

Pengembangan Metode Perhitungan Tinggi Pohon Berbasis Teknologi Drone dan Algoritma YOLOv8 dengan Pendekatan Skala Piksel Bounding Box

1st M.Haikal Febrian P
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

haikalphona@student.telkomuniver
sity.ac.id

2th Casi Setianingsih
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

Setiacasie@telkomuniversity.ac.id

3th Marisa W. Paryasto
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

marisa.paryasto@gmail.com

Abstrak — Pohon tak hanya berperan dalam industri kayu, tetapi juga mengatur siklus karbon global dan dampak perubahan iklim. Hubungan antara tinggi dan jenis pohon penting dalam struktur hutan. Data tinggi pohon bukan hanya mengungkap keragaman dan struktur hutan, tetapi juga produktivitas, evolusi, dan pertumbuhan pohon. Ini membantu ilmuwan, ahli kehutanan, dan pengambil kebijakan dalam manajemen sumber daya hutan dan pelestarian lingkungan. Ada beragam metode ukur tinggi pohon. Namun, pengukuran langsung dan tidak langsung lambat, kompleks, dan mahal. Solusinya, teknologi drone (UAV) dengan algoritma YOLOv8. Penelitian menunjukkan drone dan YOLOv8 efisien dan akurat mengukur dan klasifikasikan tinggi pohon. Model ini punya performa baik: presisi 88.57%, recall 86.14%, presisi rata-rata di atas 93.98% pada level 50%, serta sekitar 68.10% pada level 50-90%. Keakuratan terlihat dari skor kepercayaan 87%, error 36%, dan akurasi 64%. Teknologi ini berguna untuk pemantauan hutan, pelestarian alam, dan dampak perubahan iklim di hutan. Perlu pengembangan algoritma dan aspek teknis untuk masa depan.

Kata kunci— Drone, Kecerdasan Buatan, Computer Vision, YOLOv8, Pengukuran Tinggi Pohon

I. PENDAHULUAN

Dalam era teknologi modern, pemanfaatan drone, kecerdasan buatan (AI), dan teknologi computer vision telah menjadi pusat perhatian di berbagai bidang¹². Penelitian ini mengungkapkan implementasi kombinasi teknologi tersebut dalam pengukuran dan klasifikasi tinggi pohon dengan cepat dan akurat, membawa dampak positif dalam pemantauan lingkungan dan pengelolaan sumber daya alam yang berkelanjutan.

Tinggi pohon memainkan peran kunci dalam ekosistem dan penanganan isu perubahan iklim. Data yang akurat mengenai tinggi pohon memiliki implikasi yang signifikan dalam berbagai studi lingkungan, konservasi hutan, dan pengelolaan sumber daya alam. Namun, metode pengukuran konvensional seperti pengukuran langsung atau tidak langsung memiliki keterbatasan dalam hal waktu dan tenaga manusia.

Untuk mengatasi hambatan ini, penelitian ini menerapkan pemanfaatan teknologi drone dan algoritma YOLOv8 yang dioptimalkan dalam bidang computer vision untuk mendeteksi pohon secara otomatis dalam gambar atau video real-time. Teknologi drone telah mengalami kemajuan pesat, memungkinkan akses ke wilayah-wilayah yang sebelumnya sulit dijangkau. Kombinasi ini dengan kecerdasan buatan dan algoritma YOLOv8 memungkinkan pengidentifikasian pohon secara otomatis dalam berbagai kondisi visual, termasuk kemampuan untuk mengenali jenis pohon yang berbeda.

Meskipun terdapat tantangan dalam mengumpulkan dataset yang akurat untuk melatih model AI, perkembangan terus-menerus dalam teknologi drone dan kecerdasan buatan mengindikasikan potensi besar dalam meningkatkan efisiensi dan akurasi analisis tinggi pohon. Hasil penelitian ini menunjukkan bahwa penerapan drone dan YOLOv8 dapat menghasilkan pengukuran dan klasifikasi tinggi pohon dengan efisien. Model yang telah dikembangkan berhasil mencapai performa yang baik dengan nilai akurasi, presisi, recall, dan mean Average Precision (mAP) yang signifikan.

Dengan pencapaian nilai-nilai ini, teknologi ini memiliki potensi yang luas dalam mendukung berbagai aplikasi, termasuk namun tidak terbatas pada pemantauan pertumbuhan hutan, konservasi alam, serta evaluasi dampak perubahan iklim terhadap ekosistem hutan. Keunggulan utama dari pendekatan ini adalah kemampuannya untuk menghemat waktu dan tenaga manusia, sambil memberikan data yang akurat dan relevan untuk keperluan analisis lingkungan.

Penting untuk terus mengembangkan dan memperbaiki dataset yang digunakan untuk melatih model AI agar akurasi dan kemampuannya terus meningkat. Dengan demikian, teknologi ini dapat menjadi alat yang semakin efektif dalam menjawab tantangan lingkungan dan mendukung upaya global menuju pengelolaan sumber daya alam yang lebih berkelanjutan. Dengan inovasi terus menerus, kombinasi drone, kecerdasan buatan, dan computer vision memiliki potensi besar untuk merubah

paradigma dalam cara kita memahami dan mengelola lingkungan kita.

II. KAJIAN TEORI

A. Algoritma 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, YOLOv8 memiliki beberapa ukuran untuk ukurannya yaitu YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l dan YOLOv8x1 yang masing-masing mewakili *Nano, Small, Medium, Large* dan *Xtra large*. [1]. Struktur jaringan YOLOv8 terdiri dari beberapa lapisan seperti lapisan *input*, jaringan *backbone*, jaringan *neck* dan lapisan *output* [1]. YOLOv8 memiliki beberapa layer yang membangun struktur jaringannya seperti layer Conv, C2f, Concat, Upsample, SPPF.

B. Konsep Perhitungan Tinggi Pohon dengan Perbandingan Skala Piksel

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]

Pada Persamaan 2.2 menunjukkan bahwa perolehan nilai IOU didapat dari cakupan antara kedua bounding box prediksi dan ground truth yang saling tumpang tindih (Area Of Intersection) dibagi total luas dari kedua bounding box tersebut. Tujuan utama dari penggunaan IOU adalah untuk meningkatkan prediksinya sampai kedua bounding box tertumpang tindih secara sempurna, yaitu IOU antara dua kotak menjadi sama dengan 1[2].

1. Precision

Precision merupakan rasio prediksi benar positif dibagi dengan keseluruhan hasil yang diprediksi positif. Precision dapat dihitung menggunakan persamaan[3]

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3.3)$$

dengan TP merupakan True Positive dan FP merupakan False Positive.

2. Recall

Recall merupakan rasio prediksi benar positif dibagi dengan keseluruhan data yang benar positif. Recall dapat dihitung menggunakan persamaan[3]

$$\text{Recall} = \frac{TP}{TP + FN}$$

dengan TP merupakan True Positive dan FN merupakan False Negative

3. mean Average Precision (mAP)

mean Average Precision (mAP) adalah metrik evaluasi lain yang diterapkan secara luas untuk deteksi objek kategori tertentu. menunjukkan tingkat kinerja dari proses deteksi objek dan lokasinya dan mAP dapat dihitung dari

rata-rata average precision untuk semua kelas Nilai mAP dihitung menggunakan persamaan [3]

$$mAP = \frac{1}{n} \sum_{i=1}^N AP_i \quad (3.5)$$

dengan N merupakan jumlah kelas dan AP merupakan Average Precision. Average Precision digunakan untuk menghitung nilai rata-rata precision pada berbagai tingkat recall. Semakin tinggi nilai AP maka semakin baik kinerjanya begitu juga sebaliknya. Untuk nilai dari Average Precision dapat dihitung dengan menggunakan persamaan [3].

$$AP = \sum_n (R_n - R_{n-1}) P_n \quad (3.6)$$

dengan R_n dan P_n adalah recall dan precision pada threshold ke-n.

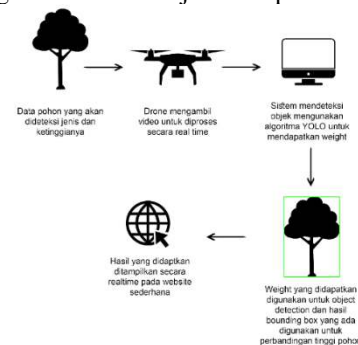
C. Penerapan Informasi Perbandingan untuk Akurasi Tinggi dalam Pengukuran

Proses Perhitungan tinggi pohon memerlukan perbandingan skala antara piksel bounding box dengan ukuran asli. Nilai piksel bounding box terdiri dari panjang dan lebar, untuk pengukuran tinggi maka nilai panjang piksel bounding box yang digunakan. Untuk mengukur tinggi pohon diperlukan mencari jarak optimal dari pendeteksian tinggi pohon, jarak optimal tersebut menjadi acuan untuk melakukan capture suatu benda sebagai pembanding. Bounding box yang muncul pada benda pembanding dihitung nilai panjang pikselnya, lalu dibandingkan dengan tinggi sesungguhnya. Berikut adalah gambaran untuk memperoleh nilai perbandingan ukuran panjang piksel dengan tinggi sesungguhnya.

III. METODE

A. Pengambilan Video Menggunakan Drone

Setelah mendapatkan model terbaik maka *weight* yang didapatkan digunakan untuk *deploy* sistem *object detection*. Hasil deteksi objek berupa prediksi dari pohon pinus, palm, dan pisang. Pada tahap ini perlu diperhatikan nilai *confidence* dari tiap objek yang terdeteksi. Nilai *confidence* nantinya dapat di atur pada website sederhana untuk memaksimalkan kerja sistem. Alur pengambilan video menggunakan drone dijelaskan pada Gambar 1.



GAMBAR 1
Pengambilan Video Menggunakan Drone

B. Pemrosesan Video Menggunakan Pemodelan Algoritma YOLOv8

1. Dataset

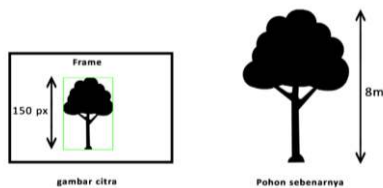
Dataset yang digunakan oleh sistem ini adalah gambar dengan format jpg, jpeg, dan png. Proses akuisisi dataset dilakukan dengan cara mencari gambar pohon yang telah ditentukan yaitu pisang-pisangan, cemara-cemaraan, dan palem-paleman. Pencarian gambar pohon dilakukan melalui berbagai macam website seperti pinterest, facebook, google image, dan lain-lain. Beberapa data gambar juga didapatkan dengan cara mengambil gambar pohon secara langsung dilapangan. Adapun lable pada dataset ini sesuai dengan jenis pohonnya yaitu pisang, palm, dan cemara. Total dataset yang digunakan berjumlah 1600 data yang terdiri dari 400 data pohon palem, 400 data pohon pisang dan 400 data pohon cemara, serta 400 data null.

2. Preprocessing Data

Pre-processing adalah proses perbaikan gambar untuk menyesuaikan dataset sebagai input training model. *Pre-processing* dilakukan pada semua dataset yang telah dibuat sebelumnya. Metode yang digunakan pada sistem ini adalah *auto-orient*, dan pengaturan *brightness*. *Auto-orient* digunakan untuk menjamin citra yang diunggah menjadi dataset yang sesuai dengan orientasi citra orisinalnya, *brightness* dilakukan dengan menambah atau mengurangi kecerahan gambar.

3. Pemodelan YOLOv8

Agar pengaplikasian model YOLO v8 bekerja dengan optimal pada sistem, terdapat beberapa faktor yang perlu diperhatikan yaitu nilai akurasi yang baik. Untuk mendapatkan nilai akurasi yang cukup baik maka model perlu dilatih menggunakan parameter yang berbeda beda. Parameter yang diuji untuk training antara lain perbandingan rasio data training, validasi, dan test, learning rate, epoch dan batch size. Jika semua tahap sudah dilakukan akan mendapatkan model terbaik yang akan diimplementasikan pada *object detection*.



GAMBAR 2 Ilustrasi Perbandingan Objek

IV. HASIL DAN PEMBAHASAN

A. Pengukuran Tinggi

Untuk mengukur tinggi pohon diperlukan nilai perbandingan. Nilai perbandingan didapatkan dengan cara membandingkan objek dengan ukuran yang sudah diketahui dibandingkan dengan ukuran piksel panjang bounding box-nya. Pada implementasi sistem yang dikembangkan, objek perbandingan merupakan banner cetak dengan gambar pohon pinus dengan panjang 4 meter. Yang selanjutnya akan diproses object detection untuk mendapatkan informasi

panjang dari bounding box yang terdeteksi. Gambar 3 merupakan implementasi pengukuran perbandingan.



GAMBAR 3 Pengukuran Perbandingan

Panjang *bounding box* terdeteksi sebesar 484 px, hasil tersebut diukur pada jarak pengukuran 5 meter dengan ketinggian 5 meter. Nilai panjang *bounding box* akan dihitung menggunakan persamaan di bawah ini untuk mendapatkan nilai perbandingannya menjadi :

$$\text{Perbandingan} = \frac{400 \text{ cm}}{484 \text{ px}} = 0.826$$

Sehingga 1 px = 0.826 cm. Hasil tersebut disimpan menjadi data pembanding setiap object detection yang dilakukan pada jarak 5 meter. Selanjutnya menggunakan metode yang sama kembali menghitung perbandingan pada jarak yang lain yaitu 10, 15, dan 20 meter untuk dijadikan data parameter perbandingan pada jarak tersebut. Tabel 1 adalah hasil perbandingan dari tiap jarak yang dihiung.

TABEL 1 Perbandingan Panjang Bounding Box

No	Distance (m)	Comparison
1	5	0.826
2	10	1.223
3	15	1.843
4	20	2.439

Setelah mendapatkan nilai perbandingan, tinggi pohon lain dapat dihitung menggunakan persamaan di bawah ini

$$\text{Tinggi Pohon} = \text{Perbandingan} \times \text{Nilai Piksel Panjang}$$

menggunakan perbandingan sesuai dengan jarak pengambilan video yang sama. Gambar 4 adalah proses pengukuran tinggi yang dilakukan sistem.



GAMBAR 4 Ilustrasi Pengukuran Tinggi

Pohon tersebut diukur pada jarak 10 meter, dan hasil dari object detection menunjukkan panjang bounding box sebesar 504 px. Maka persamaan untuk mengukur ketinggian sesungguhnya adalah :

$$\begin{aligned} \text{Tinggi Pohon Asli} &= 504 \text{ px} \times 1.223 \\ \text{Tinggi Pohon Asli} &= 616.392 \text{ cm} \end{aligned}$$

Untuk meminimalisir nilai eror pada pengukuran maka dibuat list berupa range seperti pada Tabel 2 berikut:

TABEL 2
Indeks Tinggi Pohon

Panjang Terukur	Nilai Akhir
0 – 99	Terlalu pendek
100 – 199	1-2 meter
200 – 299	2-3 meter
300 – 399	3-4 meter
400 – 499	4-5 meter
500 – 599	5-6 meter
600 – 699	6-7 meter
700 – 799	7-8 meter
800 – 899	8-9 meter
900 – 999	9-10 meter
1000 – 1099	10-11 meter
1100 – 1199	11-12 meter
1200 – 1299	12-13 meter
1300 – 1399	13-14 meter
1400 – 1499	14-15 meter
1500 – ∞	Terlalu tinggi

Setiap nilai yang terukur akan dikonversi kedalam satuan meter dan diberi range nilai untuk estimasi ketinggian yang dideteksi. Pohon pada ketinggian kurang dari 1 meter akan diberikan label terlalu pendek, dan pohon dengan ketinggian lebih dari 15 meter diberi label terlalu tinggi. Sehingga hasil akhirnya adalah sebagai berikut pada Gambar 5.



GAMBAR 5
Hasil Akhir Pendeteksian

V. KESIMPULAN

Pada implementasi sistem tree height detection using drone dilakukan menggunakan model YOLOv8s. Model terbaik yang digunakan adalah model hasil training dengan partisi data sebesar 80% data train, 10% data validation, dan 10% data test. Model tersebut ditraining menggunakan nilai parameter 8 batch size, 0.001 learning rate, dan 100 epoch. Model terbaik tersebut mendapatkan nilai performance sebesar 88.57% precision, 86.14% recall, 93.98% mAP50 dan 68.10% mAP50-90.

REFERENSI

- [1] G. Jocher, “ultralytics/ultralytics: NEW - YOLOv8 🚀 in PyTorch > ONNX > OpenVINO > CoreML > TFLite.” <https://github.com/ultralytics/ultralytics> (accessed Aug. 04, 2023).
- [2] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, “Generalized intersection over union: A metric and a loss for bounding box regression,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 658–666.
- [3] A. Shafi, “What are the definitions of Precision and Recall? | Towards Data Science.” <https://towardsdatascience.com/precision-and-recall-88a3776c8007> (accessed Aug. 05, 2023).