

**DETEKSI DAN ESTIMASI JUMLAH POHON SAWIT MENGGUNAKAN
ALGORITMA *DEEFOREST* PADA KEBUN SAWIT MESUJI**

(Skripsi)

Oleh:

**Reginia Putri Maharani
2115061004**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2026**

**DETEKSI DAN ESTIMASI JUMLAH POHON SAWIT MENGGUNAKAN
ALGORITMA *DEEFOREST* PADA KEBUN SAWIT MESUJI**

Oleh

REGINIA PUTRI MAHARANI

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA TEKNIK**

Pada

**Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2026

ABSTRAK

DETEKSI DAN ESTIMASI JUMLAH POHON SAWIT MENGGUNAKAN ALGORITMA *DEEFOREST* PADA KEBUN SAWIT MESUJI

Oleh

REGINIA PUTRI MAHARANI

Pendekatan berbasis citra udara diterapkan untuk mendeteksi kanopi, menghitung jumlah, dan menentukan titik koordinat pohon kelapa sawit secara otomatis menggunakan algoritma *DeepForest* berbasis citra udara. Seluruh citra kebun sawit diproses menggunakan QGIS dengan pembagian area menjadi tujuh zona dan dianotasi secara manual menggunakan bounding box pada Roboflow, sehingga menghasilkan 78.783 anotasi kanopi pohon sebagai ground truth untuk pelatihan dan evaluasi model. Model *DeepForest* dilatih pada *google colab* dan dievaluasi menggunakan *confusion matrix* dengan metrik *precision*, *recall*, dan *F1-score*. Hasil evaluasi menunjukkan nilai *precision* sebesar 0,9357, *recall* sebesar 0,9433, dan *F1-score* sebesar 0,9395, yang menandakan performa deteksi yang baik. Model ini berhasil mendeteksi total 73.278 pohon sawit di seluruh zona, sehingga hasil penelitian dapat dimanfaatkan sebagai dasar integrasi ke dalam sistem WebGIS untuk mendukung pengelolaan dan pemantauan kebun kelapa sawit secara efisien.

Kata Kunci: *DeepForest*, deteksi pohon sawit, citra udara, *bounding box*

ABSTRACT

DETECTION AND ESTIMATION OF THE NUMBER OF OIL PALM TREES USING THE DEEFOREST ALGORITHM IN THE MESUJI OIL PALM PLANTATION

By

REGINIA PUTRI MAHARANI

An aerial image-based approach was applied to detect canopies, count trees, and determine the coordinates of oil palm trees automatically using the DeepForest aerial image-based algorithm. All oil palm plantation images were processed using QGIS by dividing the area into seven zones and manually annotated using bounding boxes in Roboflow, resulting in 78,783 tree canopy annotations as ground truth for model training and evaluation. The DeepForest model was trained on Google Colaboratory and evaluated using a confusion matrix with precision, recall, and F1-score metrics. The evaluation results showed a precision value of 0.9357, a recall of 0.9433, and an F1-score of 0.9395, indicating good detection performance. This model successfully detected a total of 73,278 oil palm trees across all zones, so the research results can be used as a basis for integration into the WebGIS system to support efficient oil palm plantation management and monitoring.

Keywords: DeepForest, oil palm detection, aerial imagery, bounding box

Judul Skripsi : DETEKSI DAN ESTIMASI JUMLAH POHON
SAWIT MENGGUNAKAN ALGORITMA
DEEFOREST PADA KEBUN SAWIT MESUJI

Nama Mahasiswa : *Reginia Putri Maharani*

Nomor Pokok Mahasiswa : 2115061004

Program Studi : S1 Teknik Informatika

Jurusan : Teknik Elektro

Fakultas : Teknik



1. Komisi Pembimbing

Pembimbing Utama

Pembimbing Pendamping

Mona Arif Muda
Mona Arif Muda, S.T., M.T.
NIP. 197111122000031002

Ir. Gigh Forda Nama
Ir. Gigh Forda Nama, S.T., M.T.I., IPM
NIP. 198307122008121003

2. Mengetahui

Ketua Jurusan
Teknik Elektro

Ketua Program Studi
Teknik Informatika

Herlinawati
Herlinawati, S.T., M.T.
NIP. 197103141999032001

Yessi Mulyani
Yessi Mulyani, S.T., M.T.
NIP. 197312262000122001

MENGESAHKAN

1. Tim Penguji

Ketua : **Mona Arif Muda, S.T., M.T.**

Sekretaris : **Ir. Gigih Forda Nama, S.T., M.T.I., IPM**

Penguji : **Yessi Mulyani, S.T., M.T.**



2. Dekan Fakultas Teknik



Dr. H. Ahmad Herison, S.T., M.T.
NIP. 196910302000031001

Tanggal Lulus Ujian Skripsi : **29 Januari 2026**

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini, menyatakan dengan sebenarnya bahwa skripsi saya yang berjudul “Deteksi Dan Estimasi Jumlah Pohon Sawit Menggunakan Algoritma *Deepforest* Pada Kebun Sawit Mesuji” merupakan hasil karya saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi saya ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 13 Februari 2026

Pembuat Pernyataan



Reginia Putri Maharani

2115061004

RIWAYAT HIDUP



Reginia Putri Maharani lahir di kota Bandar Lampung, Provinsi Lampung, pada tanggal 30 oktober 2003. Penulis merupakan anak pertama dari dua bersaudara, putri dari pasangan Bapak Ajamuddin dan Ibu Istanawati.

Riwayat pendidikan dimulai dari Taman Kanak-Kanak (TK) Islam Al-Furqon diselesaikan pada tahun 2009. Kemudian dilanjutkan ke Sekolah Dasar Swasta Islam Al-Furqon diselesaikan pada tahun 2015. Pendidikan menengah pertama dilanjutkan di SMP Negeri 4 Tulang Bawang Tengah dan lulus pada tahun 2018. Selanjutnya, Pendidikan menengah atas diselesaikan di SMA Negeri 1 Tulang Bawang Tengan pada tahun 2021.

Pada tahun 2021, penulis diterima sebagai mahasiswa Program Studi Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung melalui jalur SNMPTN. Selama masa perkuliahan, penulis aktif mengikuti berbagai kegiatan pengembangan diri dan organisasi. Penulis menjadi anggota Himpunan Mahasiswa Teknik Elektro (HIMATRO) pada periode 2021–2023, mengikuti program Merdeka Belajar Kampus Merdeka (MBKM) Studi Independen sebagai cyber security di PT Metrodata Academy, serta menjalani praktik kerja lapangan (PKL) di PT Angkasa Pura II KC Bandar Udara Radin Inten pada periode Juni hingga Agustus 2024. Selain itu, penulis juga melaksanakan Kuliah Kerja Nyata (KKN) di Desa Sidomulyo, Kecamatan Mesuji, Kabupaten Mesuji, Provinsi Lampung.

MOTTO

“Allah tidak mengatakan hidup ini mudah. Tetapi Allah berjanji, bahwa
sesungguhnya bersama kesulitan ada kemudahan”

(Q.S Al-Insyirah: 5-6)

“Allah menaruhmu ditempat mu yang sekarang bukan karena kebetulan, tapi
allah telah memnentukan jalan terbaik untukmu, allah sedang melatih mu untuk
menjadi kuat dan hebat”

(Ustadzah Halimah Alaydrus)

“Bergerak dengan usaha, menunggu dengan sabar, menerima dengan ikhlas, maka
hasil terbaik akan hadir di setiap langkah”

(Anonymus)

“Jangan menyesali apa yang telah berlalu, jalani dengan sepenuh hati apa yang
sedang dihadapi saat ini, dan jangan khawatir terhadap apa yang akan datang”

(Anonymus)

PERSEMBAHAN

Alhamdulillahirabbil‘alamin, segala puji bagi Allah SWT yang telah melimpahkan rahmat dan karunia-Nya sehingga skripsi ini dapat diselesaikan dengan baik. Shalawat serta salam semoga senantiasa tercurah kepada Nabi Muhammad SAW, teladan sepanjang masa.

Dengan tulus, kupersembahkan skripsi ini kepada

Kedua orang tuaku tercinta, Bapak Ajamuddin dan Ibu Istanawati. Terima kasih atas doa-doa yang tak pernah putus, kasih sayang yang tulus, dukungan, serta pengorbanan tanpa batas yang tidak dapat terbalaskan. Setiap tetes keringat dan doa kalian menjadi cahaya penerang jalan hingga akhirnya aku mampu berdiri di titik ini. Terima kasih telah senantiasa menemani setiap proses yang aku jalani dan setia mendampingi hingga aku mencapai tujuan.

Adikku tersayang, Ralisa Dara Mahatidana, yang selalu menjadi penyemangat, pendengar setiap keluh kesah, dan tidak pernah lupa memberikan ucapan selamat atas setiap pencapaianku. Semoga Allah SWT senantiasa melimpahkan berkah dan perlindungan-Nya kepada kita semua, Aamiin.

Rekan seperjuangan, seluruh dosen dan civitas akademik Jurusan Teknik Elektro, Universitas Lampung, terima kasih atas semangat, diskusi, ilmu, bimbingan, dan kebersamaan yang menjadikan perjalanan ini sangat bermakna.

Dan terakhir, untuk diriku sendiri, Reginia Putri Maharani, yang telah bertahan, berjuang, dan tak pernah berhenti percaya bahwa semuanya bisa diselesaikan.

SANWACANA

Alhamdulillahirabbil'alamin, puji syukur penulis panjatkan ke hadirat Allah SWT atas rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi berjudul “Deteksi Dan Estimasi Jumlah Pohon Sawit Menggunakan Algoritma *Deepforest* Pada Kebun Sawit Mesuji”. Shalawat dan salam semoga tercurah kepada Nabi Muhammad SAW. Skripsi ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Informatika, Fakultas Teknik, Universitas Lampung. Penyusunan skripsi ini tidak terlepas dari bantuan, bimbingan, dan dukungan berbagai pihak. Oleh karena itu, dengan penuh rasa syukur penulis menyampaikan terima kasih kepada:

1. Ibu Herlinawati, S.T., M.T., selaku Ketua Jurusan Teknik Elektro Universitas Lampung.
2. Ibu Yessi Mulyani, S.T., M.T., selaku Ketua Program Studi Teknik Informatika, pembimbing akademik, sekaligus penguji, atas saran, masukan, serta ilmu yang telah diberikan demi perbaikan dan penyempurnaan penelitian ini.
3. Bapak Mona Arif Muda, S.T., M.T., selaku Pembimbing Utama, atas bimbingan, bantuan, dan arahan yang sangat berarti selama proses penelitian hingga terselesaikan dengan baik.
4. Bapak Ir. Gigih Forda Nama, S.T., M.T.I., IPM., selaku Pembimbing Pendamping, atas masukan dan arahan yang diberikan dalam penyusunan penelitian ini.
5. Terutama kedua orang tua tercinta, Bapak Ajamuddin dan Ibu Istanawati, serta adikku, atas doa, dukungan, semangat, dan bantuan materil yang tiada henti hingga skripsi ini dapat terselesaikan dengan baik.
6. Kepada diri sendiri, atas keteguhan dan usaha dalam menyelesaikan skripsi ini meskipun menghadapi berbagai rintangan.

7. Kepada rekan penelitian yang luar biasa: Tyas Nafara Andini, Vidya Adelia Anya, Azahra Nafisa, terima kasih telah berjalan bersama sejak awal penyusunan skripsi dengan segala lika-likunya.
8. Kepada sahaba-sahabat seperantauan yang sudah seperti keluarga: Murti, Chandrika, Vania, Pinka, Annisya, Hilda, Syifa serta teman-teman seperjuangan lainnya yang tidak dapat disebutkan satu per satu.
9. Kepada teman-teman masa SMA: Musyarifah Aini, Azzahra nur mauliza, Ema Feronika, Okky Nurjanah dan Novita Sari, yang selalu menjadi penghibur dan penyemangat di tengah berbagai tantangan.
10. Seluruh civitas akademik Jurusan Teknik Elektro Universitas Lampung, atas segala bantuan administratif dan dukungan selama proses penelitian.
11. Dan kepada semua pihak yang tidak dapat disebutkan satu per satu, terima kasih atas kontribusi, baik secara langsung maupun tidak langsung, dalam penyelesaian skripsi ini.

Penelitian ini masih jauh dari kata sempurna, dikarenakan terbatasnya pengalaman dan pengetahuan yang dimiliki. Oleh karena itu, diharapkan segala bentuk saran serta masukan bahkan kritik yang membangun dari berbagai pihak. Semoga skripsi ini dapat bermanfaat bagi para pembaca dan semua pihak khususnya dalam bidang pengembangan machine learning.

Bandar Lampung, 25 Januari 2026
Penulis

Reginia Putri Maharani
NPM.2115061004

DAFTAR ISI

DAFTAR ISI	xiii
DAFTAR GAMBAR	xvi
DAFTAR TABEL	xviii
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian.....	3
1.5 Batasan Masalah.....	3
1.6 Sistematika Penulisan.....	4
II. TINJUAN PUSTAKA	5
2.1 Kelapa Sawit.....	5
2.2 <i>Artificial Intelligence</i>	5
2.3 <i>Deep Learning</i>	6
2.4 <i>Artificial Neural Network</i>	7
2.6 Citra	13
2.7 <i>DeepForest</i>	15
2.8 <i>Confusion Matrix</i>	17
2.9 <i>PyTorch</i>	19
2.10 <i>Google Colaboratory</i>	19
2.11 <i>Quantum Geographic Information System (QGIS)</i>	20

2.12 Roboflow.....	20
2.13 Penelitian Terkait	21
III. METODOLOGI PENELITIAN.....	51
3.1 Waktu dan Tempat	51
3.2 Capstone Project	51
3.3 Alat Dan Bahan	53
3.3.1 Alat.....	53
3.3.2 Bahan Penelitian	53
3.4 Arsitektur Perangkat Lunak.....	54
3.5 Alur Penelitian	56
3.5.1 Identifikasi Masalah.....	57
3.5.2 Pengumpulan Data.....	57
3.5.3 Persiapan Data	58
3.5.4 Modeling.....	60
3.5.5 Evaluasi.....	60
3.5.6 Penentuan Titik Koordinat.....	61
3.5.7 Output Yang di Hasilkan.....	61
IV. HASIL DAN PEMBAHASAN.....	63
4.1 Identifikasi Masalah	63
4.2 Pengumpulan Data	63
4.3 Persiapan Data.....	64
4.4 Modeling	84
4.5 Evaluasi	88
4.6 Penentuan Titik Koordinat.....	89
4.7 Outpun Yang di Hasilkan.....	92
V. KESIMPULAN DAN SARAN	101

5.1 Kesimpulan.....	101
5.2 Saran.....	102
DAFTAR PUSTAKA.....	103
LAMPIRAN.....	107

DAFTAR GAMBAR

Gambar 2.1 Penerapan Konsep Kecerdasan Buatan	6
Gambar 2.2 Jaringan Lapis Tunggal	8
Gambar 2.3 Jaringan Multilapis	9
Gambar 2.4 Jaringan Kompetitif.....	10
Gambar 2.5 Arsitektur DeepForest	16
Gambar 2.6 Hasil Penelitian Terkait Deteksi Pohon dari Berbagai Lokasi	29
Gambar 2.7 Hasil Penelitian Terkait Alur Model Deteksi Pohon	30
Gambar 2.8 Hasil Penelitian Terkait Evaluasi Model Deteksi Pohon.....	31
Gambar 2.9 Hasil Penelitian Terkait Visualisasi Anotasi Spesies Pohon	40
Gambar 2.10 Hasil Penelitian Terkait Sebaran Lokasi Data Kanopi Pohon.....	46
Gambar 2.11 Hasil Penelitian Terkait Pohon Pada Citra Satelit Area Perkotaan	48
Gambar 3.1 Captone Project.....	52
Gambar 3.2 Arsitektur Perangkat Lunak.....	54
Gambar 3.3 Alur Penelitian.....	56
Gambar 4.1 Peta Utama Kebun Sawit.....	65
Gambar 4.2 Pemetaan 7 Zona Kebun Sawit	66
Gambar 4.3 Zona 1.....	67
Gambar 4.4 Empat Potongan Gambar Zona 1	68
Gambar 4.5 Zona 2.....	69
Gambar 4.6 Empat Potongan Gambar Zona 2	69
Gambar 4.7 Zona 3.....	70
Gambar 4.8 Empat Potongan Zona 3	71
Gambar 4.9 Zona 4.....	72
Gambar 4.10 Empat Potongan Gambar Zona 4	73
Gambar 4.11 Zona 5.....	74
Gambar 4.12 Empat Potongan Gambar Zona 5	74

Gambar 4.13 Zona 6.....	75
Gambar 4.14 Empat Potongan Gambar Zona 6	76
Gambar 4.15 Zona 7.....	77
Gambar 4.16 Empat Potongan Gambar Zona 7	78
Gambar 4.17 Anotasi Boundingbox Roboflow	79
Gambar 4.18 File CSV Metadata	80
Gambar 4.19 Kode Program Persiapan Dataset.....	82
Gambar 4.20 Pembagian Dataset	83
Gambar 4.21 Menyimpan dan Menampilkan Data Anotasi.....	83
Gambar 4.22 Kode Program Modeling.....	84
Gambar 4.23 Proses Pelatihan Model Terputus	87
Gambar 4.24 Pelatihan Model <i>DeepForest</i> Epoch 25.....	88
Gambar 4.25 Metadata Koordinat Zona.....	90
Gambar 4.26 Kode Program Pembentukan Titik Koordinat.....	90
Gambar 4.27 Kode Penyimpanan Data Koordinat Semua Zona	91
Gambar 4.28 Penambahan Kolom Zona Pada Data Koordinat.....	92
Gambar 4.29 Visualisasi Jumlah Pohon Zona 1.....	93
Gambar 4.30 Visualisasi Jumlah Pohon Zona 2.....	94
Gambar 4.31 Visualisasi Jumlah Pohon Zona 3.....	95
Gambar 4.32 Visualisasi Jumlah Pohon Zona 4.....	96
Gambar 4.33 Visualisasi Jumlah Pohon Zona 5.....	97
Gambar 4.34 Visualisasi Jumlah Pohon Zona 6.....	98
Gambar 4.35 Visualisasi Jumlah Pohon Zona 7.....	99

DAFTAR TABEL

Tabel 2.1 Mengevaluasi klasifikasi segmen.....	18
Tabel 2.2 Hasil Penelitian Terkait Perhitungan Akurasi DeepForest	22
Tabel 2.3 Hasil Penelitian Terkait Evaluasi Detektor Mendeteksi Pohon.....	24
Tabel 2.4 Hasil Penelitian Terkait Perbandingan Akurasi Klasifikasi Pohon	26
Tabel 2.5 Hasil Penelitian Terkait Perbandingan DeepForest Dan ArcGIS Pro ...	27
Tabel 2.6 Hasil Penelitian Terkait Perbandingan DeepForest dan Yolo.....	33
Tabel 2.7 Hasil Penelitian Terkait Kinerja Model Deteksi Pohon	34
Tabel 2.8 Hasil Penelitian Terkait Data Lokasi Dan Kondisi Pohon	36
Tabel 2.9 Hasil Penelitian Terkait Evaluasi Deteksi Pohon	37
Tabel 2.10 Hasil Penelitian Terkait Tabel Performa Model	38
Tabel 2.11 Hasil Penelitian Terkait Evaluasi Deteksi Pohon Pinus	42
Tabel 2.12 Hasil Penelitian Terkait Deteksi Jumlah Pohon	44
Tabel 2.13 Hasil Penelitian Terkait Akurasi Pohon Menggunakan DeepForest ...	45
Tabel 2.14 Hasil Penelitian Terkait Akurasi Pohon Menggunakan DeepForest ...	49
Tabel 2.15 Hasil Penelitian Terkait Anotasi Deteksi Pohon.....	50
Tabel 3.1 Waktu Pelaksanaan.....	51
Tabel 3.2 Alat	53
Tabel 4.1 Hasil Pengujian Dengan Kombinasi Parameter Zona 1	85
Tabel 4.2 Hasil Pengujian Dengan Kombinasi Parameter Zona 1 dan Zona 2	86
Tabel 4.3 Hasil Evaluasi Performa Model DeepForest	89
Tabel 4.4 Jumlah Pohon	92
Tabel 4.5 File CSV Titik Koordinat	100

I. PENDAHULUAN

1.1 Latar Belakang

Kelapa sawit menjadi komoditas unggulan di Indonesia dalam puluhan tahun terakhir. Berdasarkan Data Direktorat Jenderal Perkebunan Tahun 2019 (Ditjenbun, 2019), total luas perkebunan kelapa sawit Indonesia pada 2017 mencapai 14 juta hektar. Di masa lalu, industri kelapa sawit didominasi oleh perusahaan besar negara dan swasta, tetapi sejak tahun 1984 perkebunan plasma kelapa sawit telah meningkat. Pada tahun 2002, total area kebun plasma mencapai 35% dari total area perkebunan kelapa sawit negara ini meningkat menjadi 42% pada tahun 2003, dan menjadi 41% pada tahun 2018. Perluasan kebun plasma telah memperluas keterlibatan para petani kecil dalam rantai pasokan minyak sawit global. Menurut data terakhir, dari 14 juta hektar perkebunan kelapa sawit di Indonesia pada 2017, 5,6 juta hektar (40%) adalah perkebunan rakyat (PR), 7,7 juta hektar (55%) adalah perkebunan besar swasta (PBS), dan 0,7 juta hektar (5%) adalah perkebunan besar negara (PBN). Dengan skema perkebunan plasma, perkebunan kelapa sawit di Indonesia tumbuh dari perusahaan dan perkebunan rakyat yang berkembang pesat [1].

Namun, seiring dengan meningkatnya luas dan peran perkebunan rakyat dalam industri kelapa sawit, Dibutuhkan efektivitas metode untuk menghitung jumlah pohon sawit. Sebelumnya metode manual dalam perhitungan pohon sawit dan lokasinya masih banyak digunakan, tetapi memiliki keterbatasan dalam hal efisiensi dan akurasi. Perhitungan jumlah pohon sawit secara manual umumnya dilakukan melalui survei lapangan dengan pencacahan langsung atau menggunakan alat bantu seperti GPS. Namun, metode ini memiliki keterbatasan, seperti waktu yang lama, ketergantungan pada tenaga kerja, serta risiko kesalahan manusia dalam pencatatan.

Selain itu, pencatatan posisi pohon secara manual sering kali tidak presisi, sehingga menghambat analisis spasial yang akurat. Hal ini berpotensi mempengaruhi perencanaan perkebunan, pengelolaan sumber daya, serta efisiensi produksi secara keseluruhan.

Oleh karena itu, diperlukan pendekatan yang lebih efisien, akurat, dan berbasis teknologi untuk mendeteksi dan menghitung jumlah pohon sawit dalam suatu area perkebunan. Dalam beberapa tahun terakhir, perkembangan teknologi kecerdasan buatan (*Artificial Intelligence*) dan pembelajaran mendalam (*Deep Learning*) telah memberikan terobosan baru dalam analisis citra. Dengan memanfaatkan model *deep learning* seperti algoritma *DeepForest* yang dirancang untuk mendeteksi tajuk pohon pada citra udara beresolusi tinggi [2]. Sehingga algoritma *Deepforest* dapat digunakan untuk deteksi pohon sawit dan perhitungan jumlahnya dapat dilakukan secara otomatis dengan tingkat akurasi yang tinggi berdasarkan citra satelit.

Dengan demikian penggunaan algoritma *DeepForest* dapat mengatasi keterbatasan metode manual melalui sistem deteksi dan perhitungan pohon sawit yang dilakukan secara otomatis berbasis citra udara. Sistem ini mampu menghasilkan informasi jumlah pohon dan titik koordinatnya secara lebih presisi sehingga hasil yang diperoleh dapat mendukung pengelolaan perkebunan kelapa sawit dengan baik.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, rumusan masalah yang akan dibahas dalam penelitian ini adalah:

1. Bagaimana membangun sistem kecerdasan buatan untuk mendeteksi kanopi pohon sawit secara otomatis dari citra udara, menghitung jumlah pohon, serta menentukan titik koordinat setiap pohon agar dapat diintegrasikan ke dalam sistem WebGIS?
2. Bagaimana penerapan algoritma *DeepForest* untuk mendeteksi pohon sawit pada area perkebunan?
3. Bagaimana akurasi model *DeepForest* dalam estimasi jumlah pohon sawit dibandingkan dengan metode manual atau konvensional?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

1. Menerapkan algoritma *DeepForest* untuk deteksi kanopi, perhitungan jumlah, dan penentuan titik koordinat pohon sawit berbasis citra udara.
2. Menganalisis algoritma *DeepForest* untuk mendeteksi pohon sawit pada lahan perkebunan.
3. Mengukur akurasi model *DeepForest* dalam estimasi jumlah pohon sawit.

1.4 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah:

1.4.1 Manfaat Bagi Mahasiswa

1. Penelitian ini memberikan wawasan dan pemahaman tentang penerapan algoritma *DeepForest*.
2. Menjadi referensi bagi mahasiswa lain yang tertarik untuk mendalami penggunaan teknologi pemetaan berbasis GIS (*Geographic Information Systems*).

1.4.2 Manfaat Bagi Masyarakat

1. Mendorong terciptanya solusi untuk meningkatkan produktivitas perkebunan kelapa sawit melalui pemanfaatan teknologi modern.
2. Menyediakan informasi akurat yang dapat mendukung keberlanjutan lingkungan dan pengelolaan lahan secara bertanggung jawab.

1.4.3 Manfaat Bagi Universitas

Sebagai bahan referensi di perpustakaan Universitas Lampung untuk topik-topik yang berkaitan dengan permasalahan dalam penulisan Tugas Akhir ini.

1.5 Batasan Masalah

Penelitian ini akan difokuskan pada:

1. Penelitian ini terfokus pada penggunaan algoritma *DeepForest* untuk mendeteksi pohon kelapa sawit pada lahan perkebunan.

2. Penelitian ini terfokus mengukur akurasi model *DeepForest* dalam hal estimasi jumlah pohon sawit yang terdeteksi.

1.6 Sistematika Penulisan

Laporan ini dibagi menjadi beberapa bab untuk memudahkan dalam penguraian, antara lain:

BAB I : PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan laporan.

BAB II : TINJAUAN PUSTAKA

Pada bab ini membahas mengenai dasar teori yang digunakan sebagai sumber dalam memahami permasalahan dalam melakukan penelitian mengenai penerapan *deepforest* untuk deteksi dan estimasi jumlah pohon sawit pada tata kelola kebun sawit.

BAB III : METODOLOGI PENELITIAN

Menjelaskan metode penelitian, sumber data, alat dan bahan yang digunakan, serta alur kerja penelitian.

BAB IV : HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil penelitian yang mencakup penerapan *deepforest* untuk deteksi dan estimasi jumlah pohon sawit pada tata kelola kebun sawit.

BAB V : PENUTUP

Bab ini menyajikan kesimpulan dari penelitian yang dilakukan serta saran yang diberikan berdasarkan hasil analisis.

II. TINJUAN PUSTAKA

2.1 Kelapa Sawit

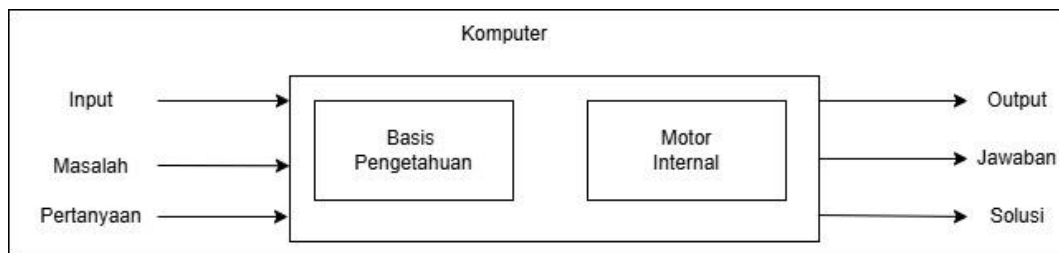
Kelapa sawit memiliki manfaat ekonomi dan strategis untuk pembangunan wilayah, terutama di negara-negara tropis seperti Indonesia dan Malaysia. Kelapa sawit sangat penting karena menghasilkan minyak industri, makan, dan biodiesel, bahan bakar nabati. Selain itu, sebagai penghasil utama minyak nabati, kelapa sawit lebih produktif daripada tanaman penghasil minyak nabati lainnya. Sebagian besar orang percaya bahwa kelapa sawit berasal dari Nigeria, di Afrika Barat.[3].

Eksokarp, mesokarp, endokarp, dan endosperma adalah empat bagian utama buah sawit. Eksokarp, atau bagian terluar buah sawit, adalah bagian yang sangat penting karena di dalamnya sebagian besar minyak (crude palm oil) tersimpan. Eksokarp terdiri dari kulit buah yang bertekstur licin dan berwarna merah jingga saat buah matang. Bagian ini adalah daging buah yang berserabut dan berwarna kuning cerah. Namun, endokarp, atau endocarp, adalah bagian lebih dalam setelah mesokarp, yang terdiri dari cangkang atau tempurung yang melindungi inti sawit atau kernel (endosperm)[4]. Selain struktur buah, kanopi pohon sawit memiliki ciri khas yang mudah dikenali dalam citra penginderaan jauh. Kanopinya berbentuk roset yaitu pola melingkar dengan daun menyirip yang memancar dari batang. Diameter kanopi pada pohon dewasa umumnya 5–8 meter, dengan daun sepanjang 4–7 meter.

2.2 *Artificial Intelligence*

Artificial Intelligence (AI) sebagai cabang ilmu komputer yang berfokus pada pengembangan sistem yang mampu menjalankan tugas-tugas yang biasanya membutuhkan kecerdasan manusia, seperti berpikir logis, memahami bahasa, belajar, dan mengambil keputusan. Dengan menjelaskan bahwa *Artificial Intelligence* (AI) dapat dikategorikan ke dalam dua pendekatan utama, yaitu AI

kuat (*strong AI*) yang bertujuan menciptakan mesin dengan kecerdasan setara manusia, dan AI lemah (*weak AI*) yang hanya meniru perilaku cerdas tanpa kesadaran. Selain itu terdapat empat sudut pandang dalam kecerdasan buatan, yaitu sistem yang meniru cara berpikir manusia, sistem yang meniru perilaku manusia, sistem yang berpikir secara rasional, serta sistem yang bertindak secara rasional [5]. Dalam penerapan kecerdasan buatan terdapat basis pengetahuan (*knowledge base*) dan motor inferensi (*inference engine*) dan jika dapat digambarkan seperti gambar 2.1 yaitu:



Gambar 2.1 Penerapan Konsep Kecerdasan Buatan

Selain kecerdasan buatan terdapat pula kecerdasan alami. Di mana kecerdasan buatan memiliki keuntungan dibandingkan kecerdasan alami antara lain:

- Kecerdasan buatan tidak akan berubah dan mengubah sistem komputer atau lebih bersifat permanen. Kecerdasan alami akan mudah mengalami perubahan.
- Kecerdasan buatan dapat dengan mudah direplikasi dan didistribusikan ke dalam sistem komputer, sedangkan kecerdasan alami memerlukan waktu dan proses pembelajaran manusia untuk mentransfer pengetahuan dari satu individu ke individu lainnya.
- Kecerdasan buatan dapat didokumentasikan secara sistematis, sedangkan kecerdasan alami sulit untuk direplikasi atau diduplikasi secara utuh.
- Kecerdasan buatan dinilai lebih efisien dari segi biaya dibandingkan kecerdasan alami, karena kecerdasan alami memerlukan keterlibatan tenaga ahli serta waktu yang relatif lama untuk menghasilkan pengetahuan atau keahlian tertentu. [6].

2.3 Deep Learning

Deep Learning merupakan pendekatan dalam pembelajaran mesin (machine learning) yang meniru cara kerja otak manusia dalam memproses data dan menciptakan pola untuk digunakan dalam pengambilan keputusan. *Deep learning*

merupakan bagian dari *machine learning* yang secara khusus menggunakan arsitektur jaringan saraf tiruan berlapis-lapis (*multi-layer neural networks*) yang disebut sebagai *deep neural networks*. Selanjutnya, deep learning dinilai sangat efektif dalam mengolah data berukuran besar dan bersifat kompleks, seperti citra, suara, dan teks, karena kemampuannya dalam melakukan ekstraksi fitur secara otomatis dari data mentah melalui serangkaian lapisan representasi yang tersusun secara berurutan [7].

2.4 Artificial Neural Network

Artificial Neural Network (ANN) atau jaringan syaraf tiruan adalah model komputasi yang terinspirasi oleh cara kerja otak manusia yang terdiri dari elemen-elemen pemrosesan berupa neuron yang saling terhubung dan berinteraksi dalam menjalankan proses pengolahan informasi. ANN digunakan untuk mengenali pola, memprediksi hasil, dan menyelesaikan berbagai masalah yang melibatkan data besar. Setiap neuron dalam ANN menerima *input* memprosesnya dengan fungsi matematis, dan menghasilkan *output* yang dikirim ke neuron lainnya. Proses ini terjadi dalam beberapa lapisan (*layer*): lapisan *input*, lapisan tersembunyi (*hidden layer*), dan lapisan *output*. Proses pembelajaran ANN melibatkan penyesuaian berat (*weight*) yang menghubungkan neuron-neuron untuk meminimalkan kesalahan dalam hasil yang diprediksi, menggunakan algoritma seperti *backpropagation* [8]. Pada arsitektur *Artificial Neural Network* (ANN) atau jaringan syaraf tiruan terdapat jumlah lapisan (*layer*) dan jumlah *node* pada setiap lapisan pada jaringan syaraf tiruan yang terbagi menjadi 3 bagian [9].

a. Lapisan *input*

Lapisan input terdiri dari sejumlah neuron yang berfungsi menerima data masukan dari lingkungan eksternal. Data yang diberikan pada lapisan ini merepresentasikan permasalahan yang akan diproses oleh jaringan syaraf tiruan.

b. Lapisan tersembunyi

Lapisan tersembunyi tersusun atas neuron-neuron yang berperan dalam mengolah informasi dari lapisan input. Hasil pemrosesan pada lapisan ini tidak dapat diamati secara langsung karena berfungsi sebagai proses internal jaringan.

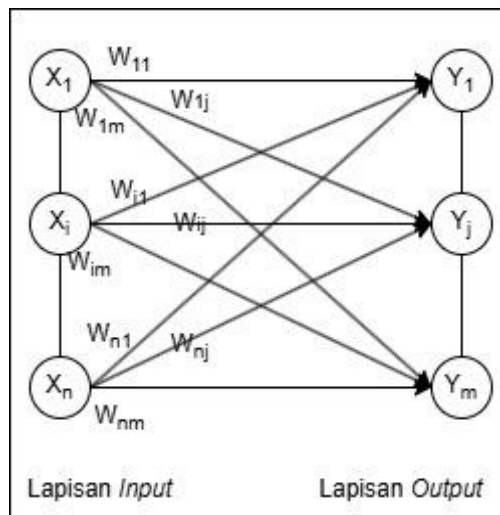
c. Lapisan *output*

Lapisan output tersusun atas neuron-neuron keluaran yang berfungsi menghasilkan nilai akhir dari proses jaringan syaraf tiruan. Keluaran yang dihasilkan pada lapisan ini merupakan respons jaringan terhadap permasalahan yang diberikan sebagai masukan.

Struktur jaringan syaraf tiruan juga dapat diklasifikasikan berdasarkan bentuk arsitekturnya. Secara umum, jaringan syaraf tiruan dibedakan menjadi tiga jenis arsitektur, yaitu sebagai berikut:

a. Jaringan lapis tunggal

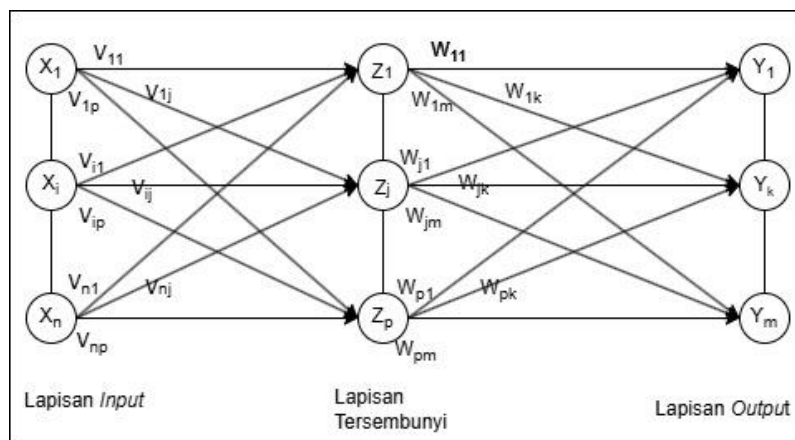
Jaringan lapisan tunggal merupakan jaringan yang hanya memiliki satu lapisan bobot penghubung antara neuron input dan neuron output. Arsitektur ini terdiri dari unit-unit input yang menerima sinyal dari lingkungan eksternal serta unit-unit output yang menghasilkan respons jaringan. Keluaran jaringan dapat langsung diamati pada lapisan output tanpa melalui lapisan tersembunyi. Pada arsitektur jaringan syaraf tiruan jenis jaringan lapisan tunggal dapat digambarkan sebagai berikut.



Gambar 2.2 Jaringan Lapis Tunggal

b. Jaringan multilapis

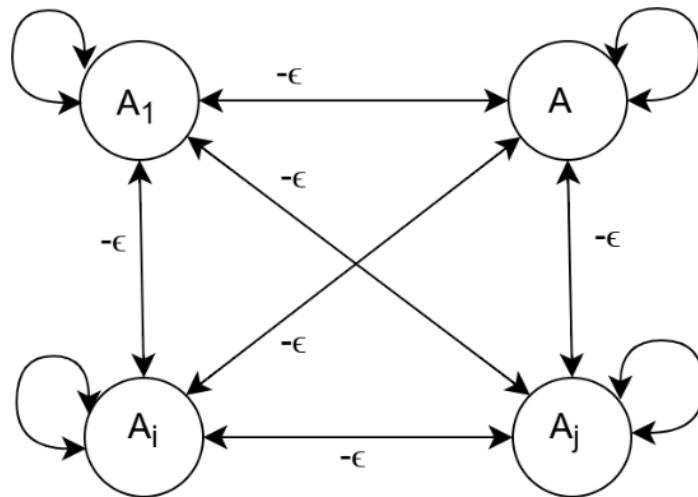
Jaringan multilapis adalah jaringan yang memiliki satu atau lebih lapisan tersembunyi di antara lapisan input dan lapisan output. Keberadaan lapisan tersembunyi memungkinkan jaringan multilapis memiliki kemampuan yang lebih baik dalam mempelajari pola dan menyelesaikan permasalahan yang kompleks dibandingkan dengan jaringan lapisan tunggal, meskipun proses pelatihannya cenderung lebih rumit. Pada arsitektur jaringan syaraf tiruan jenis jaringan lapisan *multilapis* dapat digambarkan sebagai berikut.



Gambar 2.3 Jaringan Multilapis

c. Jaringan kompetitif

Jaringan kompetitif merupakan jenis jaringan syaraf tiruan di mana neuron-neuron di dalamnya saling bersaing untuk menjadi neuron yang aktif. Neuron yang memenangkan kompetisi akan memberikan respons, sementara neuron lainnya tetap tidak aktif. Jika dapat digambarkan jaringan kompetitif.



Gambar 2.4 Jaringan Kompetitif

Kemudian terdapat cara kerja jaringan syaraf tiruan dengan model matematis dari sebuah neuron yaitu:

$$yNet = \sum_{i=1}^n x_i w_i + b$$

$$y = f(yNet)$$

Hasil penjumlahan selanjutnya dievaluasi dengan membandingkannya terhadap suatu nilai ambang menggunakan fungsi aktivasi (threshold). Nilai keluaran yang diperoleh kemudian disalurkan melalui bobot koneksi menuju neuron-neuron lain yang saling terhubung [10]. Adapun penjelasan singkat dari notasi pada persamaan tersebut adalah sebagai berikut:

- x_i : menyatakan nilai masukan ke-i
- w_i : merupakan bobot dari masukan ke-i,
- b : bias
- f : menunjukkan fungsi aktivasi.

2.5 Convolution Neural Network

Convolution Neural Network merupakan sebuah arsitektur dalam *deep learning* yang dirancang khusus untuk mengenali pola visual secara otomatis dan efisien dari data gambar. Convolutional Neural Network (CNN) dikembangkan dengan meniru

mekanisme sistem penglihatan manusia, khususnya cara neuron pada korteks visual dalam mengenali karakteristik visual seperti tepi, sudut, dan pola bentuk. Oleh karena itu, CNN memiliki kinerja yang sangat baik dalam berbagai tugas pengolahan citra, termasuk klasifikasi citra, deteksi objek, segmentasi citra, serta pengenalan wajah [11].

Komponen Utama Arsitektur *Convolution Neural Network* sebagai berikut.

a. Lapisan Konvolusi

Lapisan konvolusi pada komponen utama dalam *Convolution Neural Network*. Maka setiap lapisan konvolusi terdiri dari sejumlah *filter (kernel)* yang bergerak dalam seluruh *input*-an citra. Operasi konvolusi yang dilakukan dengan menggeser *filter* di atas citra dengan mengalikan nilai piksel dari citra yang memiliki bobot *filter* yang tepat dengan penjumlahan hasilnya. Maka *output* yang dihasilkan dari operasi konvolusi dengan peta fitur yang mengetahui keberadaan fitur-fitur khusus dalam citra seperti tepi, tekstur dan bentuk. Dalam lapisan konvolusi terdapat konsep yang harus dipahami secara detail sebagai berikut.

- Operasi Konvolusi

Operasi konvolusi merupakan operasi matematika yang menggeser *filter* di atas citra dan mengalikan nilai piksel dari citra dengan bobot *filter* yang sesuai dan diikuti dengan penjumlahan hasil. Maka hasil dari operasi konvolusi adalah sebuah nilai tunggal yang diperlihatkan tingkat kesamaan antara bagian dari citra yang berada di bawah *filter* dengan pola yang didefinisikan oleh *filter* itu sendiri.

- *Filter (kernel)*

Filter (kernel) adalah matriks kecil yang digunakan untuk melakukan operasi konvolusi. Dengan adanya *filter* ini berperan penting dalam mengekstraksi fitur-fitur khusus dari citra, seperti tepi, tekstur, atau bentuk. Dalam proses pelatihan *Convolution Neural Network* akan menyesuaikan bobot *filter* untuk menemukan fitur-fitur secara otomatis selama proses berjalan.

- Peta Fitur (*Feature Map*)

Pada peta fitur merupakan hasil dari operasi konvolusi pada setiap *filter* menyoroti keberadaan fitur-fitur khusus dalam citra. Kemudian setiap *filter* akan menghasilkan peta fitur yang berbeda dengan menyoroti fitur-fitur yang berbeda.

- *Stride* dan *Padding*

Operasi konvolusi ini melibatkan seperti *stride* dan *padding*. Pada *stride* merupakan jumlah piksel yang digeserkan guna *filter* pada setiap langkahnya selama operasi konvolusi. *Stride* 1 merupakan *filter* bergerak dalam 1 piksel pada setiap langkahnya. Sedangkan *Padding* merupakan proses untuk menambahkan piksel nol di sekeliling citra input sebelum menjalankan operasi konvolusi. Maka berguna untuk mempertahankan ukuran citra setelah operasi konvolusi dengan menggunakan *filter* dengan ukuran yang besar.

b. Operasi *Pooling*

Operasi *pooling* dilakukan dengan setiap peta fitur yang dihasilkan oleh lapisan konvolusi. Dengan proses ini wilayah tetap dari peta fitur diperiksa dan satu nilai statistik diambil. Kemudian terdapat reduksi dimensi yang membantu mengurangi dimensi dari peta fitur dengan mengurangi jumlah parameter dalam jaringan dan mempercepat komputasi. Lalu terdapat invariasi terhadap pergeseran ini guna membuat model invarian terhadap pergeseran spasial kecil dalam citra. Selanjutnya terdapat operasi *pooling* ini terdapat jenis operasi umum yang digunakan sebagai berikut.

- *Max Pooling*

Pada *max pooling* ini memiliki nilai maksimum dalam wilayah tetap dari peta maka fitur diambil sebagai representasi fitur yang paling menonjol dalam wilayah yang sudah ditetapkan sebelumnya.

- *Average Pooling*

Dalam *average pooling* memiliki rata-rata dari semua nilai dalam wilayah tetap dari peta fitur yang sedang diambil guna representasi fitur.

c. *Full Connected Layer*

Full connected layer merupakan komponen arsitektur terakhir dari *Convolution Neural Network*. Dalam *full connected layer* terdapat neuron yang terhubung ke semua neuron di lapisan sebelumnya maka setiap neuron menerima inputan dari setiap neuron dalam lapisan sebelumnya. *Full connected layer* pada *Convolution Neural Network* bertindak sebagai pengklasifikasi akhir yang menggunakan fitur-fitur yang diekstraksi dari lapisan konvolusi dan *pooling* untuk melakukan prediksi dan klasifikasi [12]

2.6 Citra

Citra adalah suatu gambaran atau kemiripan dari suatu objek dengan serangkaian teknik untuk meningkatkan kualitas visual, menghilangkan derau (*noise*) mengekstraksi informasi, atau mengompresi data gambar. Citra secara umum dapat dikelompokkan menjadi dua jenis, yaitu citra analog dan citra digital. Penjelasan masing-masing jenis citra adalah sebagai berikut [13]:

a. Citra Analog

Citra analog merupakan citra yang bersifat kontinu sehingga tidak dapat direpresentasikan secara langsung dalam sistem komputer. Oleh sebab itu, agar citra analog dapat diolah menggunakan komputer, diperlukan proses konversi terlebih dahulu ke dalam bentuk citra digital.

b. Citra Digital

Citra digital adalah citra yang dapat diolah atau diproses oleh *computer*. Pada citra digital representasi citranya diolah dengan mesin yang terdapat sistem sampling dan kuantisasi. Kemudian pada sistem sampling itu adalah sistem yang mengubah citra analog menjadi citra digital dengan cara citra analog dibagi menjadi baris dan kolom sehingga menjadi citra diskrit. Maka hasil dari pertemuan antara baris dan kolom dapat disebut dengan piksel. Selanjutnya, kuantisasi merupakan proses yang berfungsi mengubah nilai intensitas analog menjadi nilai intensitas diskrit, sehingga memungkinkan pembentukan variasi tingkat keabuan atau gradasi warna sesuai

dengan kebutuhan. Adapun jenis-jenis citra yang terdapat pada citra digital adalah sebagai berikut:

- Citra Biner

Citra biner merupakan bentuk representasi citra yang hanya menggunakan dua tingkat warna, yaitu hitam dan putih. Pada citra biner, setiap piksel direpresentasikan menggunakan satu bit memori sehingga hanya memiliki dua kemungkinan nilai intensitas, yaitu 0 dan 1. Nilai 0 merepresentasikan warna hitam, sedangkan nilai 1 merepresentasikan warna putih.

- Citra *Grayscale*

Citra grayscale direpresentasikan dalam bentuk matriks data yang berisi nilai intensitas untuk setiap piksel dengan rentang nilai antara 0 hingga 255. Setiap piksel pada citra grayscale disimpan menggunakan 8 bit memori.

- Citra RGB

Citra RGB adalah citra berwarna di mana setiap piksel tersusun atas tiga komponen warna utama, yaitu merah (red), hijau (green), dan biru (blue). Nilai warna pada setiap piksel ditentukan oleh kombinasi intensitas dari ketiga komponen warna tersebut yang disimpan pada lokasi piksel tertentu. Citra warna umumnya disimpan dalam format 24 bit, yang berasal dari penggunaan masing-masing 8 bit untuk komponen warna merah, hijau, dan biru. Dengan konfigurasi tersebut, citra RGB mampu merepresentasikan hingga sekitar 16 juta variasi warna yang berbeda. [14].

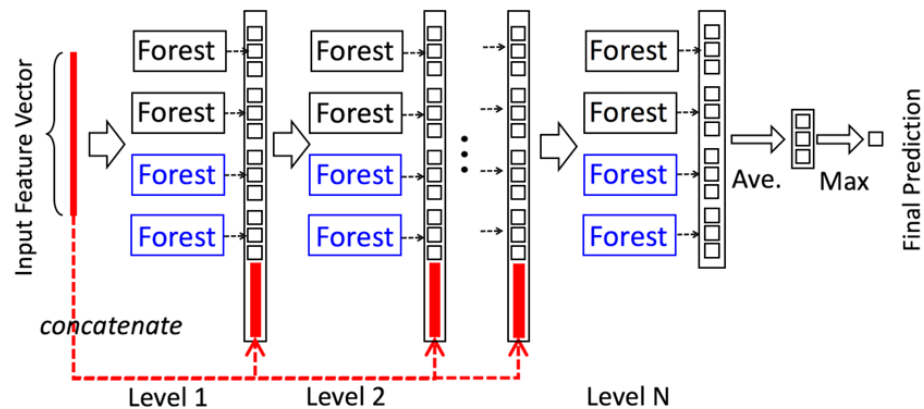
Pemahaman mengenai jenis-jenis citra dan karakteristiknya menjadi dasar penting dalam pengembangan teknologi pengolahan citra digital untuk berbagai keperluan analisis objek visual. Hal ini juga berkaitan dengan konsep data di dalam citra yang terdiri atas data spasial dan data atribut (non-spasial). Data spasial merupakan data yang menunjukkan lokasi atau posisi suatu objek di permukaan bumi yang disajikan dalam bentuk vektor atau raster di mana citra digital termasuk ke dalam kategori data spasial berbasis raster. Sementara itu data atribut (non-spasial) adalah informasi tambahan yang menjelaskan karakteristik suatu objek, seperti jumlah, kondisi, atau jenis objek yang dianalisis pada citra.

2.7 *DeepForest*

DeepForest merupakan metode pembelajaran mendalam yang dikembangkan sebagai alternatif deep neural network. Metode ini memanfaatkan layer-layer bertingkat yang tersusun dari ensambel pohon keputusan (*decision trees*) seperti random forest dan *completely random tree forest*. Melalui struktur berlapis (cascade) setiap lapisan berfungsi untuk membangun dan memperkaya representasi data secara bertahap sehingga informasi yang dihasilkan menjadi semakin kompleks dan informatif pada setiap tingkat pemodelan. *DeepForest* dirancang khusus untuk aplikasi ekologi terutama dalam mendeteksi tajuk pohon pada citra udara beresolusi tinggi. Model ini mampu memprediksi bounding box yang merepresentasikan tajuk pohon individual pada citra RGB. Selain itu, *DeepForest* mampu bekerja secara efektif pada dataset berskala kecil hingga menengah serta tidak memerlukan sumber daya komputasi yang tinggi seperti penggunaan GPU berkapasitas besar sehingga lebih efisien dan mudah diimplementasikan. Model *DeepForest* juga telah menyediakan model pra-latih (pretrained model) yang memungkinkan proses adaptasi dan pelatihan ulang dilakukan dengan cepat pada berbagai kondisi lingkungan dan resolusi citra yang berbeda.

Terdapat beberapa algoritma lain yang juga dapat digunakan untuk deteksi objek pada citra udara antara lain YOLO, Faster R-CNN, dan SSD. Algoritma-algoritma tersebut umumnya berbasis *deep neural network* dan mampu menghasilkan akurasi tinggi, namun memerlukan dataset yang besar serta sumber daya komputasi yang lebih tinggi, seperti GPU, sehingga kurang optimal untuk kondisi penelitian ini. Dibandingkan dengan algoritma tersebut, *DeepForest* lebih sesuai karena dirancang khusus untuk deteksi tajuk pohon, lebih ringan secara komputasi, serta lebih mudah diadaptasikan pada data kebun sawit dengan skala penelitian yang terbatas [2].

Kemudian terdapat arsitektur *DeepForest* yang dapat dilihat pada gambar 2.5



Gambar 2.5 Arsitektur *DeepForest*

Berdasarkan gambar 2.5 di atas merupakan arsitektur *deepforest* yang dimulai dengan tahapan pertama yaitu dengan memasukkan vektor fitur (*input feature vector*), yaitu sekumpulan nilai numerik yang mewakili karakteristik dari setiap sampel data. Dalam konteks penelitian ini, vektor fitur dapat berupa informasi dari citra kanopi pohon sawit.

Pada tahapan kedua yaitu level 1 atau level pertama yaitu vektor fitur tersebut dimasukkan ke dalam beberapa model *forest*. Setiap *forest* akan melakukan proses pelatihan dan menghasilkan probabilitas kelas atau nilai prediksi yang merepresentasikan tingkat keyakinan terhadap suatu kelas target. Hasil dari semua *forest* pada level ini kemudian digabungkan (*concatenate*) dengan informasi dasar yang diambil langsung dari citra sebelum masuk ke model.

Kemudian pada tahapan ketiga yaitu masuk pada level 2 dengan hasil gabungan dari level pertama akan menjadi masukan baru bagi model. *forest* yang ada kembali melakukan proses pembelajaran tetapi dengan informasi yang lebih lengkap karena sudah memanfaatkan fitur awal dari citra dan hasil prediksi sebelumnya. Dengan level kedua yang memperkuat pemahaman model terhadap ciri khas pohon yang mungkin belum sepenuhnya terbaca di level pertama.

Selanjutnya masuk pada tahapan di level N ini berfungsi sebagai tahap akhir yang menyempurnakan hasil prediksi. Model pada lapisan ini telah menggabungkan semua informasi dari lapisan-lapisan sebelumnya. Sehingga mampu menghasilkan prediksi yang lebih stabil dan konsisten. Lapisan terakhir ini juga menentukan hasil akhir yang akan digunakan pada proses selanjutnya, misalnya untuk klasifikasi atau deteksi objek. *Output* dari level N inilah yang kemudian diteruskan ke tahap agregasi hasil.

Pada tahapan agregasi hasil ini setelah melewati seluruh level setiap *forest* pada level terakhir akan menghasilkan keluaran berupa probabilitas kelas. Hasil-hasil tersebut kemudian digabungkan dan dilakukan proses agregasi dengan dua metode utama yaitu *Averaging* (Rata-rata) yaitu menghitung rata-rata dari seluruh hasil prediksi untuk memperoleh nilai konsensus. *Max Pooling* (Nilai Maksimum) yaitu memilih nilai prediksi tertinggi sebagai hasil akhir. Kedua metode ini digunakan untuk meningkatkan stabilitas dan akurasi hasil prediksi.

Pada Tahapan terakhir yaitu *Final Prediction model* sudah menggabungkan semua hasil prediksi dari setiap level (mulai dari level pertama hingga level terakhir) yang telah melalui proses agregasi. Informasi tersebut kemudian digunakan untuk menghasilkan keputusan akhir dengan tingkat keyakinan yang paling tinggi berdasarkan pola yang sudah dipelajari sepanjang proses pelatihan. Prediksi akhir ini bisa berbentuk label kelas, seperti membedakan antara beberapa kategori objek, atau nilai numerik [15].

2.8 Confusion Matrix

Confusion Matrix merupakan metode yang diperoleh nilai *precision*, *recall*, *F1-Score*, dan akurasi. Pada nilai *Confusion Matrix* ini dapat ditunjukkan dalam satuan persen. Kemudian *confusion matrix* digunakan untuk mengevaluasi klasifikasi segmen seperti pada tabel berikut yaitu:

Tabel 2.1 Mengevaluasi klasifikasi segmen

		Kelas Prediksi	
		<i>Positive</i>	<i>Negative</i>
Kelas Aktual	<i>Positive</i>	<i>True Positive (TP)</i>	<i>False Negative (FN)</i>
	<i>Negative</i>	<i>False Postitive (Fp)</i>	<i>True Negative</i>

Keterangan :

True Positive (TP) : merupakan data dari kelas positif yang berhasil diprediksi secara tepat sebagai kelas positif.

False Positive (FP): data dari kelas negatif yang salah diprediksi sebagai kelas positif.

True Negative (TN): data dari kelas negatif yang diprediksi dengan benar sebagai kelas negatif.

False Negative (FN): data dari kelas positif yang salah diklasifikasikan sebagai kelas negatif.

Adapun penjelasan mengenai *precision*, *recall* , dan *F1-Score* [16]

- *Precision*

Precision adalah proporsi prediksi nilai positif yang ketepatan nya dibuat oleh model.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

- *Recall*

Recall adalah mengukur seberapa baik model mendeteksi semua kasus positif yang sebenarnya ada.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

- *F1-Score*

F1-score adalah mengukur keseimbangan antara *precision* dan *recall*.

$$F1\ score = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

2.9 PyTorch

PyTorch adalah sebuah pustaka yang digunakan untuk memanipulasi *tensor* sehingga menjadi fondasi utama dalam *deep learning*. *Tensor* adalah objek matematika untuk menyimpan data multidimensi dan memungkinkan *pyTorch* fleksibel dalam proses pemodelan dan eksperimen. *Pytorch* digunakan untuk membangun graf komputasi sesuai secara otomatis. Kemudian didalam *pyTorch* terdapat fitur autograf *pyTorch* sehingga memungkinkan perhitungan gradien secara otomatis. Selain itu *PyTorch* juga menyediakan modul *torch* yang mempermudah pembuatan dan pelatihan jaringan saraf dengan abstraksi *layer-layer neural network*[16] .

2.10 Google Colaboratory

Google Colaboratory (Colab) merupakan layanan berbasis web yang memungkinkan pengguna untuk menulis, mengeksekusi, dan menguji kode Python secara langsung melalui peramban tanpa memerlukan proses instalasi atau konfigurasi tambahan. Platform ini menyediakan akses gratis ke sumber daya komputasi, seperti GPU dan TPU, serta mendukung kolaborasi melalui fitur berbagi yang terintegrasi dengan Google Drive. Platform ini sangat cocok digunakan oleh pelajar, ilmuwan data, hingga peneliti AI karena mendukung lingkungan interaktif yang memungkinkan penggabungan antara kode, teks, gambar, LaTeX, hingga HTML dalam satu dokumen yang disebut *Colab notebook*. *Colab* merupakan turunan dari *Jupyter Notebook*, namun dijalankan di *cloud server* milik Google, sehingga pengguna dapat menjalankan pemrosesan data berskala besar tanpa bergantung pada spesifikasi lokal perangkat mereka. Pengguna juga dapat mengimpor data dari berbagai sumber, seperti Google Drive, GitHub, maupun spreadsheet. Selain itu, platform ini mendukung penggunaan berbagai pustaka populer, seperti NumPy dan Matplotlib, untuk keperluan pengolahan data

dan visualisasi. *Colab* menjadi alat yang sangat mendukung analisis dan visualisasi data. Lebih lanjut, penggunaan *runtime* berbasis GPU memungkinkan akselerasi proses komputasi secara signifikan, menjadikan *Colab* sebagai solusi praktis dan efisien dalam pengembangan proyek berbasis *Python*, khususnya di bidang *data science* dan kecerdasan buatan [17].

2.11 Quantum Geographic Information System (QGIS)

Quantum Geographic Information System (QGIS) merupakan perangkat lunak Sistem Informasi Geografis (SIG) berbasis open source yang bersifat *user-friendly* dan dirilis di bawah lisensi GNU *General Public License*. QGIS merupakan salah satu proyek resmi yang dikembangkan oleh *Open Source Geospatial Foundation (OSGeo)*. Perangkat lunak ini dapat dijalankan pada berbagai sistem operasi, seperti *Linux*, *Unix*, *Mac OS X*, *Windows*, serta *Android*, dan mendukung pengolahan data vektor, raster, maupun basis data dalam berbagai format. QGIS dapat digunakan untuk melakukan pengolahan data geospasial. Salah satu keunggulan QGIS adalah dukungannya terhadap *plugin* eksternal, yang memungkinkan pengguna menambahkan berbagai fungsi seperti *geocoding*, analisis spasial lanjutan, hingga integrasi dengan peta daring seperti *Leaflet* dan *OpenLayers*. QGIS juga memiliki integrasi mendalam dengan bahasa pemrograman *Python* melalui *PyQGIS*, yang sangat berguna bagi pengguna yang ingin mengotomatiskan proses atau membuat *plugin kustom* [18].

2.12 Roboflow

Roboflow merupakan platform berbasis web yang dikembangkan untuk memfasilitasi pengelolaan dataset serta mendukung proses pengembangan proyek dalam bidang *computer vision*. Platform ini memudahkan pengguna dalam melakukan pengumpulan data secara efisien, *prapemrosesan* dataset, hingga pelatihan model. Dengan *Roboflow*, pengguna dapat melakukan anotasi atau penandaan objek yang akan dideteksi menggunakan *bounding box* serta berbagi dataset secara lebih efisien. Selain itu, platform ini juga menyediakan berbagai fitur *prapemrosesan* seperti konversi gambar ke mode *grayscale*, *resize*, *cropping*,

normalisasi, dan *augmentasi data* untuk meningkatkan variasi dan kualitas pelatihan model seperti rotasi, *flipping*, *blur*, hingga perubahan pencahayaan [19].

2.13 Penelitian Terkait

2.13.1 *Tree Counting with Deep Forest Algorithm for Kulon Progo District in Yogyakarta, Indonesia Using Pleiades Satellite Imagery.*

(Penghitungan Jumlah Pohon dengan Algoritma *DeepForest* untuk Wilayah Kabupaten Kulon Progo di Yogyakarta, Indonesia Menggunakan Citra Satelit Pleiades)

Penelitian ini dilakukan oleh David Wijaya, Ibrahim Seto Aditama, Stanley Armando Austen, Calvin Surya Widjaja, Bakti Amirul Jabar, dan Edy Irwansyah dari Bina Nusantara University dan diterbitkan pada tahun 2024. Tujuan dari penelitian ini adalah untuk menghitung jumlah pohon di wilayah Kabupaten Kulon Progo, Yogyakarta, menggunakan citra satelit Pleiades dengan memanfaatkan algoritma *Deep Forest*. Penelitian ini dilatarbelakangi oleh pentingnya pelestarian pohon dalam mengurangi polusi udara dan dampak perubahan iklim sehingga diperlukan metode otomatisasi berbasis citra satelit untuk pemetaan pohon secara efisien.

Penelitian ini menggunakan algoritma *Deep Forest* untuk mendeteksi kanopi pohon dari citra *RGB* udara yang berbasis pada data citra satelit Pleiades yang memiliki resolusi sangat tinggi. Data citra diproses menggunakan aplikasi *ArcGIS Pro* untuk melakukan *labeling* objek pohon dan membangun dataset pembelajaran mendalam (*deep learning dataset*). Dataset ini kemudian digunakan dalam proses pelatihan model deteksi di lingkungan *Jupyter Notebook*. Sebelum pelatihan, dilakukan preprocessing tambahan seperti rotasi gambar setiap 30 derajat dan resizing citra ke ukuran 256×256 piksel untuk memperkaya variasi data. Evaluasi model dilakukan dengan menghitung nilai *Precision*, *Recall*, dan *Validation F1-Score* guna mengukur kinerja deteksi pohon secara akurat.

Adapun untuk hasil penelitian pada eksperimen pertama diperoleh nilai F1-Score sebesar 0.760, kemudian meningkat menjadi 0.774 pada eksperimen kedua, dan akhirnya mencapai 0.779 pada eksperimen ketiga. Peningkatan akurasi ini diperoleh melalui perbaikan data *labeling* dan penyesuaian parameter *training* seperti *learning rate*. Hasil ini menunjukkan bahwa algoritma *DeepForest* cukup efektif dalam mendeteksi pohon dari citra satelit dengan tingkat akurasi yang baik[20].

Tabel 2.2 Hasil Penelitian Terkait Perhitungan Akurasi DeepForest

	Experiment 1	Experiment 2	Experiment 3
F1 Score	0.760	0.774	0.779
Recall	0.743	0.792	0.789
Precision	0.778	0.756	0,769

2.13.2 *Urban Tree Detection from Remote Sensing Data based on DeepForest Model*

(Deteksi Pohon di Area Perkotaan dari Data Penginderaan Jauh Menggunakan Model *DeepForest*)

Penelitian dilakukan oleh Vasil Dakova dari Delft University of Technology, Belanda, dan Dessislava Petrova-Antonova dari GATE Institute, Sofia University "St. Kliment Ohridski", Bulgaria. Penelitian ini diterbitkan pada tahun 2024. Tujuan utama dari penelitian ini adalah untuk membuktikan bahwa model deteksi pohon yang awalnya dikembangkan untuk lingkungan hutan dapat diadaptasi untuk mendeteksi pohon dalam lingkungan perkotaan yang lebih kompleks. Studi kasus dilakukan di Distrik Lozenets, Sofia, Bulgaria, yang dikenal sebagai salah satu area ter hijau di kota tersebut. Penelitian ini bertujuan mendukung inventarisasi pohon secara otomatis, yang sangat berguna untuk perencanaan dan pengelolaan kota berkelanjutan.

Penelitian ini menggunakan algoritma *DeepForest*, yaitu model *deep learning* berbasis arsitektur RetinaNet yang mengandalkan fungsi *loss (focal loss)* untuk mendeteksi pohon. *DeepForest* awalnya dilatih menggunakan data *LiDAR* dan kemudian disesuaikan dengan data manual untuk meningkatkan akurasi. Dalam penelitian ini, *DeepForest* diadaptasi untuk mendeteksi pohon di lingkungan perkotaan melalui tiga pendekatan, yaitu *Single Tree Detection Model* untuk mendeteksi pohon individu, *Tree Cluster Detection Model* untuk mendeteksi kelompok pohon yang kanopinya saling tumpang tindih, dan *Mixed Model* yang menggabungkan dua model sebelumnya menggunakan teknik *Poisson Disk Sampling*. Data yang digunakan berasal dari citra ortofoto beresolusi tinggi (0,1 meter) tahun 2020 yang dipotong menjadi *tile* berukuran $51,2 \text{ m} \times 51,2 \text{ m}$, dengan pelabelan berdasarkan data EdnoDarvo dan anotasi manual. Dataset dibagi menjadi data pelatihan, validasi, dan pengujian (80:20) dan kinerja model dievaluasi menggunakan metrik *precision*, *recall*, dan *F1-score* dengan syarat deteksi dianggap benar jika *IoU* lebih dari 50%.

Hasil penelitian menunjukkan bahwa penggunaan model *DeepForest* yang dilatih ulang untuk deteksi pohon individu menghasilkan performa yang cukup tinggi, dengan *F1-score* sebesar 0,68, *precision* 0,794, dan *recall* 0,594 pada data uji setelah satu *epoch* pelatihan. Untuk model deteksi kelompok pohon (*cluster detection*), performa lebih rendah dengan *F1-score* sebesar 0,398. Sementara itu, *Mixed Model* yang menggabungkan hasil deteksi individu dan kelompok dengan teknik *Poisson Disk Sampling* mampu mencapai *F1-score* sebesar 0,674, menunjukkan performa yang kompetitif dan stabil. Visualisasi hasil menunjukkan bahwa *DeepForest* sangat efektif dalam mendeteksi pohon individu yang tersebar, sedangkan deteksi kelompok lebih berguna di area dengan kanopi pohon yang tumpang tindih. Penggabungan dua model ini mampu mengatasi masalah deteksi yang terlewat di daerah padat pohon. Secara keseluruhan, adaptasi *DeepForest* untuk lingkungan perkotaan terbukti berhasil dan akurat, dengan performa yang sebanding dengan penerapannya di lingkungan hutan, meskipun masih menghadapi tantangan seperti bayangan bangunan dan tumpang tindih kanopi pohon[21].

Tabel 2.3 Hasil Penelitian Terkait Evaluasi Detektor Mendeteksi Pohon

Set	Precision	Recall	F1-Score	Avg. Err	Cum. Err
Training	0.762	0.623	0.685	-3.154 trees	-104/497 trees
Test	0.794	0.594	0.680	-3.909 trees	-43/185 trees
Full	0.821	0.485	0.610	-7.454 trees	-82/185 trees

Table 1: Single Tree Detector-1 Epoch

Set	Precision	Recall	F1-Score	Avg. Err	Cum. Err
Training	0.661	0.524	0.585	-0.714 clusters	-20/64 clusters
Test	0.700	0.278	0.398	-1.555 clusters	-14/21 clusters
Full	0.5208	0.426	0.468	-0.888 clusters	-8/21 clusters

Table 2: Cluster Detector-5 Epochs

Set	Precision	Recall	F1-Score	Avg. Err	Cum. Err
Training	0.660	0.629	0.644	0.393 trees	13/497 trees
Test	0.779	0.594	0.674	-3.636 trees	-40/185 trees
Full	0.772	0.4931	0.601	-5.909 trees	-65/185 trees

Table 3: Mixed Tree Detector-1 Epoch

Set	Precision	Recall	F1-Score	Avg. Err	Cum. Err
Training	0.631	0.579	0.604	5.78 trees	191/497 trees
Test	0.719	0.599	0.654	0 trees	0/185 trees
Full	0.767	0.487	0.596	-5.909 trees	-67/185 trees

2.13.3 Tree Detection and Species Classification in a Mixed Species Forest Using Unoccupied Aircraft System (UAS) RGB and Multispectral Imagery

(Deteksi Pohon dan Klasifikasi Jenis Spesies di Hutan Campuran Menggunakan Citra RGB dan Multispektral dari Sistem Pesawat Tanpa Awak (UAS))

Penelitian ini dilakukan oleh Poornima Sivanandam dan Arko Lucieer dari University of Tasmania, Australia, dan dipublikasikan pada tahun 2022. Penelitian ini bertujuan untuk menemukan cara yang efektif dalam mendeteksi pohon satu per satu dan mengelompokkan jenis-jenis pohon di hutan bercampur di Australia. Untuk itu, digunakan data gambar beresolusi tinggi dari kamera RGB dan multispektral yang diambil menggunakan *drone* (UAS).

Penelitian ini berfokus pada penggunaan algoritma *DeepForest*, yaitu jaringan saraf konvolusional (CNN) berbasis transfer learning yang digunakan untuk mendeteksi pohon individu dari citra *RGB* resolusi tinggi. *DeepForest* diterapkan dalam dua tahap, yaitu dengan menggunakan model bawaan dan pelatihan ulang (*re-training*) menggunakan data anotasi *bounding box* dari lokasi penelitian agar lebih sesuai dengan karakteristik hutan setempat. Selain itu, data citra dikumpulkan menggunakan sensor multispektral MicaSense RedEdge-MX dan kamera *RGB* DJI Phantom 4 Pro, lalu diproses melalui teknik *Structure from Motion (SfM)* untuk menghasilkan ortofoto dan model tinggi kanopi (CHM) beresolusi 5 cm. Untuk segmentasi kanopi pohon digunakan tiga pendekatan, yaitu *Multiresolution Segmentation (MRS)*, *Superpixel Segmentation* menggunakan algoritma SLICO, dan pendeteksian pohon individu dengan *DeepForest*.

Model *DeepForest* menggunakan patch citra berukuran 400 piksel dengan *patch_overlap* yang ditetapkan sebagai persentase tumpang tindih antar patch sebesar 0,05. Selanjutnya, untuk klasifikasi spesies pohon diterapkan algoritma *Random Forest Classifier* berdasarkan fitur-fitur spektral, tekstur, dan struktural yang diekstrak dari ketiga skala segmentasi tersebut.

Hasil penelitian yaitu mendeteksi pohon individu dengan cukup akurat, menghasilkan *mean Average Precision (mAP)* sebesar 0,65 di area kanopi terbuka dan sekitar 0,5 di kanopi rapat dan tumpang tindih. Meski menghadapi tantangan seperti tumpang tindih kanopi dan variasi bentuk pohon *DeepForest* tetap unggul dalam identifikasi pohon di hutan bercampur. Untuk klasifikasi spesies, pendekatan *superpixel* memberikan akurasi tertinggi hingga 84%, dan meningkat menjadi 93% saat dua spesies *Eucalyptus* digabungkan. Fitur-fitur penting seperti *Normalized Red-Blue Index (NRBI)*, CHM, dan rasio *Red-Edge* terbukti sangat membantu dalam membedakan spesies. Secara keseluruhan, *DeepForest* dan *superpixel segmentation* dinilai sangat potensial untuk diterapkan dalam pemetaan spesies pohon di berbagai jenis hutan alami[22].

Tabel 2.4 Hasil Penelitian Terkait Perbandingan Akurasi Klasifikasi Pohon

Class	Superpixel			Object			Bounding Box		
	Precision	Recall	F-Score	Precision	Recall	F-Score	Precision	Recall	F-Score
<i>Acacia mearnsii</i>	1	0.83	0.91	0.5	0.5	0.5	0.6	0.6	0.6
<i>Allocasuarina spp.</i>	0.81	1	0.9	0.67	0.67	0.67	0.86	0.6	0.71
<i>Callitris rhomboidea</i>	0.86	0.96	0.91	0.89	0.76	0.82	0.83	0.83	0.83
<i>Eucalyptus pulchella</i>	0.79	0.73	0.76	0.77	0.94	0.85	0.58	0.85	0.69
<i>Eucalyptus viminalis</i>	0.6	0.55	0.57	0.79	0.79	0.79	0.5	0.25	0.33
F-Score		0.83			0.77			0.69	

Combining the *Eucalyptus* spp. in a class (Table 7) improved *Eucalyptus* spp. accuracies and hence overall accuracies with superpixels (0.92) and bounding boxes (0.89). At the object scale (0.77), there was no improvement where *Eucalyptus* spp. Precision (0.73) was negatively impacted due to the misclassification of species (*Acacia mearnsii* and *Allocasuarina* spp.) as *Eucalyptus*.

Table 7. Class-specific accuracies: Precision, Recall and F-score with grouped *Eucalyptus* species.

Class	Superpixel			Object			Bounding Box		
	Precision	Recall	F-Score	Precision	Recall	F-Score	Precision	Recall	F-Score
<i>Acacia mearnsii</i>	1	0.83	0.91	0.43	0.38	0.4	0.75	0.6	0.67
<i>Allocasuarina spp.</i>	0.92	1	0.96	1	0.5	0.67	0.88	0.7	0.78
<i>Callitris rhomboidea</i>	0.89	0.96	0.92	0.89	0.76	0.82	0.85	0.92	0.88
<i>Eucalyptus spp.</i>	0.93	0.97	0.95	0.73	0.94	0.82	0.75	0.83	0.79
F-Score		0.93			0.77			0.76	

2.13.4 Decoding Nature's Patterns: An Innovative Approach to Tree Detection Using Deep Learning and High-Resolution Aerial Imagery

(Mendekode Pola Alam Pendekatan Inovatif untuk Deteksi Pohon Menggunakan *Deep Learning* dan Citra Udara Resolusi Tinggi)

Penelitian ini dilakukan oleh Halil İbrahim Şenol dari Harran University dan Abdurahman Yasin Yiğit dari Mersin University, Turki, serta diterbitkan pada tahun 2023 di *Turkish Journal of Geographic Information Systems*. Penelitian ini bertujuan untuk mengembangkan metode deteksi pohon individu di area perkotaan menggunakan citra udara beresolusi tinggi dan algoritma *deepforest*.

Penelitian ini menggunakan algoritma *DeepForest* yaitu model berbasis *Region Convolutional Neural Networks* (R-CNN) yang dirancang untuk mendeteksi kanopi pohon dengan akurasi tinggi. *DeepForest* dijalankan menggunakan *Python* dan library *PyTorch*, memungkinkan proses pelatihan dan prediksi dilakukan dengan cepat dan fleksibel. Untuk meningkatkan akurasi di lokasi studi, model ini dilatih ulang (*fine-tuning*) menggunakan data lokal berupa citra udara beresolusi sangat

tinggi (7 cm) yang diambil dengan *UAV Sensefly Ebee Plus* berbekal kamera *RGB* 20 MP. Pengolahan citra dilakukan menggunakan teknik *Structure from Motion (SfM)* melalui *software Agisoft Metashape* untuk menghasilkan ortomosaik dan model permukaan digital (DSM). Sebagai pembandingan, metode *TreeDetection* di *ArcGIS Pro* juga diuji, namun hasilnya menunjukkan bahwa *DeepForest* lebih unggul dalam akurasi dan mampu mendeteksi pohon dari berbagai ukuran dan kondisi.

Hasil penelitian menunjukkan bahwa model *DeepForest* berhasil mendeteksi pohon individu dengan akurasi tinggi, mencapai *precision* sebesar 82,15%, *recall* sebesar 94,59%, dan *F1-score* sebesar 87,94%, dengan tingkat akurasi keseluruhan sebesar 80,87%. Dibandingkan dengan model dari *ArcGIS Pro*, *DeepForest* menunjukkan performa yang jauh lebih unggul, karena *ArcGIS Pro* hanya mencapai akurasi 48,95%. *DeepForest* mampu mendeteksi berbagai ukuran pohon, mulai dari kecil hingga besar, dengan tingkat *false positive* yang rendah. Penelitian ini membuktikan bahwa kombinasi citra udara beresolusi tinggi dengan metode *deep learning* seperti *DeepForest* sangat efektif untuk pemetaan pohon di area urban dan dapat diadaptasi untuk penggunaan skala global di masa depan[23].

Tabel 2.5 Hasil Penelitian Terkait Perbandingan DeepForest Dan ArcGIS Pro

	Total	TP	FP	FN	Recall	F1 Score	Precision (UAcc)
Deep Forest	213	175	38	10	0.9459	0.8794	0.8215
ArcGIS Pro	70	32	38	35	0.48	0.47	0.46

2.13.5 *DeepForest: A Python package for RGB deep learning tree crown delineation*

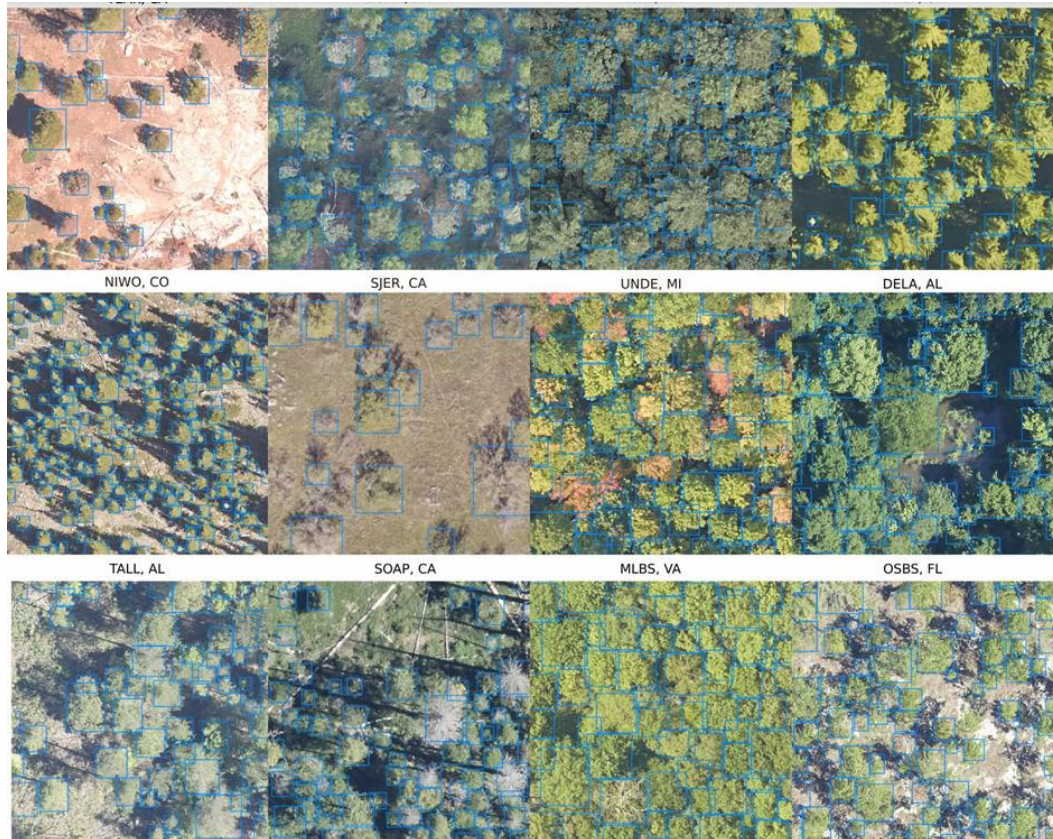
(*DeepForest*: Paket Python untuk Delineasi Tajuk Pohon Menggunakan Deep Learning dari Citra RGB)

Penelitian ini dilakukan oleh Ben G. Weinstein, Sergio Marconi, Méline Aubry-Kientz, Gregoire Vincent, Henry Senyondo, dan Ethan White. Tujuan utama dari penelitian ini adalah mengembangkan dan memperkenalkan *DeepForest* dengan sebuah paket Python open-source yang menggunakan teknik *deep learning* untuk mendeteksi dan membuat batas kanopi pohon individu dari citra *RGB* beresolusi tinggi.

Penelitian ini menggunakan algoritma *DeepForest* dengan mengandalkan model pra-latih yang dikembangkan dengan menggunakan lebih dari 30 juta kanopi pohon yang dihasilkan secara algoritmik dari 22 hutan, dan disempurnakan dengan 10.000 kanopi pohon yang diberi label manual dari 6 hutan yang berbeda. Data citra *RGB* beresolusi tinggi digunakan, dan pengolahan awal citra dilakukan dengan teknik *Structure from Motion* (SfM) melalui *Agisoft Metashape* untuk menghasilkan ortomosaik dan model permukaan digital (DSM). Model *DeepForest* dapat diterapkan langsung pada data baru atau dilatih ulang (*fine-tuning*) menggunakan dataset lokal. *DeepForest* juga dibandingkan dengan metode lain seperti *ArcGIS Pro TreeDetection*, untuk mengevaluasi efektivitasnya.

Hasil penelitian menunjukkan bahwa *DeepForest* mampu mendeteksi pohon individu dengan akurasi tinggi di berbagai tipe hutan dan kondisi data. Pada dataset dari *National Ecological Observatory Network (NEON)* di Amerika Serikat, model mencapai rata-rata 72% *recall* dan 64% *precision*. Selain itu, pengujian pada hutan tropis di French Guiana menunjukkan bahwa model prebuilt *DeepForest* tanpa pelatihan ulang memiliki *recall* 0,64, sedangkan setelah *fine-tuning* dengan data lokal, *recall* meningkat menjadi 0,78, mengungguli metode segmentasi pohon berbasis *LIDAR* terbaik dalam kompetisi terkait. Di lingkungan perkotaan, seperti pada studi pohon jalanan di Portland, *retraining model* menggunakan data lokal

meningkatkan *recall* dari 0,55 menjadi 0,72, serta mengurangi tingkat undersegmentation. Secara keseluruhan, *DeepForest* terbukti efektif dan fleksibel dalam berbagai kondisi geografis dan resolusi citra, serta menunjukkan potensi besar dalam mendukung penelitian kehutanan *berbasis remote sensing*[24].



Gambar 2.6 Hasil Penelitian Terkait Deteksi Pohon dari Berbagai Lokasi

2.13.6 Injecting Domain Knowledge Into Deep Neural Networks for Tree Crown Delineation

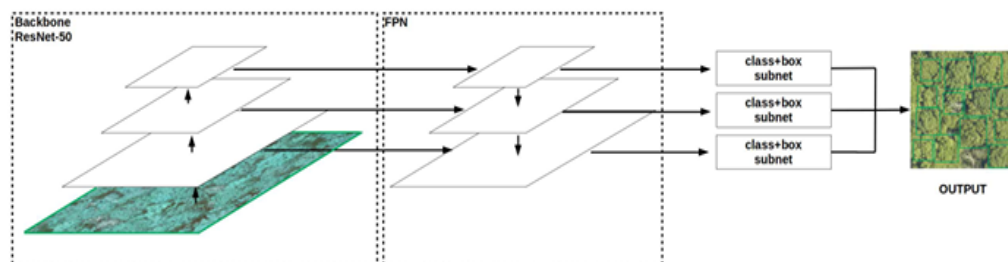
(Memasukkan Pengetahuan Domain ke dalam Jaringan Saraf Tiruan Dalam *Deep Neural Networks* untuk Delineasi Tajuk Pohon)

Penelitian ini dilakukan oleh Ira Harmon, Sergio Marconi, Ben Weinstein, Sarah Graves, Daisy Zhe Wang, Alina Zare, Stephanie Bohlman, Aditya Singh, dan Ethan White, serta dipublikasikan pada tahun 2022 melalui TechRxiv. Tujuan penelitian ini adalah untuk meningkatkan akurasi delineasi kanopi pohon individu dari citra udara melalui pendekatan berbasis *deep learning*. Secara khusus, penelitian ini

menggunakan *DeepForest* sebuah model berbasis *Convolutional Neural Network* (CNN) yang bertujuan untuk mendeteksi dan mendelineasi batas kanopi pohon dengan lebih akurat.

Penelitian ini menggunakan *DeepForest*, yaitu model berbasis *Convolutional Neural Network* (CNN) yang dibangun di atas arsitektur RetinaNet dengan backbone ResNet-50. *DeepForest* dilatih menggunakan citra udara *RGB* beresolusi tinggi (0,1 meter per piksel) dari *NEON* (*National Ecological Observatory Network*). Proses pelatihan dilakukan secara bertahap *pre-training* di *ImageNet*, pembuatan dataset otomatis dengan *self-supervised learning* dan pelatihan ulang menggunakan label manual. *DeepForest* hanya menggunakan data citra *RGB* tanpa memanfaatkan data *LiDAR*, dan bertujuan mendeteksi kanopi pohon individu secara otomatis dengan output berupa *bounding box* di setiap pohon.

Hasilnya, *DeepForest* berhasil mendeteksi kanopi pohon dengan akurasi tinggi, menghasilkan nilai *F1-score* yang baik di berbagai lokasi hutan. Model ini lebih akurat dibandingkan metode *non-deep learning* sebelumnya, terutama dalam mendeteksi pohon di area yang memiliki variasi ukuran dan jarak antar pohon. Namun, model tetap membutuhkan banyak data anotasi manual untuk mencapai performa optimal, dan kinerjanya sedikit menurun pada area dengan kanopi pohon yang sangat rapat[25].



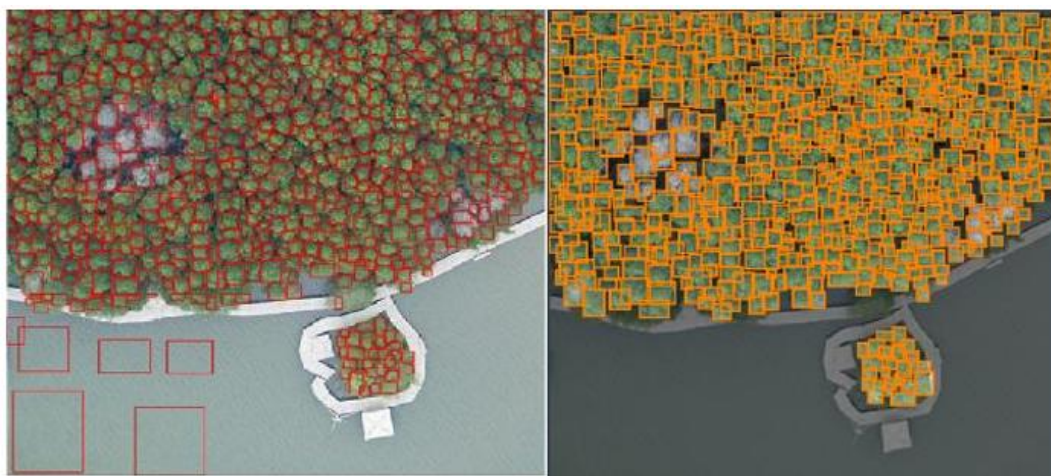
Gambar 2.7 Hasil Penelitian Terkait Alur Model Deteksi Pohon

2.13.7 Penerapan *Deepforest* Untuk Deteksi dan Estimasi Jumlah Pohon Bakau di Petengoran

Penelitian ini dilakukan oleh Abdillah Halim Hanafi serta dipublikasikan pada tahun 2024. Penelitian ini bertujuan mengatasi keterbatasan metode konvensional yang memerlukan banyak waktu dan sumber daya manusia, khususnya dalam pelestarian hutan mangrove.

Penelitian ini menggunakan algoritma *DeepForest* berbasis RetinaNet untuk mendeteksi kanopi pohon bakau dari citra udara resolusi tinggi. Data dikumpulkan menggunakan drone Fixed-Wing dengan kamera Sony Nex-5 dan diolah menjadi file GeoTIFF. Proses anotasi pohon dilakukan manual menggunakan *Roboflow*, dan pelatihan model dilakukan di *Google Colab* dengan uji kombinasi *hyperparameter* seperti *epoch*, *learning rate*, *optimizer*, dan *batch size*. Evaluasi performa model menggunakan metrik *Precision*, *Recall*, dan *F1-Score*.

Hasil Penelitian *Model DeepForest* awal memiliki *precision* tinggi (0,98) tetapi *recall* rendah (0,69). Setelah pelatihan ulang, model terbaik dengan *epoch* 50, *learning rate* 0.0001, *optimizer* SGD momentum 0.9, dan *batch size* 32, berhasil meningkatkan performa dengan *precision* 0,98, *recall* 0,95, dan *F1-score* 0,96. Pada pengujian data geospasial, model mencatat *precision* sebesar 0,99, membuktikan efektivitas *DeepForest* dalam mendeteksi pohon bakau di area kanopi rapat[26].



Gambar 2.8 Hasil Penelitian Terkait Evaluasi Model Deteksi Pohon

2.13.8 *Computer Vision System for Detecting Orchard Trees from UAV Images*

(Sistem Computer Vision untuk Mendeteksi Pohon Kebun dari Citra UAV)

Penelitian ini dilakukan oleh Hela Jemaa, Wassim Bouachir, Brigitte Leblon, dan Nizar Bouguila. Kemudian penelitian ini dipublikasikan pada tahun 2022. Tujuan utama dari penelitian ini adalah mengembangkan metode otomatis untuk mendeteksi pohon apel di kebun menggunakan citra multispektral *UAV* dengan mengutamakan penerapan model *deep learning*. *DeepForest* dipilih sebagai fokus utama karena kemampuannya dalam mendeteksi mahkota pohon secara akurat pada citra beresolusi tinggi.

Penelitian ini menggunakan algoritma *DeepForest*, yaitu model *deep learning* berbasis *RetinaNet* dengan backbone *ResNet* dan *Feature Pyramid Network* (FPN), yang dirancang untuk mendeteksi mahkota pohon dari citra *UAV*. Model ini telah dilatih sebelumnya dengan jutaan data dari 22 lokasi *NEON*. Selain *DeepForest*, model YOLO-v5 juga digunakan sebagai pembanding. Dataset dibuat dari citra *UAV* kamera *MicaSense RedEdge*, dipotong menjadi *patch* 512×512 piksel, lalu dianotasi secara manual. Data dibagi menjadi pelatihan, validasi, dan pengujian. Kedua model diuji dengan pengaturan yang sama dan dievaluasi menggunakan *precision*, *recall*, *F1-score*, dan *mAP*, dengan standar *IoU* minimal 50%.

Hasil penelitian menunjukkan bahwa *DeepForest* menghasilkan performa terbaik dalam mendeteksi pohon apel. Model ini mencapai *precision* sebesar 90,8%, *recall* sebesar 83%, *F1-score* sebesar 86,8%, dan *mAP* sebesar 81,2%, melebihi semua subversi model YOLO-v5 yang diuji. Secara visual, *DeepForest* mampu mendeteksi pohon dengan presisi tinggi meskipun ada tantangan seperti variasi umur pohon dan kemunculan jenis pohon lain di area kebun. Oleh karena itu, penelitian ini menegaskan bahwa *DeepForest* merupakan pilihan metode yang unggul untuk deteksi pohon apel berbasis citra *UAV*, dan mengarahkan penelitian

lanjutan untuk memperluas penggunaan data multispektral dan pendekatan semi-*supervised* untuk lebih meningkatkan akurasi[27].

Tabel 2.6 Hasil Penelitian Terkait Perbandingan DeepForest dan Yolo

Model	Precision	Recall	F1-score	mAP
DeepForest	0.908	0.830	0.868	0.812
YOLO-v5s	0.742	0.743	0.742	0.742
YOLO-v5m	0.668	0.706	0.686	0.69
YOLO-v5l	0.719	0.548	0.626	0.642

2.13.9 Implementing Deep Learning Algorithms for Urban Tree Detection and Geolocation with High-Resolution Aerial, Satellite, and Ground-Level Images

(Implementasi Algoritma *Deep Learning* untuk Deteksi dan Geolokasi Pohon Perkotaan Menggunakan Citra Resolusi Tinggi dari Udara, Satelit, dan Permukaan Tanah)

Penelitian ini dilakukan oleh Luisa Velasquez-Camacho, Maddi Etxegarai, dan Sergio de-Miguel. Penelitian ini diterbitkan pada tahun 2023 dalam jurnal *Computers Environment and Urban Systems*. Tujuan dari penelitian ini adalah untuk mendeteksi dan menentukan lokasi pohon perkotaan secara otomatis menggunakan citra satelit dan udara beresolusi tinggi, dengan mengutamakan penggunaan model *DeepForest* untuk identifikasi kanopi pohon.

Metode penelitian ini mengutamakan penggunaan algoritma *DeepForest* model deteksi pohon berbasis CNN yang telah dipra-latih dengan lebih dari 10 juta gambar kanopi pohon. *DeepForest* digunakan untuk mendeteksi mahkota pohon dari citra satelit dan udara beresolusi tinggi, kemudian dilatih ulang menggunakan 979 citra dari berbagai kota dunia yang telah dianotasi secara manual. Model ini dibandingkan dengan YOLOv5 dan Faster R-CNN, namun *DeepForest* menunjukkan keunggulan terutama dalam mengolah citra berukuran besar. Pelatihan dilakukan di platform SEPAL dengan evaluasi berdasarkan *precision*,

recall, dan *F1-score* menggunakan data validasi dari inventarisasi pohon kota Lleida, Spanyol.

Hasil penelitian menunjukkan bahwa *DeepForest* lebih unggul dibandingkan YOLOv5 dalam mendeteksi kanopi pohon dari citra udara dan satelit. *DeepForest* mencapai *precision* sebesar 79,6%, *recall* 77,5%, dan *F1-score* 78,0, yang lebih tinggi dibandingkan semua versi YOLOv5 yang diuji. Selain itu, *DeepForest* mampu memproses citra berukuran besar hingga 7000×6000 piksel tanpa kehilangan akurasi, yang tidak dapat dilakukan oleh YOLOv5. Dalam uji pemetaan lokasi pohon, *DeepForest* juga berhasil mendeteksi dan menggeolokasi hingga 63,5% pohon yang terlihat dengan rata-rata kesalahan posisi hanya 0,6 meter dari pusat kanopi. Keunggulan ini membuktikan bahwa *DeepForest* lebih andal dan efisien untuk pemetaan pohon perkotaan berbasis citra udara[28].

Tabel 2.7 Hasil Penelitian Terkait Kinerja Model Deteksi Pohon

Objective	Model	Precision (%)	Recall (%)	F-Score	Train epochs (n)
Tree detection	YOLOV5x	86.9	83.1	84.9	60
	Deepforest	79.6	77.5	78.0	15
	YOLOV5s	74.8	69.8	72.2	25
	YOLOV5m	72.5	66.0	69.1	25
	YOLOV5n	70.1	65.0	67.5	25
	YOLOV5l	75.3	70.6	72.9	25
	YOLOV5x	74.7	68.7	71.6	25
	YOLOV5n6 (Scratch)	70.8	62.9	66.6	300
	Faster RCNN	74.6	25.7	35.6	100

2.13.10 *Estimation of Damaged Regions by the Bark Beetle in a Mexican Forest Using UAV Images and Deep Learning*

(Estimasi Wilayah Kerusakan yang Disebabkan oleh Bark Beetle pada Hutan di Meksiko Menggunakan Citra UAV dan Metode Deep Learning)

Penelitian ini dilakukan oleh eh Gildardo Godinez-Garrido, Juan-Carlos Gonzalez-Islas, Angelina Gonzalez-Rosas, Mizraim U. Flores, Juan-Marcelo Miranda-Gomez, dan Ma. de Jesus Gutierrez-Sanchez. Penelitian ini dipublikasikan pada tahun 2024 di jurnal *Sustainability*. Tujuan utama dari penelitian ini adalah untuk mendeteksi dan memetakan secara otomatis pohon sehat dan pohon yang rusak akibat serangan serangga penggerek kulit kayu (*bark beetle*) di hutan negara bagian Hidalgo, Meksiko, dengan menggunakan citra *UAV* dan algoritma *deep learning*, terutama dengan mengutamakan *DeepForest* untuk deteksi pohon sehat.

Penelitian ini menggunakan dua algoritma *deep learning*, yaitu *DeepForest* dan *YOLOv5*, untuk mendeteksi kondisi pohon di hutan. *DeepForest* menjadi algoritma utama yang digunakan untuk mendeteksi pohon sehat dari citra *UAV* karena kemampuannya mengenali bentuk kanopi secara akurat. Sementara itu, *YOLOv5* digunakan untuk mendeteksi pohon yang rusak akibat serangan hama. Gambar udara diproses secara otomatis, dan setiap pohon yang terdeteksi dianalisis posisinya serta kondisi kesehatannya.

Hasil penelitian menunjukkan bahwa *DeepForest* lebih unggul dibandingkan *YOLOv5* dalam mendeteksi pohon sehat dari citra *UAV*. Dengan akurasi mencapai 88%, *DeepForest* secara konsisten berhasil mengenali pohon-pohon yang tidak terinfeksi, bahkan dalam kondisi lingkungan yang kompleks. Sebaliknya, *YOLOv5* digunakan untuk mendeteksi pohon yang terinfeksi, dan meskipun mencapai akurasi tinggi sebesar 90%, performanya bergantung pada ciri visual kerusakan yang lebih mencolok. Keunggulan *DeepForest* terletak pada kemampuannya mengolah citra resolusi tinggi, mengenali struktur kanopi secara detail, dan memberikan hasil yang stabil tanpa perlu banyak penyesuaian. Hal ini menjadikan

DeepForest lebih andal sebagai dasar sistem *monitoring* kesehatan hutan secara otomatis[29].

Tabel 2.8 Hasil Penelitian Terkait Data Lokasi Dan Kondisi Pohon

Pt	Lat (°)	Long (°)	Alt (m)	Dam. (ha)	Rev. (ha)	Trees		
						Tot.	Pst.	Hlthy
WP1	20.1155	-98.6224	101	0.2099	1.3791	73	39	34
WP2	20.1159	-98.6212	101	0.1686	1.3791	73	32	41
WP3	20.1152	-98.6207	101	0.1740	1.3791	86	21	65
WP4	20.1148	-98.6219	100.5	0.3500	1.3791	59	32	27
WP5	20.1140	-98.6215	100.2	0.1482	1.3791	88	15	73
WP6	20.1143	-98.6203	99.5	0.3400	1.3791	76	20	56
Total	-	-	-	1.3907	8.2743	455	159	296
%	-	-	-	16.80%	100%	100%	34.95%	65.05%

2.13.11 *Evaluation of the quality of automatic tree detection using photogrammetric canopy height models and orthomosaic.*

(Penilaian Kualitas Deteksi Pohon Secara Otomatis Menggunakan Model Tinggi Kanopi Fotogrametri dan Citra Orthomosaic)

Penelitian ini dilakukan oleh penelitian ini dilakukan oleh Natalya V. Ivanova, Aleksandr V. Lebedev, dan Maxim P. Shashkov. Dipublikasi pada tahun 2023. Tujuan dari penelitian ini adalah untuk mengevaluasi kualitas deteksi otomatis pohon menggunakan model ketinggian kanopi dan ortofoto dengan algoritma *DeepForest*.

Penelitian ini menggunakan metode utama deteksi otomatis mahkota pohon dengan algoritma *DeepForest* yaitu model jaringan saraf *konvolusional* berbasis *Python* yang dilatih secara *semi-supervised*. Data diperoleh dari citra udara resolusi tinggi menggunakan drone DJI Phantom 4, yang kemudian diolah menjadi ortofoto melalui perangkat lunak *Agisoft Metashape*. *DeepForest* digunakan untuk mendeteksi pohon pada ortofoto resolusi 10 cm/piksel dengan pengujian berbagai ukuran *tile* (200–800 piksel). Sebagai perbandingan, digunakan pula metode berbasis *canopy height model* (CHM) yang dibangun dari *point cloud* hasil fotogrametri dan diproses menggunakan fungsi *FindTreesCHM()* dari paket R dan LiDAR.

Hasil Penelitian menunjukkan deteksi pohon menggunakan *DeepForest* menunjukkan tingkat akurasi yang tinggi. Pada *tile* ukuran 400 piksel, nilai *precision* (*p*) mencapai 0,98, *recall* (*r*) sebesar 0,91, dan *F1-score* sebesar 0,94. Untuk pohon konifer, *F1-score* mencapai hingga 0,94, sementara untuk pohon berdaun lebar (*deciduous*), *F1-score* sekitar 0,81–0,94 tergantung ukuran *tile*. Sementara itu, metode CHM menghasilkan *F1-score* maksimum 0,95 untuk konifer pada parameter minimum tinggi pohon 10 m, namun deteksi terhadap pohon berdaun lebar lebih rendah, dengan *F1-score* sekitar 0,78–0,80. Secara keseluruhan, *DeepForest* lebih unggul dalam mendeteksi pohon di kawasan dengan kanopi terbuka dan pohon-pohon tersebar, terutama karena kemampuannya mengenali bentuk mahkota pohon dari citra *RGB*[30].

Tabel 2.9 Hasil Penelitian Terkait Evaluasi Deteksi Pohon

Tile size, pixels	Detected trees	TP	FP	FN	<i>p</i>	<i>r</i>	<i>F</i>
400	Coniferous	45	0	6	1	0.88	0.94
	Deciduous	76	3	6	0.96	0.93	0.94
	Total	121	3	12	0.98	0.91	0.94
600	Coniferous	45	2	4	0.96	0.92	0.94
	Deciduous	58	5	22	0.92	0.73	0.81
	Total	103	7	26	0.94	0.80	0.86

2.13.12 UAV-Based Computer Vision System for Orchard Apple Tree Detection and Health Assessment

(Sistem Visi Komputer Berbasis UAV untuk Deteksi dan Analisis Kesehatan Pohon Apel di Lahan Perkebunan)

Penelitian ini dilakukan oleh Hela Jemaa, Wassim Bouachir, Brigitte Leblon, Armand LaRocque, Ata Haddadi, dan Nizar Bouguila. Dipublikasi pada tahun 2023 dalam jurnal *Remote Sensing*. Tujuan utama dari penelitian ini adalah mengembangkan kerangka kerja berbasis *UAV* (unmanned aerial vehicle) dan visi komputer untuk mendeteksi pohon apel secara individual serta menilai kondisi kesehatannya secara otomatis.

Penelitian ini mengandalkan *DeepForest* sebagai algoritma utama untuk mendeteksi pohon apel secara otomatis dari citra udara beresolusi tinggi yang diambil menggunakan drone. *DeepForest* adalah algoritma berbasis *deep learning* yang memang dirancang khusus untuk mengenali pohon dari atas, sehingga sangat cocok untuk tugas ini. Untuk meningkatkan ketepatan deteksi, peneliti menambahkan teknik *hard negative mining*, yaitu dengan memasukkan contoh objek-objek yang sering keliru dikenali sebagai pohon ke dalam pelatihan agar *DeepForest* bisa belajar membedakannya. Meskipun model lain seperti YOLO-v5 juga diuji, hasil menunjukkan bahwa *DeepForest* memberikan akurasi yang lebih tinggi dan lebih stabil, sehingga dipilih sebagai metode utama. Setelah pohon berhasil dikenali, langkah berikutnya adalah menilai kondisi kesehatannya menggunakan 12 indeks vegetasi dari citra multispektral. Penilaian ini dilakukan dengan model *machine learning* dan *Random Forest* terbukti paling akurat dalam membedakan pohon sehat dan tidak sehat.

Hasil penelitian menunjukkan bahwa *DeepForest* yang diterapkan bersama strategi HNM memberikan hasil deteksi terbaik dengan *F1-score* sebesar 86,24%, sedikit lebih unggul dibanding YOLO-v5 yang memperoleh *F1-score* 84,81%. Untuk klasifikasi kesehatan pohon, model *Random Forest* mencapai akurasi tertinggi yaitu 97,52%, dengan *F1-score* masing-masing 98,8% untuk pohon sehat dan 95,5% untuk pohon tidak sehat[31].

Tabel 2.10 Hasil Penelitian Terkait Tabel Performa Model

Baseline Model	P_d (%)	R_d (%)	$F1-Score_d$ (%)
<i>Without HNM</i>			
Baseline 1: DeepForest	84.82	86.18	85.46
Baseline 2: YOLO	79.40	88.05	82.64
<i>With HNM</i>			
Baseline 1: DeepForest	85.85	86.67	86.24
Baseline 2: YOLO	82.01	88.99	84.81

2.13.13 *Individual canopy tree species maps for the National Ecological Observatory Network*

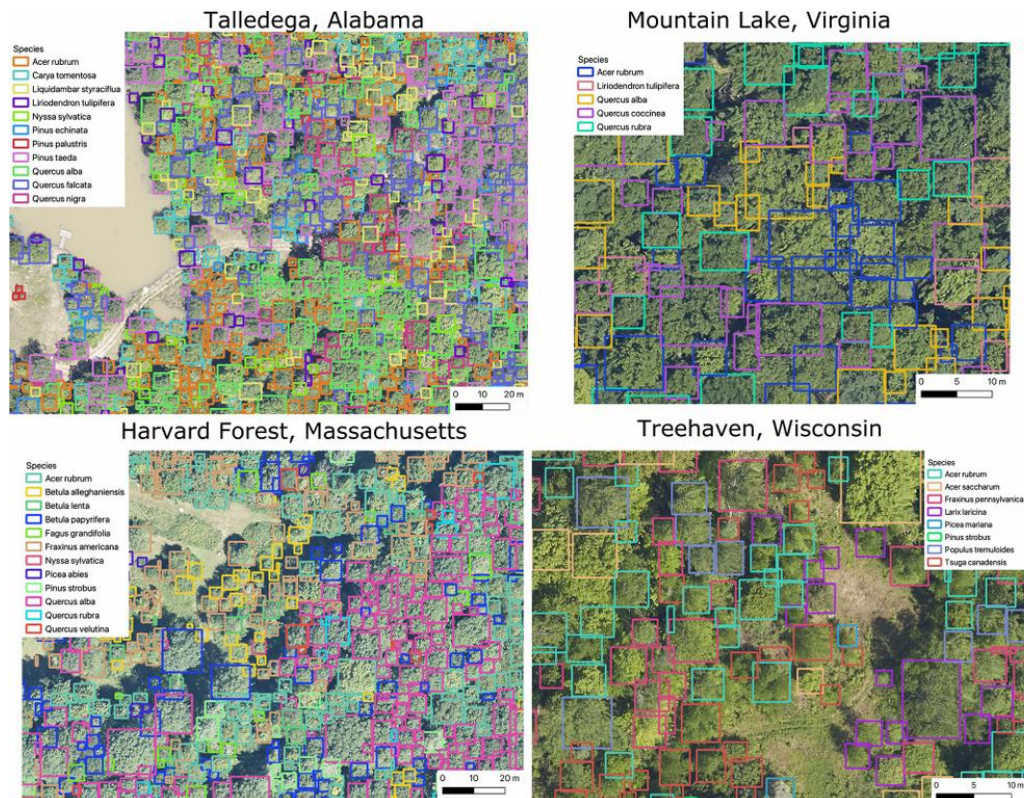
(Peta Spesies Pohon Berbasis Kanopi Individual untuk National Ecological Observatory Network)

Penelitian ini dilakukan oleh Ben G. Weinstein dan tim, yang diterbitkan pada tahun 2024 di jurnal PLOS Biology. Penelitian ini bertujuan untuk menghasilkan peta spesies pohon kanopi individual dalam skala besar di berbagai lokasi di Amerika Serikat menggunakan data dari *National Ecological Observatory Network (NEON)*.

Penelitian ini mengandalkan algoritma *DeepForest* yaitu model *deep learning* berbasis *Python* yang dirancang khusus untuk mendeteksi dan memetakan mahkota pohon individu dari citra udara *RGB*. *DeepForest* digunakan untuk memprediksi lokasi, ukuran, dan bentuk mahkota pohon, dengan tingkat akurasi tinggi untuk mendeteksi pohon pada lapisan kanopi. Model ini dilatih menggunakan lebih dari 40.000 titik pohon lapangan dari 24 lokasi *NEON*, serta dilengkapi data dari sensor udara seperti citra *RGB* 10 cm, model ketinggian kanopi (*LiDAR*), dan data hiperspektral 1 m resolusi. Setelah pohon terdeteksi oleh *DeepForest* klasifikasi spesies dilakukan menggunakan model hierarkis multi-temporal berbasis data hiperspektral. Setiap lokasi *NEON* memiliki model klasifikasi spesies yang disesuaikan secara khusus untuk meningkatkan akurasi dan mengatasi ketidakseimbangan data antar spesies.

Berdasarkan hasil penelitian algoritma *DeepForest* menunjukkan keunggulan yang signifikan dalam mendeteksi mahkota pohon secara individual dari citra udara, terutama pada kawasan hutan dengan kanopi terbuka dan jumlah spesies yang tidak terlalu banyak. *DeepForest* mampu mengidentifikasi posisi, ukuran, dan bentuk mahkota pohon secara akurat, serta memberikan dasar yang kuat untuk klasifikasi spesies lanjutan. Keunggulannya terletak pada kemampuannya menangani data berskala besar dan beragam, serta performa stabil di berbagai tipe ekosistem. Berdasarkan hasil evaluasi, model klasifikasi spesies yang dibangun dari deteksi *DeepForest* berhasil memetakan lebih dari 100 juta pohon dari 81 spesies di 24

lokasi *NEON*. Rata-rata akurasi mikro yang dicapai adalah 78,8%, dan akurasi makro sebesar 75,8%. Hasil akurasi tertinggi dicapai di San Joaquin Experimental Range (SJER), California, dengan nilai 100% untuk kedua metrik akurasi, diikuti oleh Great Smoky Mountains (GRSM) sebesar 90%, dan Teakettle Canyon (TEAK) dengan akurasi mikro 82% dan makro 83% [32].



Gambar 2.9 Hasil Penelitian Terkait Visualisasi Anotasi Spesies Pohon

2.13.14 *Monitoring the health of agricultural ecosystems from remote sensing data using semi-supervised neural networks*

(Pemantauan Kesehatan Ekosistem Pertanian dari Data Penginderaan Jauh Menggunakan Jaringan Saraf Semi-Supervised)

Penelitian ini dilakukan oleh Ingolf Roemer, Martin Schieck, Nick Harnau, dan Bogdan Franczyk dari Leipzig University dan Wroclaw University of Economics and Business. Artikel ini diterbitkan pada tahun 2024 dalam *Procedia Computer*

Science. Tujuan dilakukan penelitian ini untuk mendeteksi pohon pinus secara otomatis dari citra drone dan membedakan antara pohon sehat dan tidak sehat.

Penelitian ini menggunakan algoritma *DeepForest* untuk mendeteksi dan membedakan pohon pinus sehat dan tidak sehat dari citra *drone*. *DeepForest* yang berbasis CNN, diadaptasi menjadi model klasifikasi dua kelas dan dilatih dengan data dari 145 citra udara yang dianotasi secara manual. Dibandingkan model lain seperti YOLOv5, *DeepForest* terbukti lebih efisien karena membutuhkan lebih sedikit data pelatihan namun tetap memberikan akurasi tinggi.

Hasil penelitian menunjukkan bahwa *DeepForest* sangat efektif dan akurat dalam mengenali serta mengklasifikasikan pohon sehat dan tidak sehat. Model terbaik dicapai pada kombinasi 40 *epoch* dan 58 citra latih, dengan hasil *F1-score* sebesar 0,84 untuk pohon sehat dan 0,76 untuk pohon tidak sehat, serta *F1-score* keseluruhan di level kelas sebesar 0,79. Selain itu model menghasilkan *precision* sebesar 0,81 dan *recall* sebesar 0,77, menunjukkan keseimbangan yang baik dalam kinerja deteksi. Hasil ini membuktikan bahwa *DeepForest* tidak hanya unggul dalam deteksi pohon individu tetapi juga sangat menjanjikan untuk pemantauan kesehatan pohon secara otomatis dalam konteks pertanian dan kehutanan. Keunggulan lain dari pendekatan ini adalah efisiensi waktu, karena hanya membutuhkan sekitar 213 menit untuk anotasi per kelas, jauh lebih cepat dibanding model *supervised* seperti YOLOv5 yang membutuhkan hingga 1.333 menit. Penelitian ini menegaskan bahwa pendekatan *semi-supervised* dengan *DeepForest* adalah solusi yang kuat dan hemat sumber daya untuk pemantauan ekosistem menggunakan data *UAV* [33].

Tabel 2.11 Hasil Penelitian Terkait Evaluasi Deteksi Pohon Pinus

total	Metrics at box level with an IoU > 0.4	Metrics with a focus on unhealthy pines	Metrics with a focus on healthy pines	Total metrics at class level
6523 true pines	Number of TP: 5066	Number of TP: 3235	Number of TP: 1812	Number of TP: 5047
4243 unhealthy pines	Number of FP: 1196	Number of FP: 980	Number of FP: 235	Number of FP: 1215
2280 healthy pines	Number of FN: 1457	Number of FN: 1008	Number of FN: 468	Number of FN: 1476
6262 predictions with a score > 0.1	Box_Precision = 0.81	Precision_individual_class = 0.77	Precision_individual_class = 0.89	Precision_all_classes = 0.81
4215 predictions for unhealthy pines	Box_Recall = 0.78	Recall_individual_class = 0.76	Recall_individual_class = 0.79	Recall_all_classes = 0.77
2047 predictions for healthy pines	F1-Score = 0.79	F1 score = 0.76	F1 score = 0.84	F1 score = 0.79

2.13.15 Estimating the Number of Trees in Margasari Mangrove Forests of Lampung Through Aerial Images Using Adaptive Thresholding and Contour Extraction Methods.

(Estimasi Jumlah Pohon di Hutan Mangrove Margasari, Lampung Menggunakan Citra Udara Berbasis Metode *Adaptive Thresholding* dan Ekstraksi Kontur)

Penelitian ini dilakukan oleh Mona A. Batubara, Syaiful Alam, Rizka W.A. Utami, Helmy Fitriawan, dan Ardian Ulvan dari Universitas Lampung serta Khoirin Nisa dari Universitas Lampung (Departemen Matematika). Artikel ini diterbitkan pada tahun 2021 dalam *prosiding Advances in Engineering Research, Atlantis Press*. Tujuan penelitian ini adalah mengestimasi jumlah pohon mangrove secara otomatis menggunakan citra udara UAV (*drone*) serta mengukur tingkat akurasinya melalui metode pemrosesan citra digital.

Penelitian ini menerapkan algoritma *Adaptive Thresholding dan Contour Extraction* untuk mendeteksi objek pohon pada citra udara UAV. Pemrosesan citra dilakukan menggunakan bahasa pemrograman *Python* dengan pustaka *OpenCV*, yang meliputi tahapan konversi RGB ke *grayscale*, *median filtering* untuk mengurangi *noise*, segmentasi dengan *adaptive thresholding*, pemisahan objek menggunakan operasi *morphological opening*, serta pendeteksian objek melalui

fungsi *findContours* dan *drawContours*. Perhitungan jumlah pohon dilakukan berdasarkan jumlah kontur yang terdeteksi. Peneliti melakukan validasi dengan membandingkan hasil deteksi otomatis terhadap perhitungan manual pada 32 sampel citra dan pengecekan langsung di 4 titik area mangrove untuk mengukur tingkat kesalahan sistem serta menganalisis pengaruh nilai konstanta pada *adaptive thresholding*.

Hasil yang diperoleh menunjukkan bahwa metode ini mampu mendeteksi pohon mangrove dengan baik, di mana nilai konstanta terbaik berada pada $C = -1$. Sistem menghasilkan tingkat kesalahan (*error*) sebesar 3,3% dibandingkan perhitungan manual. Berdasarkan pemrosesan keseluruhan citra *orthophoto* total pohon mangrove yang berhasil terdeteksi berjumlah 69.227 pohon. Waktu pemrosesan juga tergolong efisien, yaitu 21 detik untuk citra 500×500 piksel dan 54 detik untuk citra 16.190×19.220 piksel, menunjukkan bahwa metode ini efektif digunakan untuk pemantauan vegetasi berbasis citra UAV[34].

Tabel 2.12 Hasil Penelitian Terkait Deteksi Jumlah Pohon

Image	RGB images	The number of trees detected
(a)		7.407
(b)		30.433
(c)		18.953
(d)		12.484
Total		69.227

2.13.16 A benchmark dataset for canopy crown detection and delineation in co-registered airborne RGB, LiDAR and hyperspectral imagery from the National Ecological Observation Network

(Dataset Benchmark untuk Deteksi dan Delineasi Tajuk Pohon dalam Citra Airborne RGB, LiDAR, dan Hiperspektral yang Terkoregistrasi dari *National Ecological Observation Network*)

Penelitian ini dilakukan Ben G. Weinstein, Sarah J. Graves, Sergio Marconi, yang diterbitkan pada tahun 2021 di jurnal *PLOS Computational Biology*. Tujuan dari penelitian ini adalah menyediakan dataset standar untuk mendeteksi dan menggambar mahkota pohon secara individu dari citra udara. Dataset ini digunakan untuk mengevaluasi dan membandingkan algoritma seperti *DeepForest* secara terbuka dan konsisten.

Dengan menggunakan algoritma *DeepForest* yaitu sebuah model *deep learning* berbasis RetinaNet yang dikembangkan khusus untuk mendeteksi *bounding box* mahkota pohon dari citra *RGB*. Model *DeepForest* diuji menggunakan data *RGB* resolusi tinggi (0,1 m²/piksel) yang telah diproses dan direktifikasi, dan hasil prediksinya dibandingkan dengan data anotasi mahkota pohon dari citra dan lapangan. Evaluasi dilakukan dengan menghitung metrik *precision*, *recall*, dan *intersection-over-union (IoU)*, menggunakan ambang kecocokan sebesar 0,4 untuk menyatakan prediksi yang benar. Penilaian dilakukan terhadap tiga jenis data evaluasi: anotasi gambar (6490 pohon), anotasi lapangan (562 pohon), dan titik batang dari survei vegetasi *NEON* (4365 pohon)[35].

Tabel 2.13 Hasil Penelitian Terkait Akurasi Pohon Menggunakan DeepForest

Image-annotated Crowns		Field-annotated Crowns	Field-collected Stems
Recall	Precision	Recall	Recall
79.0	65.9	72.2	74.0

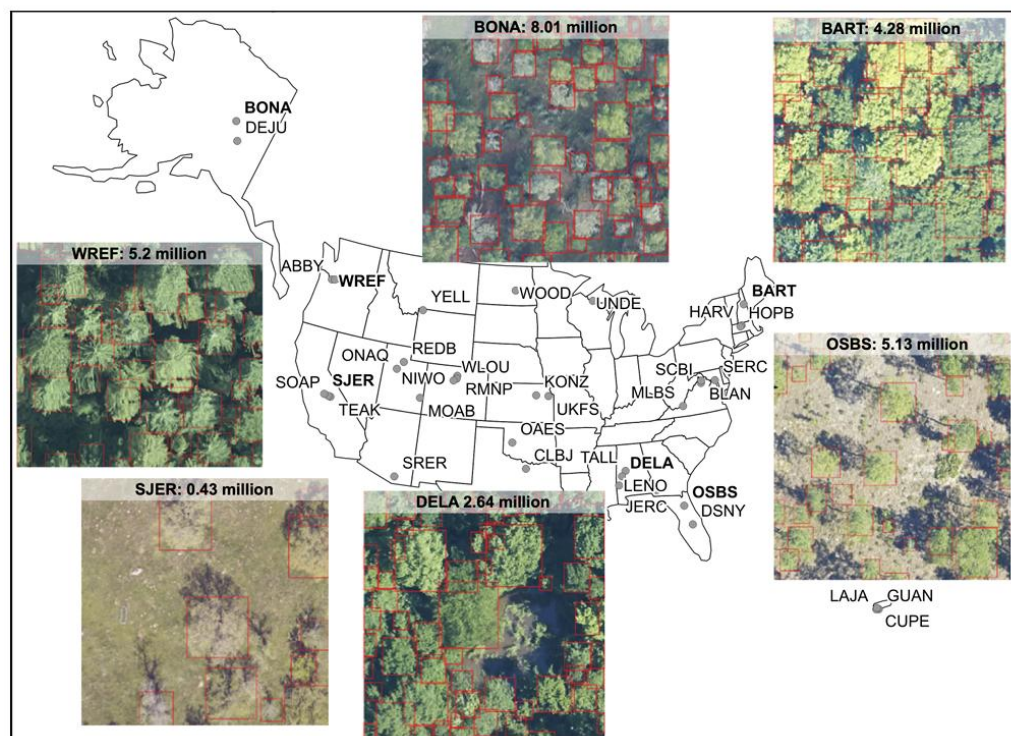
2.13.17 A remote sensing derived data set of 100 million individual tree crowns for the National Ecological Observatory Network

(Dataset Berbasis Penginderaan Jauh yang Memuat 100 Juta Tajuk Pohon Individual untuk National Ecological Observatory Network)

Penelitian ini dilakukan oleh Ben G. Weinstein, Sarah J. Graves, Sergio Marconi. Diterbitkan pada tahun 2021 dalam jurnal *eLife*. Penelitian ini menggunakan algoritma *DeepForest* yaitu model *deep learning* berbasis RetinaNet dan ResNet50 yang dirancang untuk mendeteksi mahkota pohon dari citra *RGB* resolusi tinggi.

Model ini dilatih dalam dua tahap, yaitu dengan data hasil deteksi *LiDAR* sebagai label awal, lalu disempurnakan dengan 10.000 anotasi manual dari citra *NEON*. Hasil deteksi berupa *bounding box* setiap pohon yang mencakup posisi, tinggi (dari *LiDAR*), luas mahkota, dan skor kepercayaan.

Hasil penelitian menunjukkan bahwa *DeepForest* memiliki akurasi deteksi yang tinggi, dengan *recall* 79,1% dan *precision* 72,6% saat diuji pada 6926 pohon dari 21 lokasi *NEON*. Setelah disaring untuk pohon dengan tinggi ≥ 3 meter, *precision* meningkat 7%, dan *recall* hanya turun 0,2%. Validasi dengan data batang pohon lapangan menunjukkan model berhasil mendeteksi 69,4% pohon, terutama pohon kanopi atas. Dataset ini menjadi sumber data yang akurat dan luas untuk analisis ekologi seperti klasifikasi spesies dan pemantauan hutan[36].



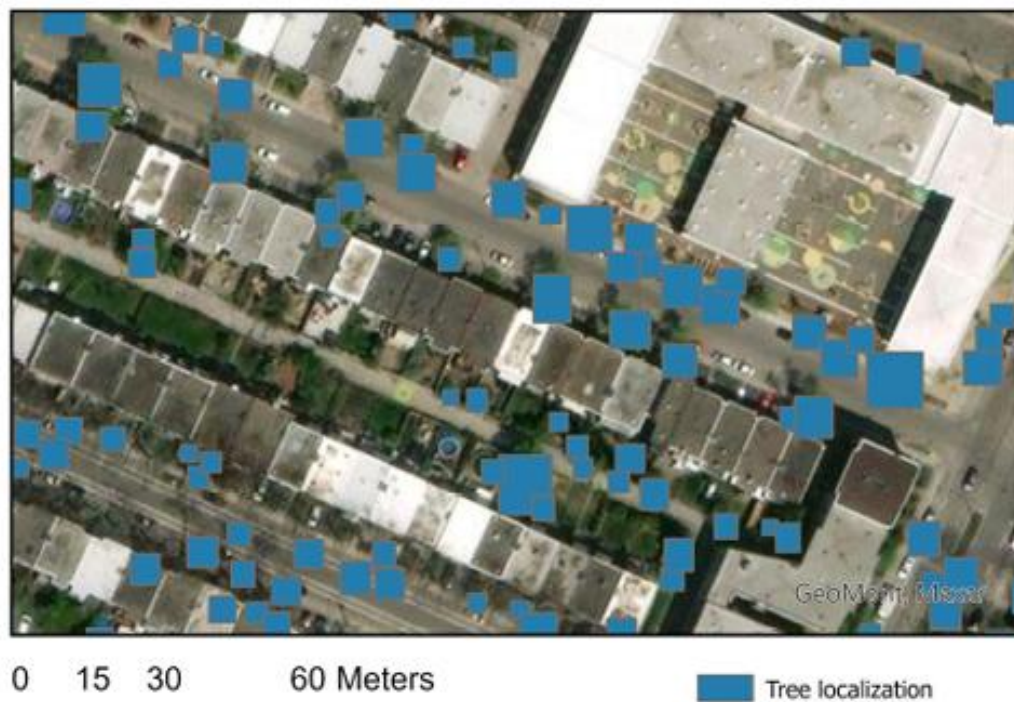
Gambar 2.10 Hasil Penelitian Terkait Sebaran Lokasi Data Kanopi Pohon

2.13.18 *Using the 3-30-300 Indicator to Evaluate Green Space Accessibility and Inequalities: A Case Study of Montreal, Canada.*

(Penggunaan Indikator 3-30-300 untuk Mengevaluasi Aksesibilitas dan Ketimpangan Ruang Hijau: Studi Kasus Montreal, Kanada)

Penelitian ini dilakukan oleh Éric Robitaille dan Cherlie Douyon. Diterbitkan pada tahun 2025 dalam jurnal *Geographies*. Tujuan utama penelitian ini adalah untuk mengevaluasi sebaran tutupan kanopi pohon minimal 30% di lingkungan tempat tinggal dan mengukur keterlibatan minimal tiga pohon dari jendela rumah penduduk. Fokus utamanya adalah menilai ketimpangan akses terhadap elemen vegetasi pohon di kawasan perkotaan Montreal.

Penelitian ini mengandalkan analisis spasial berbasis GIS dengan memanfaatkan data ortofoto resolusi tinggi, data *LiDAR*, serta informasi geospasial terkait bangunan dan ruang hijau. Untuk deteksi pohon penelitian ini mengunggulkan algoritma *DeepForest* berbasis arsitektur *RetinaNet* yang diaplikasikan pada citra ortofoto *RGB* beresolusi 20 cm untuk mendeteksi mahkota pohon secara akurat. Model ini berhasil mengidentifikasi sekitar 1.544.879 pohon di seluruh Pulau Montreal. Hasil penelitian ini deteksi pohon menggunakan model *DeepForest* yang berbasis arsitektur *RetinaNet* menunjukkan nilai *precision* sebesar 0,66 dan *recall* sebesar 0,79[37].



Gambar 2.11 Hasil Penelitian Terkait Pohon Pada Citra Satelit Area Perkotaan

2.13.19 *Identifying Streetscape Features Using VHR Imagery and Deep Learning Applications*

(Identifikasi Fitur *Streetscape* Menggunakan Citra VHR dan Aplikasi Deep Learning)

Penelitian ini dilakukan oleh Deepank Verma, Olaf Mumm, dan Vanessa Miriam Carlow, dan dipublikasikan pada tahun 2021 dalam jurnal *Remote Sensing*. Tujuan penelitian ini adalah untuk mendeteksi dan memetakan pohon serta elemen-elemen jalan perkotaan secara akurat menggunakan citra udara resolusi tinggi.

Penelitian ini menggunakan *DeepForest* untuk mendeteksi pohon dari citra udara resolusi tinggi. Model *DeepForest* dilatih ulang (*finetune*) dengan 2000 gambar pohon dari area kota agar lebih akurat. Selain itu, digunakan model *deep learning* lain, seperti RetinaNet untuk mendeteksi kendaraan dan D-LinkNet untuk memetakan jalan dan trotoar. Semua model diujikan pada citra udara kota dengan

resolusi 0.2 meter, lalu hasilnya dibandingkan dengan data peta yang ada untuk mengukur akurasi.

Hasil penelitian ini menunjukkan bahwa *DeepForest* berhasil mendeteksi pohon di area kota dengan akurasi 87.8% setelah model dilatih ulang. Deteksi kendaraan menggunakan RetinaNet memiliki akurasi 91.2%, dan peta jalan yang dibuat dengan D-LinkNet memiliki akurasi sekitar 74%. Semua hasil deteksi ini cukup akurat dan dapat digunakan untuk membuat peta elemen jalan dan ruang kota secara detail[38].

Tabel 2.14 Hasil Penelitian Terkait Akurasi Pohon Menggunakan DeepForest

S. no.	Features	Classification Method	Model Used	Evaluation Metrics (IoUd/mAPd)	Evaluation Metrics (IoUs/mAPs)
1	Corridor	Segmentation	D-LinkNet34	0.681	0.656
2	Roadway	Segmentation	D-LinkNet34	0.742	0.720
3	Sidewalk	Segmentation	D-LinkNet34	0.670	0.650
4	Bikepath	Segmentation	D-LinkNet34	0.558	0.535
5	On-street Parking	Segmentation	D-LinkNet34	0.551	0.500
6	Trees	Detection	RetinaNet	-	0.878
7	Vegetation	Pixel wise unsupervised classification	Iso-cluster	-	0.812
8	Vehicle	Detection	RetinaNet	0.930	0.912
9	Road Marking	CV Methods +Segmentation	Edge Detection + D-LinkNet34	0.721	0.693

2.13.20 A Benchmark Dataset for Individual Tree Crown Delineation in Co-Registered Airborne RGB, LiDAR and Hyperspectral Imagery from the National Ecological Observation Network

(Dataset Benchmark untuk Delineasi Tajuk Pohon Individual pada Citra Airborne RGB, LiDAR, dan Hiperspektral yang Terkoregistrasi dari National Ecological Observation Network)

Penelitian ini dilakukan oleh Ben G. Weinstein, Sarah J. Graves, Sergio Marconi, Aditya Singh, Alina Zare, Dylan Stewart, Stephanie A. Bohlman, dan Ethan P. White, serta dipublikasikan pada tahun 2020.

Penelitian ini menggunakan *DeepForest* berbasis RetinaNet untuk mendeteksi mahkota pohon dari citra *RGB* resolusi tinggi. Dataset dikumpulkan dari 22 lokasi *NEON*, berisi citra *RGB*, *LiDAR*, dan hiperspektral, dengan lebih dari 6.000 anotasi

mahkota pohon. Evaluasi dilakukan menggunakan *Intersection-over-Union (IoU)*, *precision*, dan *recall*. Data anotasi dibuat dengan aplikasi RectLabel dan evaluasi otomatis menggunakan *NeonTreeEvaluation R package*.

Hasil penelitian menunjukkan bahwa *DeepForest* mencapai *recall* 79.0% dan *precision* 65.9% pada citra anotasi, serta hasil stabil pada data validasi lapangan. *DeepForest* berhasil mendeteksi mahkota pohon dalam berbagai tipe hutan dan kondisi kanopi, menunjukkan kemampuan generalisasi yang baik[39].

Tabel 2.15 Hasil Penelitian Terkait Anotasi Deteksi Pohon

Image-annotated		Field-annotated	Field collected
Crowns		Crowns	Stems
Recall	Precision	Recall	Recall
79.0	65.9	72.2	74.0

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat

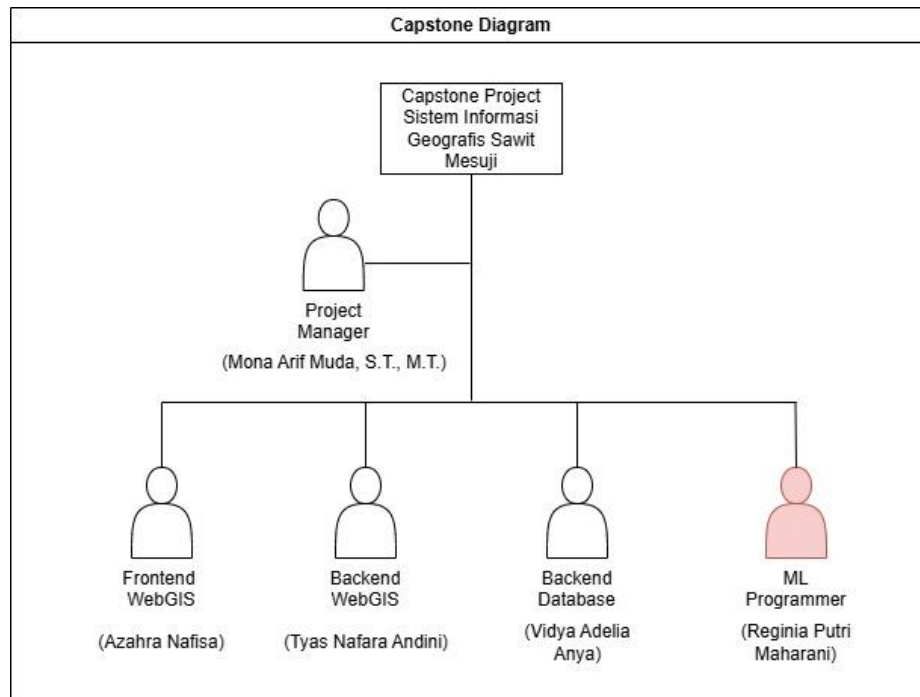
Waktu penelitian berlangsung selama kurang lebih 6 bulan semenjak bulan Mei hingga bulan Oktober 2025. Penelitian ini dilakukan di Gedung H teknik Elektro Universitas Lampung.

Tabel 3.1 Waktu Pelaksanaan

No.	Aktivitas	Mei	Juni	Juli	Agustus	September	Oktober
1.	Pengumpulan Data						
2.	<i>Requirements Planning</i>						
3.	<i>User Design</i>						
4.	<i>Construction</i>						
5.	<i>Cutover</i>						
6.	Pengerjaan Laporan						

3.2 Capstone Project

Adapun untuk mendukung keberhasilan pengembangan *Capstone Project* berjudul "*Sistem Informasi Geografis Sawit Mesuji*," diperlukan pembagian tugas dan tanggung jawab yang jelas dalam tim. Struktur organisasi tim ini dituangkan dalam bentuk diagram agar memudahkan pemahaman peran masing-masing anggota. Adapun struktur tim dalam proyek ini dapat dilihat pada gambar berikut:



Gambar 3.1 *Capstone Project*

Berdasarkan Gambar 3.1 dapat dilihat menunjukkan struktur tim dalam proyek *capstone* yang berjudul "*Sistem Informasi Geografis Sawit Mesuji.*" proyek ini dipimpin oleh seorang *project manager*, yaitu Mona Arif Muda, S.T., M.T., yang bertanggung jawab dalam mengoordinasikan seluruh kegiatan tim. Sehingga yang memberikan arahan adalah *project manager* terdapat empat anggota tim dengan tugas yang telah dibagi sesuai keahlian masing-masing. Terdapat Azahra Nafisa bertugas sebagai *Frontend WebGIS* dalam merancang tampilan antarmuka pengguna agar mudah digunakan. Kemudian terdapat Tyas Nafara Andini sebagai *Backend WebGIS* dan bertanggung jawab pada logika sistem dan pengolahan data di sisi server. Vidya Adelia Anya menangani bagian *Backend Database* yang berfokus pada pengelolaan dan integrasi basis data dalam sistem. Sementara itu, Reginia Putri Maharani berperan sebagai *ML Programmer* yang bertugas mengembangkan dan menerapkan model *deep learning* untuk mendukung analisis spasial pada sistem. Pembagian tugas yang jelas dalam struktur tim ini bertujuan untuk mendukung keberhasilan pengembangan sistem informasi geografis berbasis sawit di wilayah Mesuji.

3.3 Alat Dan Bahan

3.3.1 Alat

Pada penelitian ini, terdapat beberapa alat penelitian yang digunakan untuk mendukung dan menunjang kelancaran setiap proses penelitian yang dilakukan. Adapun alat yang digunakan dalam penelitian ini adalah sebagai berikut

Tabel 3.2 Alat

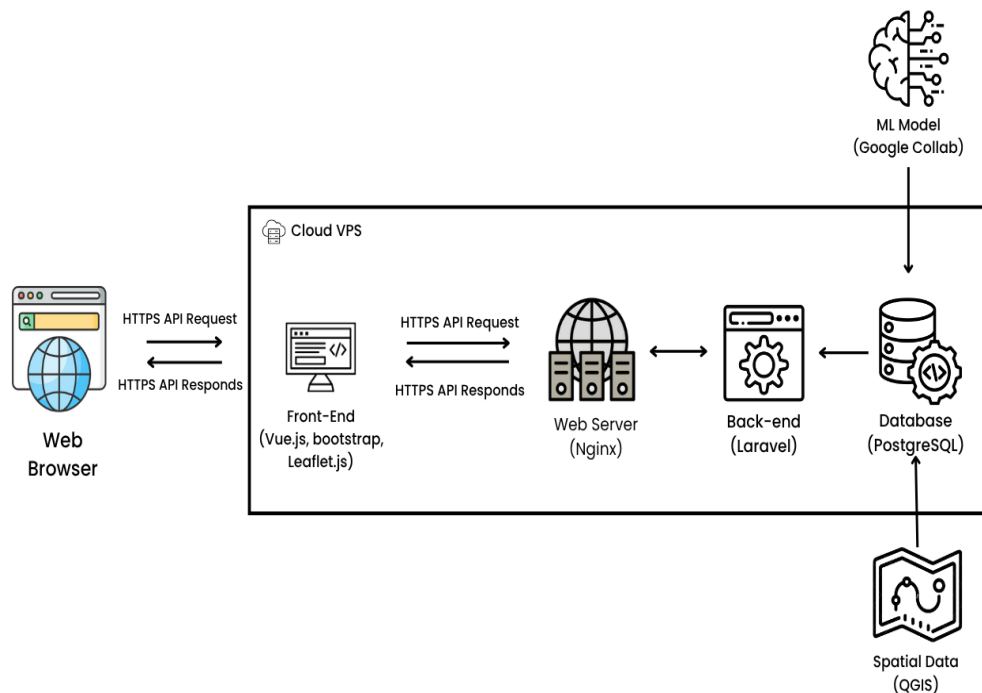
No.	Perangkat	Spesifikasi	Deskripsi
1.	Laptop	AMD Ryzen 3 4300U, RAM 8,00 GB	Perangkat keras yang digunakan dalam proses pemodelan.
2.	<i>Google Colab</i>	CPU: x86_64, RAM Total: 12.67 GB, GPU NVIDIA Tesla T4 , VRAM sebesar 15.36 GB.	Perangkat lunak yang digunakan untuk proses pemodelan.
3.	<i>DeepForest</i>	versi 1.5.2	<i>Library</i> yang digunakan dalam merancang model.
4.	<i>Python</i>	Python 3.12.12	Bahasa pemrograman yang digunakan.
5.	<i>Roboflow</i>	<i>Roboflow Public (Free)</i>	Perangkat lunak yang digunakan dalam merancang dan mempersiapkan dataset.

3.3.2 Bahan Penelitian

Bahan penelitian berupa citra udara kebun sawit dari peta utama berformat TIFF yang dibagi menjadi 7 zona, lalu dipotong menjadi 28 citra JPG. Seluruh citra dianotasi menggunakan bounding box sehingga menghasilkan 78.783 anotasi kanopi pohon sebagai ground truth untuk pelatihan dan evaluasi model *DeepForest* dalam mendeteksi dan menghitung jumlah pohon sawit.

3.4 Arsitektur Perangkat Lunak

Pada tahapan ini terdapat arsitektur perangkat lunak yang dirancang untuk menggambarkan alur kerja sistem secara keseluruhan. Untuk melihat arsitektur perangkat lunak dapat dilihat pada gambar 3.2.



Gambar 3.2 Arsitektur Perangkat Lunak

Berdasarkan gambar 3.2 di atas menggambarkan proses yang dimulai dari Model *DeepForest* yang dijalankan melalui Google Colab. Model ini dilatih menggunakan dataset citra udara kebun sawit sehingga mampu mengenali, mendeteksi kanopi, serta menghitung jumlah pohon sawit secara otomatis beserta koordinat dan ID setiap pohon.

Hasil deteksi disimpan dalam *Database PostgreSQL* yang terintegrasi dengan *PostGIS* untuk pengelolaan data spasial. *Database* ini juga memuat data dari QGIS, seperti batas kebun, jalan, dan area non-produktif, sehingga setiap titik deteksi terhubung langsung dengan lokasi aktual di peta.

Selanjutnya *Back-End system* menggunakan *framework* Laravel yang bertugas mengatur alur data, validasi hasil deteksi, serta menyediakan API bagi antarmuka pengguna. *Web Server* Nginx di Cloud VPS mengelola permintaan data dari pengguna memastikan komunikasi antara *front-end*, *back-end*, dan *database* berjalan cepat dan aman.

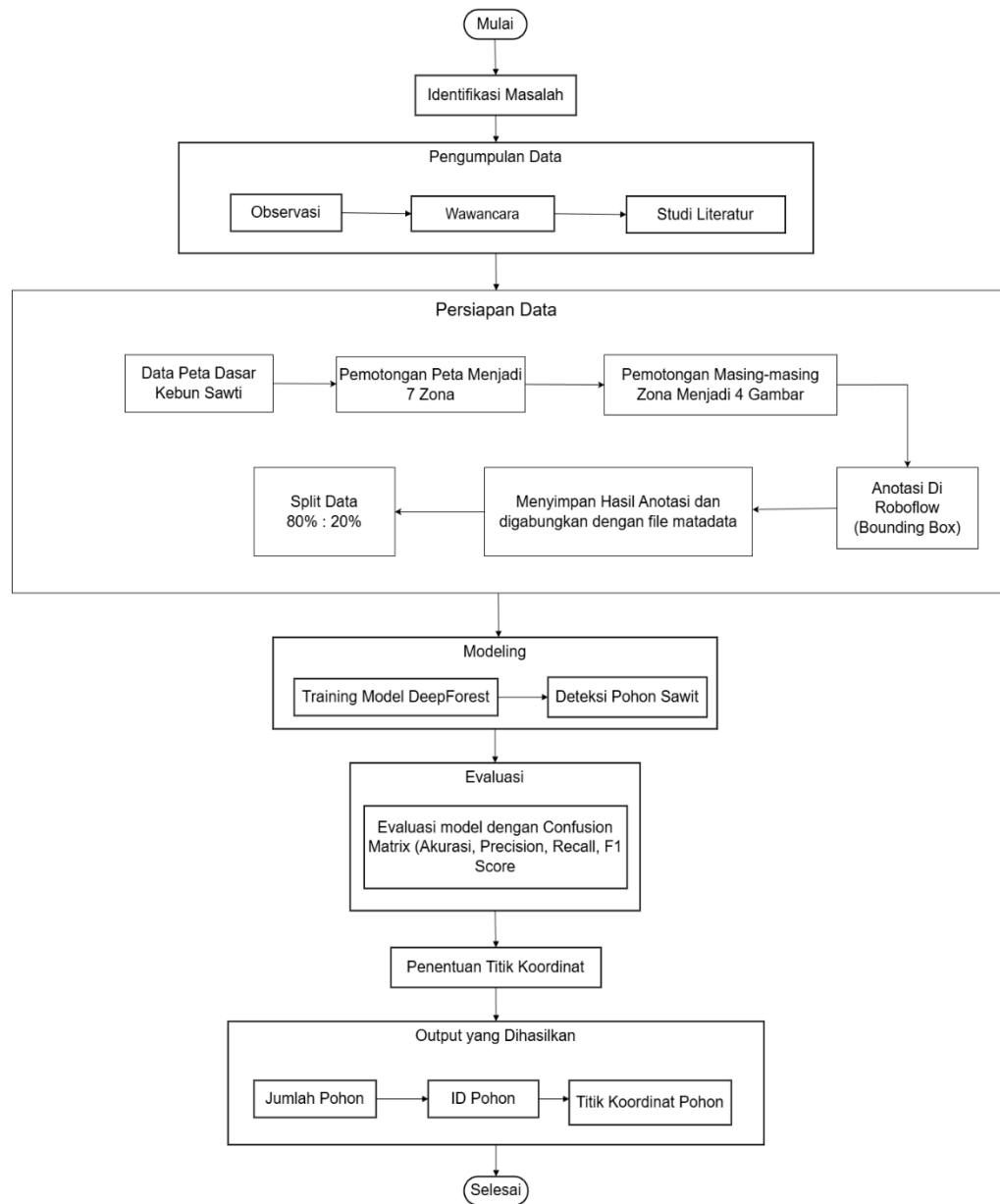
Bagian *Front-End* dibangun dengan Vue.js, Bootstrap, dan Leaflet.js. Komponen ini menampilkan peta interaktif yang berisi titik-titik deteksi pohon sawit, termasuk informasi jumlah pohon per zona, ID, dan koordinatnya. Peta dapat diperbesar, dipindahkan, serta menampilkan detail tambahan sesuai kebutuhan pengguna.

Web Browser menjadi media bagi pengguna untuk mengakses sistem melalui HTTPS API *Request*. Setiap permintaan dari browser diteruskan ke *back-end*, diproses, lalu dikembalikan sebagai API *Response* dalam bentuk data spasial yang divisualisasikan di peta.

Secara keseluruhan, sistem ini mampu menampilkan jumlah dan persebaran pohon sawit secara digital dan *real-time*, tanpa perlu melakukan pengamatan langsung di lapangan. Integrasi antara model *DeepForest*, *database* spasial, dan *WebGIS* membuat sistem ini efektif digunakan untuk pemantauan, pengelolaan, serta pengambilan keputusan terkait kebun sawit.

3.5 Alur Penelitian

Berikut merupakan alur penelitian yang dapat dilihat pada gambar 3.3.



Gambar 3.3 Alur Penelitian

3.5.1 Identifikasi Masalah

Identifikasi masalah dilakukan untuk memahami permasalahan utama dalam pengelolaan dan pemantauan kebun kelapa sawit, khususnya terkait pendataan jumlah pohon dan lokasi masing-masing pohon. Proses pendataan yang dilakukan secara manual membutuhkan waktu yang lama, tenaga yang besar, serta berpotensi menimbulkan kesalahan pencatatan. Selain itu, data yang tersedia umumnya belum terintegrasi dengan sistem spasial sehingga sulit digunakan untuk pemantauan kondisi kebun secara berkala. Oleh karena itu, diperlukan suatu pendekatan berbasis kecerdasan buatan yang mampu mendeteksi kanopi pohon sawit secara otomatis dari citra udara, menghitung jumlah pohon, serta menentukan titik koordinat setiap pohon agar dapat diintegrasikan ke dalam sistem WebGIS.

3.5.2 Pengumpulan Data

Proses pengumpulan data dilakukan melalui beberapa metode yaitu observasi, wawancara dan studi literatur. Dengan menggunakan ketiga metode tersebut diperoleh data yang diperlukan untuk proses deteksi dan estimasi jumlah pohon kelapa sawit.

a. Observasi

Observasi dilakukan oleh *project manager* pada Juni 2023 di area perkebunan pohon kelapa sawit PT. Bina Sawit Makmur (Sampoerna Agro Group) untuk mengamati proses kerja dan aktivitas pengelolaan data kebun sawit. Observasi ini difokuskan pada proses pendataan dan perhitungan pohon kelapa sawit di lapangan. Hasil observasi digunakan sebagai dasar dalam merancang sistem deteksi dan perhitungan pohon kelapa sawit berbasis citra udara menggunakan algoritma *DeepForest* yang dikembangkan dalam penelitian ini.

b. Wawancara

Wawancara dilakukan dengan *project manager* sebagai narasumber yang bertanggung jawab dalam menggali dan menyampaikan kebutuhan sistem dari manager PT. Bina Sawit Makmur (Sampoerna Agro Group), yaitu Bapak Ruli Wandri. Wawancara ini bertujuan untuk memperoleh informasi terkait kebutuhan

sistem, alur pendataan pohon, serta permasalahan dalam proses perhitungan jumlah pohon. Hasil wawancara tersebut digunakan sebagai dasar dalam perancangan sistem deteksi dan perhitungan pohon kelapa sawit berbasis citra udara menggunakan algoritma *DeepForest* yang dikembangkan dalam penelitian ini.

c. Studi literatur

Studi literatur dilakukan dengan mengumpulkan berbagai referensi dan informasi teoritis yang relevan dengan topik penelitian. Tujuan utama dari studi literatur ini adalah untuk mempelajari, memahami dan mencari pendekatan yang dapat digunakan dalam sistem WebGis dengan menggunakan kecerdasan buatan. Adapun studi literatur yang digunakan yaitu kelapa sawit, *artificial intelligence*, *deep learning*, *artificial neural network*, citra, *deepforest*, *confusion matrix*, *pytorch*, *google colab*, *quantum geographic information system (QGIS)*, *roboflow*. Semua literatur ini sudah dijelaskan pada bab 2.

Dengan demikian, studi literatur ini menjadi landasan konseptual dan metodologis dalam pelaksanaan penelitian, serta berperan penting dalam mendukung perancangan, pengembangan, dan evaluasi sistem deteksi dan perhitungan pohon kelapa sawit berbasis citra udara menggunakan algoritma *DeepForest* yang terintegrasi dengan WebGIS.

3.5.3 Persiapan Data

Tahap persiapan data diawali dengan penggunaan data peta dasar kebun kelapa sawit sebagai sumber data utama dalam penelitian ini. Peta dasar tersebut berupa citra udara berformat geospasial (GeoTIFF) yang memuat informasi spasial lengkap, seperti sistem koordinat, batas wilayah kebun, serta resolusi citra yang merepresentasikan kondisi aktual di lapangan. Data ini menjadi fondasi utama karena seluruh proses deteksi pohon sawit, penentuan titik koordinat, dan penghitungan jumlah pohon mengacu pada peta dasar tersebut.

Selanjutnya, peta dasar kebun sawit dilakukan pemotongan (*cropping*) menjadi 7 zona. Pembagian zona ini bertujuan untuk membagi area kebun yang luas menjadi

beberapa bagian yang lebih kecil agar proses pengolahan data menjadi lebih terstruktur dan efisien. Selain itu, pembagian zona memudahkan proses pelatihan dan evaluasi model secara bertahap sehingga performa model dapat dianalisis mulai dari skala kecil hingga skala keseluruhan kebun.

Pada tahap berikutnya setiap zona yang telah terbentuk dipotong kembali menjadi 4 bagian citra. Pemotongan ini dilakukan untuk menyesuaikan ukuran citra dengan kebutuhan proses anotasi dan pelatihan model *DeepForest*. Dengan ukuran citra yang lebih kecil dan seragam, proses pelabelan objek dapat dilakukan dengan lebih teliti serta dapat mengurangi beban komputasi saat proses pelatihan model berlangsung. Meskipun dipotong menjadi beberapa bagian setiap citra tetap mempertahankan kualitas resolusi dan keterkaitan spasial dengan zona asalnya.

Setelah proses pemotongan citra selesai dilakukan anotasi data menggunakan platform Roboflow dengan metode *bounding box*. Pada tahap ini, setiap kanopi pohon kelapa sawit pada citra diberi kotak pembatas (*bounding box*) secara manual untuk menandai posisi dan ukuran objek pohon. Proses anotasi ini sangat penting karena hasilnya digunakan sebagai data latih (*ground truth*) bagi model *DeepForest* agar mampu mempelajari pola visual kanopi pohon sawit dan membedakannya dari objek lain di sekitarnya.

Hasil anotasi *bounding box* yang diperoleh dari Roboflow selanjutnya disimpan dalam bentuk file CSV dan digabungkan dengan file metadata spasial. Metadata ini berisi informasi batas koordinat citra hasil pemotongan, yaitu xmin, ymin, xmax, dan ymax, yang berasal dari peta dasar GeoTIFF hasil pemetaan di QGIS. Nilai koordinat tersebut merepresentasikan batas geografis setiap citra zona dalam sistem koordinat bumi. Penggabungan antara anotasi *bounding box* dan metadata koordinat ini bertujuan untuk menghubungkan koordinat piksel hasil deteksi (xmin, ymin, xmax, ymax dalam satuan piksel) dengan koordinat geografis sebenarnya (*longitude dan latitude*).

Pada pembagian dataset (split data) menjadi dua bagian yaitu 80% data training dan 20% data testing. Data training digunakan untuk melatih model *DeepForest* dalam mengenali kanopi pohon sawit sedangkan data testing digunakan untuk mengevaluasi performa model selama proses pelatihan. Pembagian ini dilakukan secara acak agar model dapat belajar secara optimal serta menghasilkan evaluasi yang objektif.

3.5.4 Modeling

Proses pelatihan model dilakukan menggunakan algoritma *DeepForest* dengan memanfaatkan citra udara kebun sawit yang telah melalui tahap persiapan data dan dianotasi secara manual menggunakan metode bounding box. Setiap anotasi merepresentasikan posisi kanopi pohon sawit dalam citra dan disimpan dalam bentuk koordinat xmin, ymin, xmax, dan ymax, yang menunjukkan batas area objek pohon sawit. Data anotasi ini berperan sebagai referensi utama bagi model dalam proses pembelajaran, sehingga model dapat memahami karakteristik visual kanopi pohon sawit, seperti bentuk, ukuran, dan pola sebarannya pada citra resolusi tinggi. Selama proses pelatihan *DeepForest* mempelajari keterkaitan antara citra input dan anotasi yang diberikan untuk membangun kemampuan deteksi objek secara otomatis. Melalui proses ini, model dilatih agar mampu membedakan kanopi pohon sawit dari elemen lain di lingkungan sekitarnya, seperti tanah, vegetasi non-sawit, atau bayangan. Maka model *DeepForest* terlatih yang mampu melakukan deteksi pohon sawit dalam mendeteksi dan mengenali pohon sawit secara otomatis.

3.5.5 Evaluasi

Evaluasi performa model *DeepForest* dilakukan menggunakan confusion matrix yang menghasilkan empat metrik utama, yaitu akurasi, *precision*, *recall*, dan *F1-score*, untuk menilai kemampuan model dalam mendeteksi kanopi pohon sawit secara menyeluruh. Akurasi menunjukkan proporsi keseluruhan prediksi yang benar dibandingkan dengan seluruh data, sehingga memberikan gambaran umum kinerja model, meskipun pada kasus deteksi objek akurasi tidak selalu menjadi indikator utama karena dominasi area non-objek. *Precision* mengukur tingkat ketepatan model dalam mendeteksi pohon sawit, yaitu seberapa banyak deteksi

yang dihasilkan model benar-benar merupakan pohon sawit, sehingga nilai *precision* yang tinggi menandakan minimnya kesalahan deteksi objek yang bukan pohon sawit. *Recall* menggambarkan kemampuan model dalam menemukan seluruh pohon sawit yang ada pada citra, yaitu seberapa besar proporsi pohon sawit yang berhasil terdeteksi dari total pohon sebenarnya, sehingga nilai *recall* yang tinggi menunjukkan bahwa sedikit pohon yang terlewat. *F1-score* merupakan nilai rata-rata harmonis antara *precision* dan *recall* yang digunakan untuk menilai keseimbangan kinerja model, terutama ketika diperlukan deteksi yang tidak hanya tepat tetapi juga menyeluruh. Dengan menggabungkan keempat metrik ini, evaluasi model menjadi lebih terstruktur dan spesifik dalam menggambarkan kualitas deteksi pohon sawit yang dihasilkan oleh model *DeepForest*.

3.5.6 Penentuan Titik Koordinat

Penentuan titik koordinat dilakukan untuk mengubah hasil deteksi pohon sawit dari model *DeepForest* yang masih berbasis koordinat piksel menjadi koordinat geografis yang dapat digunakan pada peta. Setiap hasil deteksi berupa bounding box memiliki nilai *xmin*, *ymin*, *xmax*, dan *ymax* yang menunjukkan batas objek kanopi pohon sawit pada citra. Dari nilai tersebut dihitung titik tengah (*centroid*) sebagai representasi posisi pohon kemudian titik ini dikonversi ke koordinat geografis dengan memanfaatkan metadata spasial setiap zona yaitu batas koordinat *xmin*, *ymin*, *xmax*, dan *ymax* yang diperoleh dari peta GeoTIFF hasil pemetaan QGIS dengan proses konversi ini menghasilkan nilai longitude dan latitude. Seluruh hasil koordinat kemudian disimpan dalam format CSV yang berisi informasi ID pohon, longitude, latitude, dan zona, sehingga data tersebut mudah dikelola, dianalisis, serta dapat langsung diintegrasikan ke dalam database dan ditampilkan pada sistem WebGIS sebagai titik lokasi pohon sawit.

3.5.7 Output Yang di Hasilkan

Output yang dihasilkan dari proses pengolahan data menggunakan model *DeepForest* berupa informasi spasial yang menggambarkan hasil deteksi kanopi pohon pada area kebun. Berdasarkan hasil pemrosesan citra beresolusi tinggi, sistem mampu menghasilkan tiga jenis output utama, yaitu jumlah pohon, identitas

pohon, dan titik koordinat pohon. Jumlah pohon merupakan hasil perhitungan total kanopi pohon yang berhasil terdeteksi oleh model pada area kebun yang dianalisis. Setiap pohon yang terdeteksi selanjutnya diberikan identitas unik berupa ID pohon, yang berfungsi sebagai penanda untuk membedakan satu pohon dengan pohon lainnya. Selain itu, sistem menghasilkan titik koordinat pohon yang merepresentasikan posisi geografis setiap pohon, sehingga dapat diintegrasikan dan divisualisasikan pada WebGIS.

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian mengenai deteksi dan estimasi jumlah pohon sawit menggunakan model *DeepForest* diperoleh beberapa kesimpulan sebagai berikut:

1. Dengan membangun sistem kecerdasan buatan untuk mendeteksi kanopi pohon kelapa sawit menggunakan citra udara dan algoritma *DeepForest*. Sistem bekerja dengan membagi area kebun ke dalam beberapa zona, membuat dataset citra beserta anotasi *bounding box* serta melatih model untuk mendeteksi setiap pohon secara otomatis. Hasil deteksi digunakan untuk menghitung jumlah pohon dan menentukan koordinat spasial masing-masing pohon, yang selanjutnya diintegrasikan ke dalam WebGIS.
2. Penerapan algoritma *DeepForest* dalam mendeteksi pohon sawit pada area perkebunan menunjukkan kinerja yang baik. Model *DeepForest* yang dibangun menggunakan arsitektur RetinaNet dengan *backbone ResNet-50* dan memiliki sekitar 25 juta parameter mampu mengenali kanopi pohon sawit melalui citra udara yang telah diproses. Hasil deteksi tersebut dapat digunakan untuk menghitung jumlah pohon, menentukan ID masing-masing pohon, serta memperoleh titik koordinat secara presisi. Penerapan ini tidak hanya mempercepat proses identifikasi dan pemetaan pohon sawit, tetapi juga menjadi dasar penting dalam pengelolaan data spasial perkebunan ketika diintegrasikan dengan sistem WebGIS.
3. Algoritma *DeepForest* menunjukkan kemampuan yang baik dalam mengestimasi jumlah pohon sawit berdasarkan hasil evaluasi model. Nilai *precision* sebesar 0.9357, *recall* sebesar 0.9433, dan *f1-score* sebesar 0.9395 menggambarkan bahwa model memiliki tingkat ketepatan, kelengkapan, dan keseimbangan performa yang tinggi. Hasil pengujian ini membuktikan bahwa sebagian besar pohon yang terdeteksi benar-benar sesuai dengan objek

sebenarnya. Dibandingkan dengan metode manual atau konvensional yang memerlukan waktu dan tenaga besar serta penggunaan *DeepForest* dalam proses pemetaan serta analisis jumlah pohon sawit pada area perkebunan berskala luas.

5.2 Saran

Untuk pengembangan lebih lanjut, berikut adalah beberapa saran yang dapat dipertimbangkan:

1. Untuk menjaga akurasi data diperlukan pemantauan secara berkala yaitu dengan melakukan pembaruan citra dan hasil deteksi secara rutin sehingga perubahan jumlah serta kondisi pohon sawit dapat terpantau dengan baik dan berkelanjutan.
2. Penelitian selanjutnya dapat menambahkan analisis persentase area tutupan kanopi untuk mengetahui seberapa luas area kebun yang sudah tertanami pohon sawit secara efektif. Hasil analisis ini juga bisa membantu melihat potensi area yang masih kosong untuk dimanfaatkan lebih lanjut.

DAFTAR PUSTAKA

- [1] A. A. Nugroho, R. R. Yusuf, and N. S. Rahayu, *Eksklusi Petani pada Industri Kelapa Sawit: Dalih Kemitraan pada Perkebunan Inti-Plasma Skema Kredit Koperasi Primer Anggota*. Eds. Jakarta, Indonesia: IGPA Press, 2024, pp. 153–170.
- [2] D. Persson, *Tree Crown Detection Using Machine Learning: A study on using the DeepForest deep-learning model for tree crown detection in drone-captured aerial footage*, 2021.
- [3] J. MAPPI and H. T. F. GP, *Penilaian Perkebunan Kelapa Sawit*. Jakarta, Indonesia: Merdeka Kreasi Group, 2022.
- [4] A. Nugroho, *Buku Teknologi Agroindustri Kelapa Sawit*, 1st ed. Banjarbaru, Indonesia: Lambung Mangkurat University Press, 2019, ISBN: 978-602-6483-97-3.
- [5] B. Coppin, *Artificial Intelligence Illuminated*. Jones & Bartlett Learning, 2004.
- [6] H. Jaya, Sabran, and M. Idris, *Kecerdasan Buatan*. Makassar, Indonesia: Fakultas MIPA Universitas Negeri Makassar, 2018.
- [7] D. J. Hemanth and V. V. Estrela, *Deep Learning for Image Processing Applications*. IOS Press, 2017.
- [8] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*. MIT Press, 1995.
- [9] W. Maharani, “Klasifikasi Data Menggunakan Jst Backpropagation Momentum Dengan Adaptive Learning Rate,” 2009.
- [10] F. Millstein, *Convolutional Neural Networks In Python: Beginner’s Guide To Convolutional Neural Networks In Python*. Frank Millstein, 2020.
- [11] S. Rahman, M. Ramli, F. Arnia, R. Muharar, M. Zen, and M. Ikhwan, *Convolutional Neural Networks Untuk Visi Komputer Jaringan Saraf*

Konvolusional untuk Visi Komputer (Arsitektur Baru, Transfer Learning, Fine Tuning, dan Pruning). Deepublish, 2021.

- [12] J.-L. Starck, F. Murtagh, and A. Bijaoui, *Image Processing and Data Analysis: The Multiscale Approach*. Cambridge University Press, 1998.
- [13] P. N. Andono, T. Sutojo, and Muljono, *Pengolahan Citra Digital*. Penerbit Andi, 2018.
- [14] Z.-H. Zhou and J. Feng, *Deep Forest: Towards an Alternative to Deep Neural Networks*, in Proc. 26th International Joint Conference on Artificial Intelligence (IJCAI), Melbourne, Australia, 2017, pp. 3553–3559.
- [15] N. A. D. Istanto and B. S. W. Poetro, “Implementasi Algoritma Faster R-Cnn Dalam Deteksi Aktivitas Merokok Di Lingkungan Kampus,” vol. 3, no. 1, 2025.
- [16] A. Muzakir, K. Adi, dan R. Kusumaningrum, *Penerapan Konsep Machine Learning dan Deep Learning: Pendekatan Ekspansi Semantik untuk Klasifikasi Ujaran Kebencian*, Cet. 1. Semarang, Indonesia: UNDIP Press, Feb. 2024, ISBN: 978-623-417-282-9.
- [17] “Google Colab.” Accessed: May 16, 2025. [Online]. Available: <https://colab.research.google.com/>
- [18] E. Astutik, *Pemanfaatan Aplikasi Qgis 3.10 untuk Pemetaan Pola Penyakit dan Risiko*. Airlangga University Press, 2022.
- [19] A. D. Lestari, *Aplikasi Deteksi Hama Dan Penyakit Pada Buah Kakao Menggunakan Yolo V8 Berbasis Website*. 2024.
- [20] D. Wijaya et al, *Tree Counting with Deep Forest Algorithm for Kulon Progo District in Yogyakarta, Indonesia Using Pleiades Satellite Imagery*, in 2024 Arab ICT Conference (AICTC), Manama, Bahrain: IEEE, Feb. 2024, pp. 227–231. doi: 10.1109/AICTC58357.2024.10735046.
- [21] V. Dakov and D. Petrova-Antonova, “Urban Tree Detection from Remote Sensing Data based on DeepForest Model,” *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. X-4/W4-2024, pp. 35–41, May 2024, doi: 10.5194/isprs-annals-X-4-W4-2024-35-2024.
- [22] P. Sivanandam and A. Lucieer, “Tree Detection and Species Classification in a Mixed Species Forest Using Unoccupied Aircraft System (UAS) RGB and

- Multispectral Imagery,” *Remote Sensing*, vol. 14, no. 19, p. 4963, Oct. 2022, doi: 10.3390/rs14194963.
- [23] H. İ. Şenol and A. Y. Yiğit, *Decoding Nature's Patterns: An Innovative Approach to Tree Detection Using Deep Learning and High-Resolution Aerial Imagery*, Turkey, " Turkish Journal of Geographic Information Systems, vol. 5, no. 1, pp. 52–59, 2023, doi: 10.56130/tucbis.1307926.
- [24] B. G. Weinstein, S. Marconi, M. Aubry-Kientz, G. Vincent, H. Senyondo, and E. White, *DeepForest: A Python Package for RGB Deep Learning Tree Crown Delineation*, 2020, doi: 10.1101/2020.07.07.191551.
- [25] I. Harmon et al., *Injecting Domain Knowledge Into Deep Neural Networks for Tree Crown Delineation*, TechRxiv, 2022. DOI: 10.36227/techrxiv.20323698.v1.
- [26] A. H. Hanafi et al, Penerapan Deepforest untuk Deteksi dan Estimasi Jumlah Pohon Bakau di Petengoran,
- [27] H. Jemaa, W. Bouachir, B. Leblon, and N. Bouguila, “Computer Vision System For Detecting Orchard Trees From Uav Images,” *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. XLIII-B4-2022, pp. 661–668, Jun. 2022, doi: 10.5194/isprs-archives-XLIII-B4-2022-661-2022.
- [28] S. Jemaa, M. Jendoubi, and S. Ben Jabra, “Computer Vision System for Detecting Orchard Trees,” *Procedia Computer Science*, vol. 198, pp. 502–507, 2022. DOI: 10.1016/j.procs.2021.12.254.
- [29] J. Godínez-Garrido, D. A. Martínez-Álvarez, J. D. Dávila-Aguilar, and J. E. Vargas-Larreta, *Estimation of Damaged Regions by the Bark Beetle in Pine Forests Using Deep Learning and UAV Imagery*, *Remote Sensing*, vol. 16, no. 3, p. 557, 2024. DOI: 10.3390/rs16030557.
- [30] J.-L. Starck, F. Murtagh, and A. Bijaoui, *Image Processing and Data Analysis: The Multiscale Approach*. Cambridge, U.K.: Cambridge University Press, 1998.
- [31] S. Jemaa, M. Jendoubi, and S. Ben Jabra, *UAV-Based Computer Vision System for Orchard Apple Tree Detection*, *Procedia Computer Science*, vol. 217, pp. 2075–2082, 2023. DOI: 10.1016/j.procs.2022.12.259.

- [32] B. G. Weinstein *et al.*, Individual canopy tree species maps for the National Ecological Observatory Network,” *PLoS Biol.*, vol. 22, no. 7, p. e3002700, 2024. DOI: 10.1371/journal.pbio.3002700.
- [33] I. Roemer, M. Schieck, N. Harnau, and B. Franczyk, “Monitoring the health of agricultural ecosystems from remote sensing data using semi-supervised neural networks,” *Procedia Computer Science*, vol. 246, pp. 1299–1308, 2024, doi: 10.1016/j.procs.2024.09.559.
- [34] M. A. Batubara, S. Alam, K. Nisa, R. W. Utami, H. Fitriawan, and A. Ulvan, “Estimating the Number of Trees in Margasari Mangrove Forests of Lampung Through Aerial Images Using Adaptive Thresholding and Contour Extraction Methods;,” presented at the International Conference on Sustainable Biomass (ICSB 2019), Bandar Lampung, Indonesia, 2021. doi: 10.2991/aer.k.210603.022.
- [35] B. G. Weinstein, S. J. Graves, S. Marconi, A. Singh, A. Zare, D. Stewart, S. A. Bohlman, and E. P. White, “A benchmark dataset for canopy crown detection and delineation in co-registered airborne RGB, LiDAR and hyperspectral imagery from the National Ecological Observation Network,” *PLOS Computational Biology*, vol. 17, no. 7, pp. 1–18, Jul. 2021, doi: 10.1371/journal.pcbi.1009180.
- [36] B. G. Weinstein *et al.*, “A remote sensing derived data set of 100 million individual tree crowns for the National Ecological Observatory Network,” *eLife*, vol. 10, p. e62922, Feb. 2021, doi: 10.7554/eLife.62922.
- [37] É. Robitaille and C. Douyon, “Using the 3-30-300 Indicator to Evaluate Green Space Accessibility and Inequalities: A Case Study of Montreal, Canada,” *Geographies*, vol. 5, no. 1, p. 6, Feb. 2025, doi: 10.3390/geographies5010006.
- [38] D. Verma, O. Mumm, and V. M. Carlow, “Identifying streetscape features using VHR imagery and deep learning applications,” *Remote Sens.*, vol. 13, no. 17, p. 3363, 2021. DOI: 10.3390/rs13173363.
- [39] B. G. Weinstein *et al.*, *A benchmark dataset for individual tree crown delineation in co-registered airborne RGB, LiDAR and hyperspectral imagery from the National Ecological Observation Network*, Remote Sensing of Environment, vol. 247, p. 111979, 2020, doi: 10.1016/j.rse.2020.111979.