensitas cahaya sore hari.....	57
Tabel 5.25 Pengujian kecepatan inferensi.....	57
Tabel 5.26 Pengujian alpha.....	59
Tabel 5.27 Keterangan skala likert.....	60
Tabel 5.28 Hasil pengujian beta.....	60
Tabel 5.29 Analisis pengujian partisi data.....	61
Tabel 5.30 Analisis pengujian batch size.....	62
Tabel 5.31 Analisis pengujian learning rate.....	63
Tabel 5.32 Analisis pengujian epoch.....	64
Tabel 5.33 Analisis confidence threshold.....	66
Tabel 5.34 Analisis rata rata error dan akurasi.....	68
Tabel 5.35 Analisis intensitas cahaya.....	68
Tabel 5.36 Analisis alpha testing.....	69
Tabel 5.37 Analisis beta testing.....	70

DAFTAR SINGKATAN

AI	: <i>Artificial Intelligence</i>
UAV	: <i>Unmanned Aerial Vehicle</i>
LiDAR	: <i>Light Detection and Ranging</i>
R-CNN	: <i>Region-based Convolutional Neural Network</i>
YOLO	: <i>You Only Look Once</i>
CNN	: <i>Convolutional Neural Network</i>
IOU	: <i>Interesection Over Union</i>
CSS	: <i>Cascading Style Sheets</i>
HTML	: <i>HyperText Markup Language</i>
mAP	: <i>mean Average Precision</i>
UAT	: <i>User Acceptance Testing</i>

BAB 1

USULAN GAGASAN

1.1 Latar Belakang Masalah

Pohon memainkan peran yang sangat penting tidak hanya dalam sektor kayu, tetapi juga dalam siklus karbon global dan perubahan iklim. Tanpa pohon, makhluk hidup tidak dapat bertahan hidup karena pohon merupakan sumber dari segala kehidupan bagi makhluk hidup. Salah satu elemen terpenting dari struktur pohon adalah hubungan antara tinggi pohon dan jenisnya. Pengukuran tinggi pohon penting dalam bidang-bidang seperti ekologi, penelitian lingkungan, konservasi hutan, dan industri kehutanan. Data tinggi pohon memberikan informasi terkait struktur dan keanekaragaman hutan, produktivitas hutan, serta pertumbuhan dan perkembangan pohon. Informasi tersebut membantu para ilmuwan, ahli kehutanan, dan pengambil kebijakan dalam membuat keputusan yang berkaitan dengan pengelolaan sumber daya hutan dan upaya pelestarian alam.

Ada beberapa metode yang digunakan untuk mengukur tinggi pohon. Metode yang paling umum digunakan adalah pengukuran langsung atau tidak langsung [1]. Pengukuran langsung dapat dilakukan dengan tongkat pengukur teleskopik, tetapi metode ini lambat dan rumit. Sebagai alternatif, pengukuran langsung dapat dilakukan dengan cara menebang pohon dan panjangnya ditentukan di tanah dengan alat ukur seperti rol meter. Sedangkan pengukuran tidak langsung dapat dilakukan dengan metode trigonometri menggunakan perangkat laser [2]. Metode pengukuran tidak langsung juga dapat menggunakan teknik penginderaan jauh dengan sensor aktif seperti *Light Detection and Ranging* (LiDAR)[3].

Dengan kemajuan teknologi, penggunaan dan aplikasi kendaraan udara tanpa awak (*Unmanned Aerial Vehicle/UAV*) atau *drone* menjadi opsi yang membantu manusia dalam menjalani kehidupan sehari-hari. UAV merupakan pesawat terbang yang dapat beroperasi tanpa awak yang dikendalikan secara jarak jauh atau otomatis[4]. Pemanfaatan *drone* menjadi solusi efektif untuk mengatasi tantangan dalam berbagai bidang, seperti pengiriman barang, keamanan, pemetaan, termasuk penelitian. Penggunaan *drone* semakin berkembang karena *drone* dapat menjangkau wilayah yang sulit dijangkau dan dapat menyediakan data secara *real-time* yang pada akhirnya membuka atau mengubah cara kerja dan interaksi dengan lingkungan sekitar.

Dalam beberapa tahun terakhir, perkembangan AI (*Artificial Intelligence*) telah mengalami kemajuan pesat. AI adalah kemampuan sistem komputer untuk meniru, memahami,

dan melakukan tugas yang biasanya secara tradisional menggunakan kecerdasan manusia. AI melibatkan penggunaan algoritma dan teknik komputasi yang kompleks untuk memungkinkan mesin mengumpulkan, menganalisis, dan memproses informasi, serta membuat keputusan atau tindakan berdasarkan pemahaman tersebut. [5].

Drone dan AI memiliki potensi besar untuk proses klasifikasi jenis pohon dan juga pengukuran ketinggiannya. Dengan memanfaatkan kedua teknologi tersebut dapat melakukan klasifikasi dan juga pengukuran ketinggiannya lebih cepat dan akurat, sehingga menghemat waktu dan sumber daya manusia yang dibutuhkan sebelumnya. Salah satu tantangannya adalah keterbatasan dataset yang diperlukan untuk proses *training*. Untuk melatih model AI yang efektif diperlukan data dengan jumlah yang cukup banyak. Mengumpulkan data yang relevan dan memadai untuk *training* model AI menjadi tugas yang memakan waktu.

Dengan keterbatasan dari proses klasifikasi dan deteksi tinggi pohon secara manual, maka dirancanglah sistem klasifikasi jenis dan estimasi tinggi pohon menggunakan *drone* dengan memanfaatkan AI untuk mengoptimalkan proses tersebut agar lebih cepat dan akurat.

1.2 Informasi Pendukung Masalah

Penelitian yang dilakukan oleh Nurwahyu Malik (2014) menyatakan bahwa respon pertumbuhan tinggi tanaman sambiloto hasil pemberian pupuk dan intensitas cahaya matahari yang berbeda, diperoleh kesimpulan adanya perbedaan pertumbuhan tinggi tanaman sambiloto terhadap pemupukan dan intensitas cahaya matahari yang berbeda [6]. Imam Wahyudi, Dicky Krista Dinata, dan Lidia Binti Jasni (2014) melakukan penelitian yang memberikan kesimpulan Jarak tanam lebih memengaruhi karakteristik pertumbuhan pohon terutama diameter batang, tinggi pohon total, tinggi pohon bebas cabang dan jumlah percabangan [7] Sedangkan Ying, C. C., dan Morgenstern, E.K. (1979) melakukan penelitian terhadap pohon cemara yang memiliki kesimpulan bahwa perbedaan induk pohon cemara pada hutan *Picea Glauca* mempengaruhi ketinggian keturunannya [8]

Tiga penelitian yang telah disebutkan menggunakan ketinggian pohon sebagai bahan penelitian yang dilakukan. Tidak hanya tiga penelitian tersebut saja yang menggunakan ketinggian sebagai bahan penelitiannya tetapi masih banyak lagi penelitian – penelitian lain yang menggunakan ketinggian pohon sebagai bahan penelitian mereka, artinya adalah pengukuran ketinggian pohon sangat berguna untuk mendapatkan informasi yang akan digunakan untuk penelitian terutama tentang ekologi kehutanan.

Object Detection adalah sebuah teknik dalam bidang pengolahan citra dan pengenalan pola dalam kecerdasan buatan yang bertujuan untuk mendeteksi dan mengidentifikasi objek atau benda tertentu dalam gambar atau video. Tujuan utama dari *Object Detection* adalah untuk menemukan lokasi dan menentukan jenis atau kelas dari objek yang ada dalam gambar atau frame video [9]. Dalam *Object Detection* terdapat banyak algoritma yang di kembangan diantaranya adalah R-CNN (*Region-based Convolutional Neural Network*), *Fast R-CNN*, YOLO (*You Only Look Once*), serta masih banyak yang lainnya. Dalam pendeteksian objek secara realtime algoritma YOLO lebih umum digunakan karena frameworknya lebih stabil dalam kecepatan dan akurasi [10].

1.3 Analisis Umum

Pada proyek ini kami memperhatikan beberapa aspek, diantaranya yaitu:

1.3.1 Aspek Ekonomi

Pada aspek ekomoni pendeteksi tinggi pohon dan jenisnya menggunakan *drone* akan lebih hemat dibandingkan dengan menggunakan kamera handphone karena hanya handphone kelas atas yang harganya tidak murah yang dapat digunakan untuk mengukur tinggi pohon menggunakan sensor LiDAR, adapun biaya yang dikeluarkan jika menggunakan *drone* hanya untuk membeli *drone* dengan kelas menengah dan juga biaya untuk *maintenance* saja. Dibantu dengan menggunakan program deep learning membuat proses pengukuran menjadi lebih cepat dan efisien yang dampaknya akan mengurangi biaya operasional dibandingkan dengan pengukuran secara manual menggunakan hagameter dan clinometer ataupun menggunakan kamera handphone.

1.3.2 Aspek Manufakturabilitas

Dalam perancangan proyek ini diperhatikan juga aspek manufakturabilitasnya. Di indonesia sendiri sangat mudah bagi kita untuk memperoleh *drone* yang menjadi alat utama pada proyek ini, apalagi saat ini sudah banyak marketplace yang memudahkan kita untuk mendapatkan *drone* tersebut. Dengan mudahnya kita mendapatkan *drone* maka dari aspek manufakturabilitas sangat baik. Selain itu pengembangan sistem ini membutuhkan tenaga kerja yang memiliki keahlian dalam bidang Python, *Deep Learning*, *Computer Vision*, Data, Skill pengoperasian *drone*.

Alat dan bahan yang dibutuhkan dalam perancangan diantaranya:

- A. *Drone* sebagai alat utama penangkap gambar video secara *Real-time*
- B. Satu set komputer digunakan untuk merancang, dan mengoperasikan program tersebut.

C. Kamera untuk mengambil gambar pohon sebagai dataset.

D. Aplikasi anotasi dataset seperti roboflow

1.3.3 Aspek Efisiensi

Sistem ini dapat mengukur tinggi pohon dalam jumlah banyak sekaligus artinya tidak menghitung secara manual satu persatu. Dengan proses tersebut maka akan lebih efisien dari segi waktu dan tenaga operasionalnya.

1.3.4 Aspek Keberlanjutan (*sustainability*)

Ditengah perkembangan teknologi yang semakin canggih dan juga semakin cepat, kedepannya projek ini akan semakin berkembang dan juga akan semakin dibutuhkan. Jika saat ini projek bertujuan untuk mengukur tinggi pohon dan jenis pohon berdasarkan familinya, maka pengembangan selanjutnya memungkinkan identifikasi pohon menjadi spesifik berdasarkan spesiesnya, serta memungkinkan juga pengembangan pengembangan yang lainnya seperti identifikasi pohon berbuah siap panen dan lainnya.

1.4 Kebutuhan yang Harus Dipenuhi

Berdasarkan masalah, latar belakang, dan analisis yang telah dipaparkan, maka kebutuhan yang harus dipenuhi dari solusi yang akan diajukan antara lain:

- A. Sistem mendeteksi jenis pohon menggunakan citra *drone* berdasarkan famili yang sudah ditentukan.
- B. Sistem mendeteksi tinggi pohon menggunakan citra dari *drone*.
- C. Sistem dapat diimplementasikan secara *real-time* melalui *website* sederhana.

1.5 Solusi Sistem yang Diusulkan

1.5.1 Karakteristik Produk

A. Fitur Utama:

Memudahkan membuat dataset dengan cepat dan melakukan perhitungan tinggi dan jenis pohon yang tidak dapat dijangkau oleh kamera manual.

B. Fitur Dasar:

1. Dapat mendeteksi objek berupa pohon.
2. Dapat mengklasifikasi jenis pohon berdasarkan familinya.

C. Fitur Tambahan:

1. Pengambilan gambar dari atas ketinggian.
2. Dapat mendeteksi tinggi pohon.

D. Sifat solusi yang diharapkan:

1. Tidak perlu banyak mengeluarkan biaya operasional.
2. Solusi diharuskan untuk dapat dioperasikan dengan mudah.
3. Tidak membutuhkan perawatan yang terlalu intensif.

1.5.1.1 Solusi 1

Sistem diimplementasikan pada *drone* secara *real-time* untuk mendeteksi jenis pohon dan mengukur ketinggian pohon, sistem menggunakan algoritma YOLO untuk proses pendeteksian, serta menggunakan *drone* agar area pengukuran lebih luas dan waktu pengukuran menjadi lebih efisien dibandingkan menggunakan alat ukur manual seperti hagameter dan clinometer ataupun menggunakan kamera manual.

1.5.1.2 Solusi 2

Sistem diimplementasikan pada *drone* secara tidak langsung. *Drone* hanya digunakan untuk mengambil video pohon yang akan dideteksi. Selanjutnya video yang didapatkan di proses terpisah tidak secara *real-time*.

1.5.1.3 Solusi 3

Sistem diimplementasikan pada citra satelit untuk mendeteksi jenis pohon dan mengambil data tinggi pohon, sama seperti solusi satu sistem menggunakan algoritma YOLO untuk proses pengukuran dan pendeteksiannya. Namun dalam proses ini tidak dilakukan secara *real-time*, karena data yang didapat dari citra satelit bukan data *real-time*.

1.5.2 Skenario Penggunaan

1.5.2.1 Solusi 1

1. *Drone* diterbangkan untuk proses pengambilan gambar secara *real-time*
2. Dengan menggunakan algoritma YOLO gambar yang diambil oleh *drone* akan diproses untuk pendeteksian objek.
3. *Bounding box* yang nampak akan menjadi acuan untuk nilai perbandingan untuk mengukur nilai ketinggian pohon.
4. *Drone* akan mendeteksi jenis dan tinggi pohon secara *real-time*, serta menampilkannya pada *website* sederhana.

1.5.2.2 Solusi 2

1. *Drone* diterbangkan untuk proses pengambilan gambar.
2. Video yang sudah didapatkan diproses menggunakan algoritma YOLO secara terpisah.

3. *Bounding box* yang nampak akan menjadi acuan untuk nilai perbandingan untuk mengukur nilai ketinggian pohon.
4. Hasil pendeteksian tinggi dan klasifikasi pohon dari video yang digunakan akan ditampilkan pada *website* sederhana.

1.5.2.3 Solusi 3

1. Mengambil citra gambar di web satelit.
2. Citra satelit digunakan untuk pengambilan gambar dan pemetaan dalam mengambil data.
3. Gambar yang diambil akan diproses menggunakan algoritma YOLO
4. Hasil pendeteksian tinggi dan klasifikasi pohon berupa gambar akan ditampilkan pada *website*.

1.6 Kesimpulan dan Ringkasan CD-1

Dengan perkembangan teknologi saat ini, penguunaan *drone* dan AI dapat dimanfaatkan untuk berbagai macam kebutuhan manusia. Pada rancangan proyek ini, AI diimplementasikan menggunakan *drone* untuk mendeteksi tinggi pohon dan juga identifikasi jenis pohon tersebut. Penggunaan *drone* akan lebih efisien karena dapat mengukur dalam area yang luas sekaligus jika dibandingkan menggunakan kamera manual yang harus mengukurnya satu persatu. Dengan pendekatan AI diharapkan sistem dapat bekerja dengan akurat dan efisien sehingga nantinya data tinggi dan jenis pohon dapat didapatkan dengan mudah oleh pihak yang membutuhkannya.

BAB 2

DESAIN KONSEP SOLUSI

2.1 Spesifikasi Produk

Berdasarkan CD-1, kebutuhan sistem adalah :

1. Sistem dapat mendeteksi jenis pohon

Sistem dapat mendeteksi jenis pohon berdasarkan famili yang sudah ditentukan yaitu famili pohon pinus – pinusan, pisang – pisangian, dan pinang - pinangan. *Drone* mampu terbang dalam jarak yang jauh sehingga memungkinkan sistem untuk mendeteksi banyak jenis pohon sekaligus dan meningkatkan efisiensi pendeteksian. Dalam mendeteksi jenis pohon yang perlu diperhatikan adalah sudut dan jarak *drone* ke pohon yang akan dideteksi jenisnya pada saat pengujian.

2. Sistem dapat mendeteksi tinggi pohon

Sistem dapat mendeteksi tinggi pohon menggunakan citra gambar *drone*. Sistem melakukan perhitungan perbandingan antara *pixel* tinggi *bounding box* hasil deteksi jenis pohon dengan ketinggian pohon sebenarnya. Yang perlu diperhatikan saat sistem ini digunakan adalah pencahayaan dan jarak *drone* ke pohon yang akan dihitung tingginya pada saat pengujian.

3. Sistem dapat diimplementasikan secara *real-time* melalui *website* sederhana.

Hasil pendeksian oleh sistem ditampilkan secara *real-time* melalui *website* sederhana yang berisikan video *streaming* dari *drone*. *Website* juga memiliki fitur untuk mengatur nilai *confidence threshold* dan mengatur jarak dari hasil deteksi jenis pohon yang dilakukan.

Karakteristik sistem:

1. Fitur utama

- A. Dapat mendeteksi tinggi pohon.
- B. Dapat mendeteksi jenis pohon berdasarkan famili yang sudah ditentukan.

2. Fitur dasar:

- A. Sistem dapat mendeteksi tinggi pohon seakurat mungkin.
- B. Sistem dapat mendeteksi jenis pohon berdasarkan famili yang sudah ditentukan menggunakan citra gambar yang tersedia secara *real-time*.

3. Fitur tambahan:

- A. Sistem dapat menampilkan hasil deteksi secara *real-time*.

Tabel 2.1 Spesifikasi produk

No	Hal	Rincian
1	Mendeteksi jenis pohon	Sistem mendeteksi jenis pohon menggunakan citra <i>drone</i> berdasarkan famili yang sudah ditentukan.
2	Mendeteksi tinggi pohon	Sistem mendeteksi tinggi pohon menggunakan citra dari <i>drone</i> .
3	Implementasi secara <i>real-time</i>	Sistem dapat diimplementasikan secara <i>real-time</i> melalui <i>website</i> sederhana.

2.2 Verifikasi

Dari penjelasan spesifikasi diatas, langkah-langkah sebelum melakukan spesifikasi yang sesuai perlu adanya verifikasi dari beberapa spesifikasi diatas, diantaranya.

2.2.1 Verifikasi Spesifikasi 1

Tabel 2.2 Verifikasi 1

No	Hal	Pengolahan data
1	Rincian	Sistem mendeteksi jenis pohon menggunakan citra <i>drone</i> berdasarkan famili yang sudah ditentukan.
2	Metode Pengujian	Pengujian dilakukan secara langsung menggunakan <i>drone</i> .
3	Prosedur Pengujian	Sistem diberikan citra dari <i>drone</i> yang terdapat pohon yang berbeda didalamnya, untuk menguji apakah sistem dapat mendeteksi objek berupa pohon dan dapat membedakan jenis pohon yang ada. <i>Drone</i> juga dijalankan pada sudut dan jarak yang berbeda, untuk menemukan posisi yang tepat agar mendapatkan hasil pendeteksian yang maksimal

2.2.2 Verifikasi spesifikasi 2

Tabel 2.3 Verifikasi 2

No	Hal	Pengolahan data
1	Rincian	Sistem mendeteksi tinggi pohon menggunakan citra dari <i>drone</i> secara <i>realtime</i> .
2	Metode Pengujian	Pengukuran dilakukan secara langsung menggunakan <i>drone</i> .

No	Hal	Pengolahan data
3	Prosedur Pengujian	Sistem diberikan citra dari <i>drone</i> yang terdapat beberapa pohon didalamnya. <i>Drone</i> juga dijalankan pada sudut dan jarak terbaik saat pendeteksian jenis pohon sebagai acuan kalibrasi perbandingan tinggi pohon. setelah tinggi pohon terdeteksi maka dilakukan pengukuran pohon secara manual untuk membandingkan hasil pengukuran sistem apakah akurat atau tidak.

2.2.3 Verifikasi spesifikasi 3

Tabel 2.4 Verifikasi 3

No	Hal	Pengolahan data
1	Rincian	Sistem dapat diimplementasikan secara <i>real-time</i> melalui <i>website</i> sederhana.
2	Metode Pengujian	Akses <i>website</i> sederhana
3	Prosedur Pengujian	Sistem diuji dengan pengaksesan <i>website</i> sederhana untuk melihat hasil deteksi secara realtime. Serta dilakukan pengujian fitur yang ada pada <i>website</i> tersebut apakah berfungsi atau tidak.

2.3 Kesimpulan dan Ringkasan CD-2

Sistem yang dibuat memiliki 3 spesifikasi diantaranya sistem dapat mendeteksi jenis pohon, sistem dapat mendeteksi tinggi pohon, serta sistem dapat diimplementasikan secara *real-time* melalui *website* sederhana. Untuk memastikan sistem sesuai spesifikasi yang disebutkan maka dilakukan verifikasi pada tiap spesifikasi yang disebutkan. Sistem yang dibuat diharapkan dapat memenuhi tiga spesifikasi agar sistem dapat bekerja secara maksimal.

BAB 3

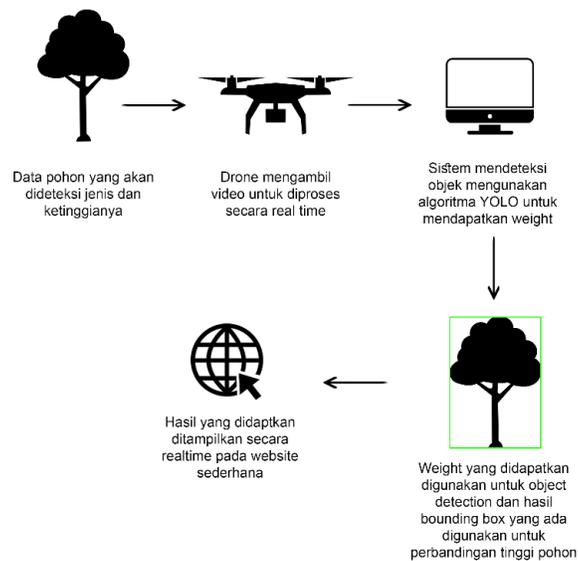
DESAIN RANCANGAN SOLUSI

3.1 Konsep Sistem

Dalam pembahasan bab sebelumnya terdapat tiga solusi yang ditawarkan. Diantaranya sistem diimplementasikan menggunakan *drone* secara realtime, sistem diimplementasikan menggunakan *drone* secara tidak *real-time*, dan sistem diimplementasikan menggunakan citra satelit secara tidak realtime.

3.1.1 Pilihan Sistem

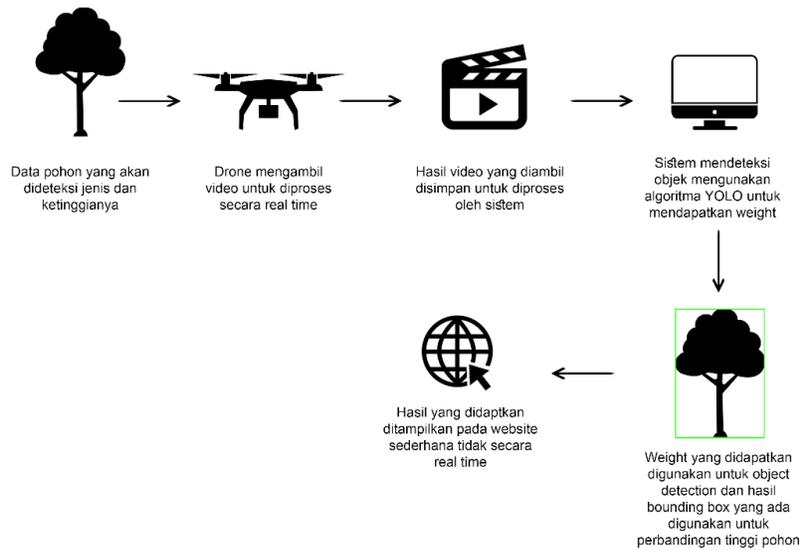
A. Solusi 1 Sistem diimplementasikan menggunakan *drone* secara *real-time*



Gambar 3.1 Gambaran umum solusi 1

Sistem diimplementasikan pada *drone* secara *real-time* untuk mendeteksi jenis pohon dan mengukur ketinggian pohon, sistem menggunakan algoritma YOLO untuk proses pendeteksian, serta menggunakan *drone* agar area pengukuran lebih luas dan waktu pengukuran menjadi lebih efisien dibandingkan menggunakan alat ukur manual seperti hagameter dan clinometer ataupun menggunakan kamera manual.

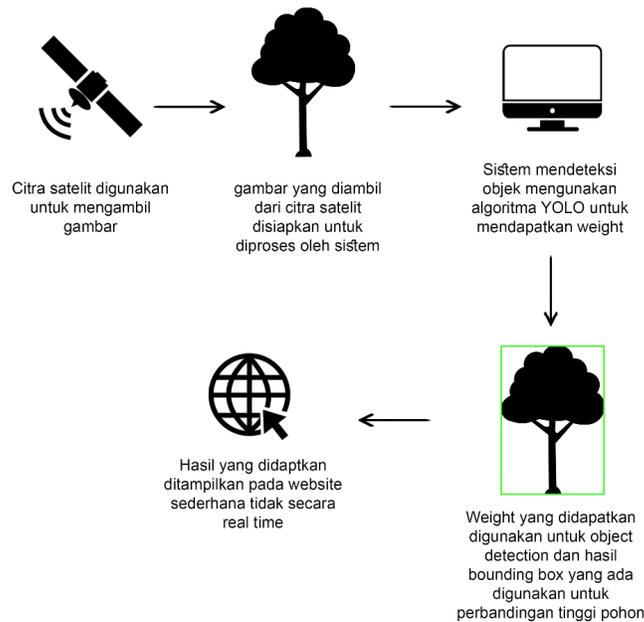
B. Solusi 2 sistem diimplementasikan menggunakan *drone* secara tidak *real-time*



Gambar 3.2 Gambaran umum solusi 2

Sistem diimplementasikan pada *drone* secara tidak langsung. *Drone* hanya digunakan untuk mengambil video pohon yang akan dideteksi. Selanjutnya video yang didapatkan di proses terpisah tidak secara *real-time*.

C. Solusi 3 sistem diimplementasikan menggunakan citra satelit secara tidak realtime.



Gambar 3.3 Gambaran umum solusi 3

Sistem diimplementasikan pada citra satelit untuk mendeteksi jenis pohon dan mengambil data tinggi pohon, sama seperti solusi satu sistem menggunakan algoritma YOLO untuk proses pengukuran dan pendeteksiannya. Namun dalam proses ini tidak dilakukan secara *real-time*, karena data yang didapat dari citra satelit bukan data *real-time*.

3.1.2 Analisis

1. Kriteria

A. Pengambilan gambar

Drone yang digunakan harus memiliki kualitas video yang bagus, dimana dapat mengambil gambar dengan halus dan memiliki *frame rate* yang tinggi. Sehingga AI dapat memberikan keputusan yang terbaik.

B. Waktu Pemrosesan Sistem

Sistem yang berjalan harus beroperasi secepat mungkin untuk meningkatkan efisiensi waktu dalam proses pendeteksian tinggi dan jenis pohon.

C. Akurasi Sistem

Sistem yang berjalan harus memiliki keakurasian yang cukup tinggi agar pengguna tidak salah dalam mengambil data yang ditampilkan pada *output* sistem.

2. Analisis Konsep

Analisis konsep perlu dilakukan untuk menentukan solusi mana yang cocok untuk diterapkan.

Tabel 3.1 adalah analisis konsep yang dilakukan.

Tabel 3.1 Analisis Konsep

Solusi	Pengambilan gambar	Waktu pemrosesan	Akurasi sistem
1	Pengambilan gambar bisa dilakukan dari berbagai sudut	Waktu pemrosesan lebih cepat karena sistem berjalan secara <i>real-time</i>	Akurasi sistem lebih akurat karena pohon yang dideteksi secara langsung
2	Pengambilan gambar bisa dilakukan dari berbagai sudut	Waktu pemrosesan tidak lebih cepat karena tidak <i>real-time</i>	Akurasi sistem lebih akurat karena pohon yang dideteksi secara langsung
3	Pengambilan gambar tidak dapat dilakukan dari berbagai macam sudut	Waktu pemrosesan tidak lebih cepat karena tidak <i>real-time</i>	Akurasi sistem kurang akurat karena deteksi pohon berdasarkan citra satelit yang tidak <i>real-time</i>

3.1.3 Sistem yang akan Dikembangkan

Sistem yang akan dikembangkan adalah solusi 1 yaitu pengimplementasian sistem menggunakan *drone* secara *real-time* untuk mendeteksi tinggi dan jenis pohon. Sistem tersebut dipilih karena dapat memudahkan pengguna dalam proses pendeteksian. Pengguna dapat mendeteksi banyak pohon sekaligus. Selain itu sistem juga memenuhi aspek – aspek karakteristik yang telah ditentukan.

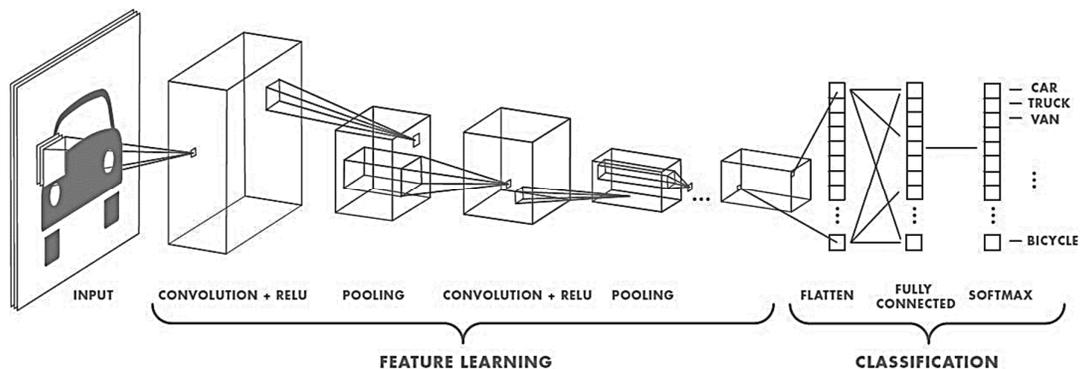
3.2 Modul Yang Memenuhi Fungsi dan Spesifikasi

3.2.1 Computer Vision

Computer vision merupakan bidang ilmu yang mempelajari tentang bagaimana mesin mengenali sebuah objek. Sebelumnya objek diobservasi untuk mendapatkan data dengan mengekstraksi info dari gambar. *Computer vision* bersifat visual karena semua data yang direpresentasikan bekerja seperti meniru sistem kerja visual dari manusia. *Computer vision* gabungan antara pengolahan citra (*image processing*) dan pengenalan pola (*pattern recognition*) [11].

3.2.2 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu subset dari *machine learning*. CNN merupakan salah satu dari berbagai jenis *artificial neural network* yang digunakan untuk berbagai aplikasi dan jenis data. CNN adalah jenis arsitektur jaringan untuk algoritma *deep learning* yang khusus digunakan untuk pengenalan gambar dan tugas-tugas yang melibatkan pemrosesan data piksel. Hal ini membuatnya sangat cocok untuk tugas-tugas *computer vision* (CV) dan untuk aplikasi di mana pengenalan objek sangat penting, seperti pengenalan gambar, klasifikasi objek, dan pengenalan pola. CNN terdiri dari tiga layer yaitu *convolutional layer*, *pooling layer*, dan *fully connected layer* [12].



Gambar 3.4 Arsitektur CNN

3.2.2.1 Convolution Layer

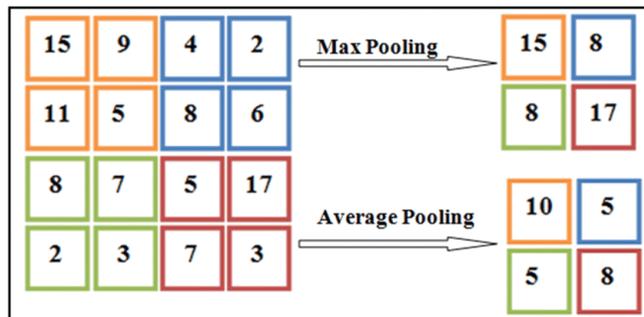
Convolutional layer adalah komponen utama dari CNN. *Convolutional layer* menggunakan kernel konvolusi sebagai filter untuk bergeser pada citra. Nilai dari setiap piksel yang dilewati oleh filter akan di digandakan dan ditambahkan kepada hasil konvolusi. Persamaan dari proses konvolusi didapatkan dari Persamaan [13] :

$$h(x) = f(x) \times g(x) \tag{3.1}$$

Dimana $f(x)$ dengan $g(x)$ adalah citra dari input dan filter. Penggunaan *stride* dan *padding* menjadi salah satu faktor dari hasil konvolusi. *Stride* sendiri merupakan jumlah pergeseran filter terhadap piksel pada suatu citra dan *padding* merupakan parameter yang digunakan untuk menambah jumlah piksel yang terdapat pada setiap sisi pada suatu citra sehingga pergeseran filter dapat dilakukan dengan optimal [13].

3.2.2.2 Pooling Layer

Pooling layer digunakan untuk mengurangi dimensi yang dihasilkan dari *convolution layer*. Ada tiga jenis teknik *pooling*, yaitu *general pooling*, *overlapping pooling* and *Spatial Pyramid Pooling* (SPP). Namun, yang paling sering digunakan ialah *general pooling* yang memiliki dua teknik, yaitu *max pooling* dan *average pooling*, kedua teknik tersebut direpresentasikan pada Gambar 3.5. *Max pooling* merupakan teknik *pooling* yang mengambil nilai maksimum pada filter yang bergeser. Sementara pada *average pooling*, menggunakan nilai rata-rata nilai yang terdapat pada filter[14].



Gambar 3.5 Max pooling dan average pooling

3.2.2.3 Fully-Connected Layer

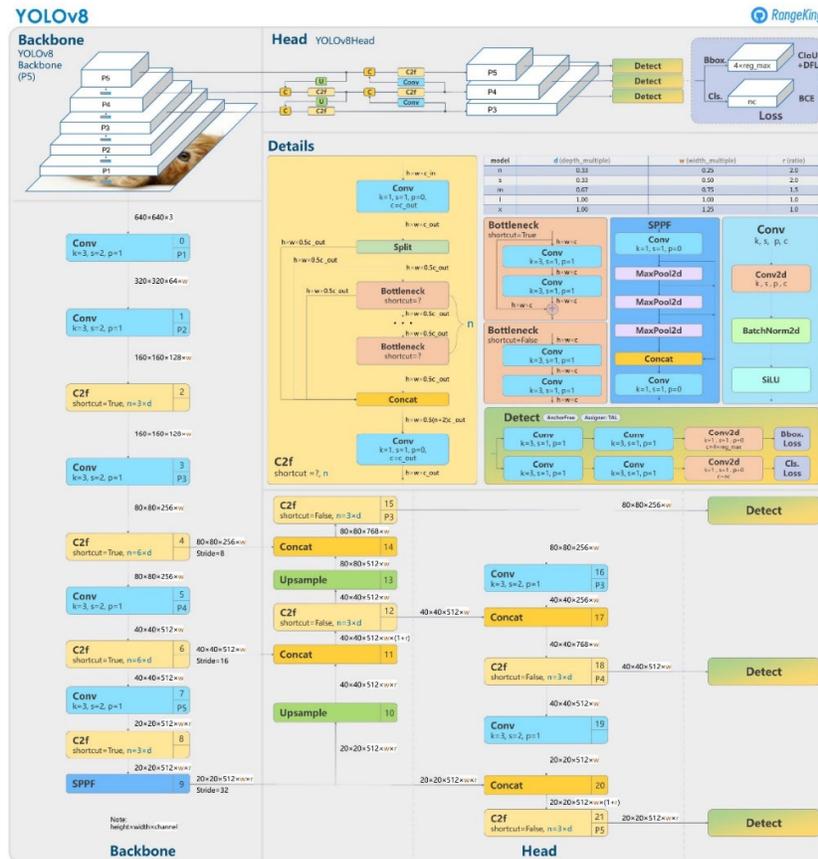
Fully-Connected Layer berfungsi untuk mengklasifikasi data masukan. *Feature map* yang dihasilkan dari *Convolution Layer* masih berbentuk multidimensi, sehingga harus dilakukan proses *flatten* atau *reshape feature map* untuk mentransformasikan data menjadi satu dimensi agar bisa dijadikan input pada *Fully-Connected Layer* dan dapat diklasifikasi secara linear. Saat

proses *flatten*, nilai input matriks dari layer sebelumnya akan diubah menjadi *vector*. Dalam proses ini, ditambahkan metode *dropout* untuk menghindari *overfitting* dengan cara menonaktifkan beberapa *edge* yang terhubung di setiap neuron [14].

3.2.3 YOLO (You Only Look Once)

YOLO (*You Only Look Once*) adalah algoritma *Object Detection* secara *real-time* yang umum digunakan. YOLO terus dikembangkan dari versi pertama dari Joseph Redmon, sampai saat ini yang terbaru adalah YOLOv8 yang dirilis secara resmi pada Januari 2023. Hasil dari riset yang dilakukan terkait perbandingan dengan versi sebelumnya, YOLOv8 menghasilkan mAP yang lebih baik daripada YOLO versi sebelumnya [15]. Algoritma YOLO membagi semua gambar *input* yang diberikan ke dalam sistem *grid* $S \times S$. Setiap *grid* bertanggung jawab untuk mendeteksi objek. Sel-sel *grid* itu memprediksi *bounding boxes* untuk objek yang terdeteksi. Untuk setiap *box* memiliki lima atribut utama yaitu x dan y untuk koordinat, w dan h untuk lebar dan tinggi objek, dan *confidence score* untuk probabilitas bahwa *box* tersebut berisi objek [16].

3.2.3.1 YOLOv8



Gambar 3.6 Struktur jaringan YOLOv8

YOLOv8 dirilis pada tahun 2023 oleh Glenn Jocher (*Founder & CEO of Ultralytics*). YOLOv8 adalah salah satu algoritma *object detection* terbaru untuk seri YOLO. Sebagai algoritma terbaru dalam seri YOLO saat ini, Struktur jaringan YOLOv8 terdiri dari beberapa lapisan seperti lapisan *input*, jaringan *backbone*, jaringan *neck* dan lapisan *output* [17]. YOLOv8 memiliki beberapa layer yang membangun struktur jaringannya seperti layer Conv, C2f, Concat, Upsample, SPPF. *Layer* C2f pada YOLOv8 berisi CSPBottleneck dengan 3 proses konvolusi yang berfungsi untuk memelihara fitur melalui propagasi, mendorong jaringan untuk menggunakan kembali fiturnya, mengurangi jumlah parameter jaringan dan membantu mempertahankan fitur yang halus untuk diteruskan ke lapisan yang lebih dalam dengan lebih efisien [17].

Kemudian sebelum masuk ke bagian *neck*, fitur-fitur akan diproses di *layer* SPPF (*Spatial Pyramid Pooling Fast*) yang terdiri dari *layer* conv dan *layer* max pooling dan SPPF berfungsi meningkatkan kecepatan komputasi lebih dari dua kali lipat [18]. YOLOv8 menggunakan CSPDarknet53 sebagai *backbone* dimana *backbone* ini dapat mengurangi kecepatan inferensi, meningkatkan akurasi dan mengurangi ukuran sebuah model dengan mengurangi total parameter kemudian menggunakan *Path Aggregation Network* (PANet) sebagai jaringan *neck* yang berfungsi untuk meningkatkan arus informasi. PANet mengadopsi *Feature Pyramid Network* (FPN) baru yang mencakup beberapa lapisan dari bawah ke atas dan atas ke bawah. Hal ini dapat meningkatkan penyebaran fitur tingkat rendah dalam model dan juga meningkatkan akurasi lokalisasi objek. Bagian output pada YOLOv8 mengadopsi head dari YOLOv5 yang terdiri dari 3 *scale detection head* [18]. Gambar 3.6 merupakan struktur jaringan dari algoritma YOLOv8.

3.2.4 Intersection Over Union (IOU)

Intersection over union (IOU) adalah sebuah istilah yang digunakan untuk mengukur nilai tumpang tindih antara dua kotak. Pada bidang deteksi objek IOU digunakan untuk mengevaluasi *bounding box* prediksi dan *bounding box* yang sebenarnya. Hal ini menentukan seberapa benar prediksi yang dilakukan saat melakukan deteksi objek. Nilai IOU dapat dihitung dengan Persamaan [19]

$$IOU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \quad (3.2)$$

Pada Persamaan 3.2 menunjukkan bahwa perolehan nilai IOU didapat dari cakupan antara kedua *bounding box* prediksi dan ground truth yang saling tumpang tindih (*Area Of*