

**IMPLEMENTASI METODE FASTER R-CNN DAN XGBOOST UNTUK
KUANTIFIKASI OTOMATIS POLONG DAN KLASIFIKASI
VARIETAS TERPILIH KEDELAI**

SKRIPSI

STEVEN MANURUNG

211402088



**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2025**

**IMPLEMENTASI METODE FASTER R-CNN DAN XGBOOST UNTUK
KUANTIFIKASI OTOMATIS POLONG DAN KLASIFIKASI
VARIETAS TERPILIH KEDELAI**

SKRIPSI

Diajukan untuk melengkapi dan memenuhi syarat memperoleh ijazah
Sarjana Teknologi Informasi

STEVEN MANURUNG

211402088



**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2025**

UCAPAN TERIMA KASIH

Puji syukur kepada Tuhan Yang Maha Esa, yang telah memberikan rahmat dan karunia-NYA yang tidak terhitung sehingga penulis dapat menyelesaikan skripsi ini dengan judul “Implementasi Metode Faster R-CNN dan XGBoost Untuk Kuantifikasi Otomatis Polong dan Klasifikasi Varietas Terpilih Kedelai”. Skripsi ini disusun sebagai salah satu syarat memperoleh gelar Sarjana Komputer pada Program Studi S1 Teknologi Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara. Meskipun mengalami beberapa kendala selama proses penulisan, penulis berhasil mengatasi tantangan tersebut dengan sukses berkat dukungan dan panduan dari berbagai pihak.

Penulis menyadari bahwa skripsi ini tidak akan terselesaikan tanpa dukungan dan bantuan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, izinkan penulis menyampaikan ucapan terima kasih kepada:

1. Tuhan Yang Maha Esa yang telah memberikan segala kemudahan, kelancaran dan rezeki-Nya sehingga penulis mampu menyelesaikan penulisan skripsi ini.
2. Keluarga inti penulis yaitu, ayah Robert Kristian Alpendo Manurung, ibu Rumeri Hutarabat, abang Reynald Daniel Nicholas Manurung, Mathew Jhonathan Manurung, dan adik Stevany Alfenda Manurung yang selalu memberikan semangat pada penulis.
3. Ibu Rossy Nurhasanah S.Kom., M.Kom, selaku Dosen Pembimbing I, yang dengan tulus hati telah meluangkan waktu, memberikan bimbingan yang mendalam, serta curahan pemikiran dan kritik konstruktif.
4. Bapak Prof. Dr. Romi Fadillah Rahmat B.Comp.Sc., M.Sc. sebagai Dosen Pembimbing II, yang juga telah banyak memberikan kontribusi pemikiran, saran, dan kritik konstruktif selama proses penulisan skripsi.
5. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi.
6. Bapak Dedy Arisandi S.T., M.Kom. selaku Ketua Program Studi dan Bapak Ivan Jaya, S.Si., M.Kom selaku Sekretaris Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.

7. Ibu Prof. Dr. Diana Sofia Hanafiah SP., MP. sebagai Dosen di Pertanian Universitas Sumatera Utara, yang telah banyak memberikan kontribusi pemikiran, saran, kritik selama melakukan penelitian.
8. Bapak dan Ibu Dosen di Program Studi Teknologi Informasi, Universitas Sumatera Utara, yang telah memberikan pengetahuan dan pembelajaran kepada penulis selama masa perkuliahan.
9. Seluruh staf dan pegawai Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara, yang telah memberikan bantuan dan pelayanan dalam mengurus berbagai keperluan administrasi selama masa perkuliahan.
10. Teman-teman penulis M. Azis Saputra, M. Ariyo Syahraza selalu menemani, menyemangati, dan membantu penulis selama perkuliahan hingga selesai melakukan penulisan tugas akhir penulis.
11. Teman-teman penulis Usman Ismail Pane, Yunus Tio Buntoro, Josua Peter Corda, dan Gali Rakasiwi yang membantu penulis dalam pengambilan data hingga selesai.
12. Teman-teman Raffael Tariparantau Manurung dan Handrianta Ginting yang selalu mendorong dan menemani penulis selama masa perkuliahan.
13. Seluruh teman Angkatan 2021 yang tidak dapat disebut satu per satu.
14. Semua pihak yang secara langsung maupun tidak langsung turut membantu penyelesaian tugas akhir ini dan tidak dapat dituliskan satu per satu.

Penulis menyadari bahwa masih terdapat kekurangan dalam penulisan skripsi ini, oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan untuk penyempurnaan skripsi ini.

Medan, Desember 2025
Penulis,

Steven Manurung
211402088

ABSTRAK

Kedelai (*Glycine max* (L.) Merr) merupakan komoditas strategis bagi ketahanan pangan nasional Indonesia yang merupakan sumber protein nabati utama. Namun, produksi domestik belum mampu memenuhi kebutuhan karena rendahnya produktivitas dan keterbatasan varietas unggul. Penelitian ini mengusulkan sistem deteksi, kuantifikasi polong, dan klasifikasi tanaman terpilih berbasis *computer vision* dan *machine learning* dengan menggabungkan metode *Faster Region-Convolutional Neural Network* (Faster R-CNN) dengan *backbone* ResNet50 dan *eXtreme Gradient Boosting* (XGBoost). Sistem ini dirancang untuk melakukan perhitungan pada jumlah polong dan seleksi tanaman terpilih dan tidak terpilih. Total data yang digunakan untuk deteksi dan kuantifikasi polong sebanyak 715 data, dengan data *training* sebanyak 561 data, data *validation* sebanyak 76, dan data *testing* sebanyak 76 data serta total data yang digunakan untuk klasifikasi tanaman terpilih sebanyak 553 data, dengan data training sebanyak 442 data dan data testing sebanyak 111 data. Model deteksi dan kuantifikasi polong kedelai mencapai *mean Average Precision* (mAP) 0,906 dan *Intersection over Union* (IoU) 0,9588 dengan konfigurasi *hyperparameter* optimal diperoleh pada batch size 4, *learning rate* 0,005, dan *weight decay* 0,0001. Sementara itu, model klasifikasi tanaman terpilih mencapai *accuracy training* 0,95, *accuracy testing* 0,99, *precision* 1.00 dan, serta *recall* 0.98 dan 1.00 (*F1-Score* = 0.99) dengan konfigurasi *hyperparameter* optimal diperoleh pada *learning_rate* = 0,01, *n_estimators* = 200, *max_depth* = 3.

Kata kunci : Kedelai, Citra Polong, Faster R-CNN, ResNet50, XGBoost

*IMPLEMENTATION OF THE FASTER R-CNN AND XGBOOST METHODS FOR
AUTOMATIC POD QUANTIFICATION AND CLASSIFICATION OF
SELECTED SOYBEAN VARIETIES*

ABSTRACT

Soybean (Glycine max (L.) Merr) is a strategic commodity for Indonesia's national food security, which is the main source of vegetable protein. However, domestic production has not been able to meet the needs due to low productivity and limited superior varieties. This study proposes a detection system, pod quantification, and classification of selected plants based on computer vision and machine learning by combining the Faster Region-Convolutional Neural Network (Faster R-CNN) method with the ResNet50 backbone and eXtreme Gradient Boosting (XGBoost). This system is designed to calculate the number of pods and select selected and unselected plants. The total data used for pod detection and quantification is 715 data, with 561 training data, 76 validation data, and 76 testing data, and the total data used for classification of selected plants is 553 data, with 442 training data and 111 testing data. The soybean pod detection and quantification model achieved a mean Average Precision (mAP) of 0.906 and an Intersection over Union (IoU) of 0.9588 with optimal hyperparameter configurations obtained at batch size 4, learning rate 0.005, and weight decay 0.0001. Meanwhile, the selected plant classification model achieved a training accuracy of 0.95, a testing accuracy of 0.99, a precision of 1.00 and a recall of 0.98 and 1.00 (F1-Score = 0.99) with the optimal hyperparameter configuration obtained at learning_rate = 0.01, n_estimators = 200, and max_depth = 3.

Keywords : *Soybean, Pod Image, Faster R-CNN, ResNet50, XGBoost*

DAFTAR ISI

ABSTRAK	iii
ABSTRACT	iv
DAFTAR ISI.....	v
DAFTAR TABEL	viii
DAFTAR GAMBAR	ix
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	5
1.6 Metodologi Penelitian.....	5
1.7 Sistematika Penulisan	6
BAB II LANDASAN TEORITIS.....	8
2.1 Kedelai (Glycine max (L.) Merr).....	8
2.2 Varietas Unggul Kedelai	9
2.2.1 Anjasmoro	9
2.2.2 Dega	9
2.2.3 Grobogan.....	9
2.3 Citra Polong Kedelai.....	10
2.4 Morfologi Tanaman Kedelai	11
2.5 Pengolahan Citra.....	12
2.6 Labeling	12
2.7 Resizing.....	13
2.8 Augmentation.....	13
2.9 Convolutional Neural Network.....	14
2.10 ResNet 50.....	15
2.11 Fast R-CNN.....	17
2.12 Faster R-CNN	17
2.13 XGBoost	18

2.14 Confusion Matrix	18
2.15 Penelitian Terdahulu	19
2.16 Perbedaan Penelitian	27
BAB III ANALISIS DAN PERANCANGAN SISTEM.....	29
3.1 Data yang digunakan.....	29
3.1.1 Data Citra Polong Kedelai	29
3.1.2 Data Morfologi Kedelai	31
3.2 Analisis Sistem.....	33
3.3 Data Acquisition.....	35
3.3.1 Faster R-CNN	35
3.3.2 XGBoost	35
3.4 Image Pre-preprocessing.....	35
3.5 Faster R-CNN	39
3.5.1 Hyperparameter Tuning	41
3.5.2 Training Process.....	43
3.6 XGBoost	44
3.6.1 Hyperparameter Tuning	45
3.6.2 Grid Search	46
3.6.3 Training Process.....	46
3.7 Model Evaluation.....	47
3.8 Best Model Selection & Deployment	48
3.9 Output	48
3.10 Perancangan Antarmuka Sistem	48
3.10.1 Rancangan Tampilan Halaman Beranda	48
3.10.2 Rancangan Tampilan Halaman Hitung Polong.....	49
3.10.3 Rancangan Halaman Hasil Hitung Polong	49
3.10.4 Rancangan Tampilan Halaman Klasifikasi	50
3.10.5 Rancangan Tampilan Halaman About.....	51
BAB IV IMPLEMENTASI DAN PENGUJIAN	52
4.1 Implementasi Sistem.....	52
4.1.1 Perangkat Keras	52
4.1.2 Perangkat Lunak	52
4.2 Implementasi Data	53

4.3 Training Pada Metode Faster R-CNN.....	53
4.3.1 Evaluasi pada citra polong kedelai	53
4.4 Training Pada Metode XGBoost.....	58
4.4.1 Evaluasi pada parameter morfologi kedelai.....	59
4.5 Implementasi Desain Antarmuka.....	60
4.5.1 Tampilan Halaman Depan.....	60
4.5.2 Tampilan Pengenalan Website	60
4.5.3 Tampilan Halaman Hitung Polong.....	61
4.5.4 Tampilan Hasil Hitung Polong.....	62
4.5.5 Tampilan Halaman Klasifikasi.....	62
4.6 Prosedur Operasional	63
4.7 Pengujian Sistem.....	64
4.7.1 Pengujian Model Deteksi Objek Polong Kedelai	64
4.7.2 Validasi dengan Pakar	69
BAB V KESIMPULAN DAN SARAN	70
5.1 Kesimpulan	70
5.2 Saran	67
DAFTAR PUSTAKA	68

DAFTAR TABEL

Tabel 2.1 Rincian Layer Arsitektur ResNet50	16
Tabel 2.2 Penelitian Terdahulu.....	23
Tabel 3.1 Rincian Jumlah <i>Instance</i> tiap Biji pada Polong	30
Tabel 3.2 Rincian Jumlah Data Citra Polong per-tipe Pengambilan Gambar.....	31
Tabel 3.3 Rincian Jumlah Data	31
Tabel 3.4 Skenario <i>Training</i>	43
Tabel 3.5 <i>Hyperparameter</i> XGBoost.....	46
Tabel 4.1 Hasil Evaluasi Model Polong Kedelai menggunakan <i>Full Layers</i>	53
Tabel 4.2 Perbandingan Hasil Evaluasi <i>Fine-Tuning</i> dengan Parameter Terbaik.....	55
Tabel 4.3 Hasil Model dengan <i>Hyperparameter</i> Terbaik	59
Tabel 4.4 Hasil Pengujian Model Deteksi Polong Kedelai.....	64

DAFTAR GAMBAR

Gambar 2.1 Citra Polong Kedelai	10
Gambar 2.2 Arsitektur CNN (Sumber: Developersbreach.com)	14
Gambar 2.3 Faster R-CNN	17
Gambar 2.4 XGBoost (Sumber: Nvidia.com).....	18
Gambar 2.5 <i>Confusion Matrix</i>	18
Gambar 3.1 (a) Polong pada tanaman, (b) Polong yang sudah dipetik, (c) Polong per biji (1, 2, dan 3).....	30
Gambar 3.2 (a) Polong pada Tanaman, (b) Polong yang sudah dipetik.....	30
Gambar 3.3 Data Kuantitatif Morfologi Kedelai	31
Gambar 3.4 <i>Correlation Matriks</i>	32
Gambar 3.5 Arsitektur Umum Penelitian.....	34
Gambar 3.6 Proses Anotasi Pada Citra	35
Gambar 3.7 Struktur File Json dari Hasil Anotasi	36
Gambar 3.8 Arsitektur Faster R-CNN	39
Gambar 3.9 Arsitektur ResNet-50.....	39
Gambar 3.10 Arsitektur <i>Region Proposal Network</i> (RPN).....	40
Gambar 3.11 <i>Bounding Box</i>	41
Gambar 3.12 <i>Install Requirement</i>	44
Gambar 3.13 <i>Model Installation</i>	44
Gambar 3. 14 <i>Import Library</i>	44
Gambar 3.15 Arsitektur XGBoost.....	45
Gambar 3.16 <i>Install Requirement</i>	47
Gambar 3.17 <i>Import Library</i>	47
Gambar 3.18 Rancangan Halaman Beranda	49
Gambar 3.19 Rancangan Halaman Hitung Polong	49
Gambar 3.20 Rancangan Halaman Hasil Hitung Polong.....	50
Gambar 3.21 Rancangan Halaman Klasifikasi	50
Gambar 3.22 Rancangan Halaman <i>About</i>	51
Gambar 4.1 Grafik Matrik <i>Fully Layer</i>	54
Gambar 4.2 Grafik Metrik <i>Fully Layer</i> dengan <i>Dropout</i>	56

Gambar 4.3 Grafik Metrik <i>Head Layer</i>	56
Gambar 4.4 Deteksi Polong pada Tanaman	57
Gambar 4.5 Deteksi Polong setelah dipetik	58
Gambar 4.6 Klasifikasi Pada Parameter Morfologi Kedelai	59
Gambar 4.7 Tampilan Halaman Depan	60
Gambar 4.8 Tampilan Halaman Pengenalan Website	61
Gambar 4.9 Tampilan Halaman Hitung Polong	61
Gambar 4.10 Tampilan Hasil Hitung Polong	62
Gambar 4.11 Tampilan Halaman Klasifikasi	63
Gambar 4.12 Validasi dengan Pakar	69

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kedelai (*Glycine max* (L.) Merr) merupakan komoditas strategis bagi ketahanan pangan nasional Indonesia. Sebagai sumber protein nabati utama, kedelai menjadi bahan baku berbagai produk pangan seperti tahu, tempe, dan kecap. Namun, produksi kedelai didalam negeri masih belum mampu memenuhi kebutuhan domestik. Berdasarkan Pragnosa Neraca Pangan yang dihimpun NFA, total produksi dalam negeri pada tahun 2023 hanya mencapai 289 ribu ton, sementara kebutuhan kedelai nasional sekitar 248 ribu ton per bulan (Badan Pangan Nasional, 2023). Rendahnya produktivitas dan keterbatasan varietas unggul menjadi faktor utama dalam kesenjangan antara produksi dan kebutuhan ini. Meski demikian, penggunaan varietas kedelai unggul terbukti dapat memberikan keuntungan finansial yang layak, sehingga berpotensi meningkatkan pendapatan dan kesejahteraan petani (Krisdiana *et al.*, 2021).

Peneliti di bidang pemuliaan tanaman Universitas Sumatera Utara saat ini sedang melakukan pengembangan varietas kedelai dengan melakukan penyilangan pada tiga varietas unggul kedelai yaitu Anjasmoro, Dega-1, dan Grobogan. Anjasmoro memiliki keunggulan berupa polong yang tidak mudah pecah. Dega-1 unggul dengan biji yang berukuran besar, memiliki bobot rata-rata 22,98 gram per 100 biji, serta waktu panen yang relatif cepat. Sementara itu, Grobogan memiliki kelebihan pada bulir yang tidak mudah pecah dan potensi hasil yang besar. Pengembangan varietas kedelai unggul terkendala oleh lamanya proses pemuliaan tanaman konvensional. Umur panen kedelai umumnya berkisar 70-80 hari untuk mendapatkan galur murni yang stabil (Lagiman *et al.*, 2022). Hal ini menyebabkan pengembangan varietas baru menjadi lamban dan tidak responsif terhadap dinamika permintaan pasar.

Menurut penelitian yang dilakukan oleh Sjamsijah *et al.* (2018), jumlah polong isi per tanaman merupakan salah satu indikator utama produktifitas kedelai. Karakter jumlah polong total per tanaman memiliki korelasi positif yang kuat dengan hasil biji per tanaman dan analisis regresi menunjukkan bahwa jumlah polong total adalah karakter yang paling berpengaruh langsung terhadap sifat unggul tanaman kedelai. Namun, meskipun jumlah polong dan biji kedelai merupakan indikator penting untuk menilai sifat unggul dan potensi hasil tanaman kedelai, perhitungan manual jumlah polong dan biji menghadapi kendala praktis yang signifikan, terutama dalam skala besar dan penelitian lapangan. Proses perhitungan hasil panen yang masih dilakukan secara manual memerlukan waktu yang lama dan seringkali membutuhkan pengulangan perhitungan untuk menentukan hasil panen (Rahman *et al.*, 2023), karena harus menghitung polong dan biji satu per satu pada setiap tanaman dalam plot yang luas. Selain itu, variasi jumlah polong dan biji per tanaman mempersulit pencatatan yang akurat dan konsisten.

Penelitian ini bertujuan mengembangkan sistem deteksi polong dan klasifikasi kedelai berbasis *machine learning* dengan mengintegrasikan citra dan parameter morfologi tanaman pada fase generatif. Tantangan utama dalam seleksi bibit unggul adalah proses perhitungan jumlah polong dan biji yang selama ini dilakukan secara manual sehingga memakan waktu dan rentan akan kesalahan. Solusi yang diusulkan adalah menggunakan *computer vision* untuk mengotomasi perhitungan polong, biji dan klasifikasi tanaman terpilih kedelai sehingga dapat meningkatkan efisiensi dan mempercepat proses seleksi varietas kedelai.

Dengan mengintegrasikan *computer vision* dengan *machine learning* akan memungkinkan perhitungan dan klasifikasi polong yang akurat, mengurangi subjektifitas dalam melakukan proses seleksi, serta mempercepat pengambilan keputusan dalam pemuliaan tanaman. Selain itu, penelitian ini juga menggunakan parameter-parameter penting pada tanaman seperti tinggi tanaman, jumlah cabang, jumlah polong per tanaman, jumlah polong berisi, bobot biji, dan bobot 100 biji yang memberikan hasil yang komprehensif terhadap potensi hasil panen.

Dengan menggunakan teknik *high-throughput phenotyping* (HTP) yang mengintegrasikan *computer vision* dan analisis parameter morfologi untuk melakukan identifikasi varietas yang dapat diterapkan untuk mengoptimalkan kualitas produk pertanian (Zhang *et al.*, 2023). Sistem ini diharapkan mampu mempercepat proses

perhitungan dan seleksi untuk meningkatkan akurasi serta objektivitas dalam penilaian varietas tanaman melalui pendekatan yang menggabungkan visual dan parameter morfologi kedelai.

Penelitian ini menerapkan metode Faster R-CNN karena kemampuannya dalam deteksi objek yang memberikan hasil yang sangat baik. Berdasarkan penelitian yang dilakukan oleh Datta *et al.* (2020), Faster R-CNN terbukti unggul untuk tugas objek deteksi dibandingkan dengan metode lain seperti YOLO V3, SSD dengan akurasi 86,42%. Sementara itu, penelitian yang dilakukan oleh Hakim *et al.* (2024) membahas tentang penerapan metode Faster R-CNN untuk mendeteksi enam kategori kendaraan seperti bus, mobil, motor, mobil *pick up*, truk, dan truk kotak serta membandingkan empat arsitektur *backbone* yaitu ResNet50, ResNet50V2, MobileNetV2 Large, dan MobileNetV3 Large 320. Hasil menunjukkan bahwa ResNet50 memberikan kinerja keseluruhan yang terbaik dengan mAP 0,966 yang menawarkan *trade-off* yang seimbang antara *precision* dan *recall* dari pada metode lainnya seperti ResNet50V2 dan MobileNetV3.

Penelitian ini juga menerapkan XGBoost untuk melakukan klasifikasi, XGBoost dikenal karena kecepatan, efisiensi, dan kemampuannya dalam menskalakan dengan baik kumpulan data yang besar. Berdasarkan penelitian yang dilakukan oleh Li *et al.* (2020) membandingkan beberapa metode dalam melakukan klasifikasi data seperti Support Vector Machine, K-Nearest Neighbor, Native Bayes, Decision Tree, dan XGBoost. XGBoost terbukti memberikan hasil yang lebih baik dari model lainnya dengan akurasi sebesar 80,1% dengan menggunakan beberapa matrik seperti *accuracy*, *recall*, dan *F1-score*.

Faster R-CNN merupakan algoritma pengembangan dari metode Fast R-CNN, sebagai generasi terdahulu dari metode *object selection* yang menunjukkan hasil yang mengesankan pada objek deteksi (Charli *et al.*, 2020). Oleh karena itu, metode ini memiliki hasil potensi yang besar untuk melakukan deteksi dan kuantifikasi polong kedelai serta penggunaan XGBoost untuk melakukan klasifikasi tanaman terpilih dan tidak terpilih.

Berdasarkan latar belakang dan hasil dari penelitian terdahulu, penulis mengajukan sebuah penelitian yang akan menerapkan metode Faster R-CNN dengan arsitektur ResNet50 untuk melakukan deteksi dan kuantifikasi polong otomatis berdasarkan citra polong kedelai. Penggunaan Faster R-CNN diharapkan dapat

meningkatkan akurasi dan efisiensi waktu dalam melakukan perhitungan polong dan biji kedelai. Selain itu, XGBoost sebagai model prediksi akan memperkuat analisis dan memberikan hasil objektif dalam proses seleksi tanaman terpilih dan tidak terpilih. Oleh karena itu, penulis mengajukan penelitian dengan judul “Implementasi Metode Faster R-CNN dan XGBoost untuk Kuantifikasi Otomatis Polong dan Klasifikasi Varietas Terpilih Kedelai”.

1.2 Rumusan Masalah

Proses seleksi bibit unggul kedelai saat ini membutuhkan waktu yang lama sehingga menghambat pengembangan varietas baru untuk meningkatkan produktivitas nasional. Metode konvensional ini memiliki keterbatasan waktu, sumber daya, subjektivitas penilaian, dan kemampuan menganalisis data kompleks. Oleh karena itu, diperlukan teknologi yang dapat melakukan kuantifikasi perhitungan otomatis polong dan seleksi tanaman terpilih kedelai. Penerapan *artificial intelligence* dan *machine learning* berpotensi mengatasi masalah ini dengan meningkatkan efisiensi waktu dalam melakukan perhitungan polong, biji, dan melakukan seleksi tanaman terpilih kedelai berdasarkan ciri morfologi tanaman pada fase generatif. Penilaian ini bertujuan mengembangkan sistem berbasis *AI* untuk membantu pemuliaan tanaman melakukan seleksi lebih cepat, objektif, dan akurat sehingga dapat mempercepat pengembangan varietas unggul kedelai.

1.3 Batasan Masalah

Dalam penelitian ini, penulis membuat batasan masalah untuk menghindari adanya penyimpangan maupun pelebaran pokok masalah agar penelitian tersebut lebih terarah dan memudahkan dalam pembahasan sehingga tujuan penelitian tercapai. Adapun batasan masalah tersebut diantaranya yaitu:

1. Data citra polong kedelai dikumpulkan menggunakan *smartphone* oleh penulis, penelitian yang dilakukan oleh Jiang *et al.* (2025), dan data morfologi kedelai pada fase generatif di lahan pertanian Fakultas Pertanian Universitas Sumatera Utara.
2. Data citra polong yang digunakan memiliki format .png dan data morfologi kedelai memiliki format .xlsx.
3. Penelitian ini berfokus pada polong tanaman kedelai biji 1, 2, dan 3.

4. Penelitian ini difokuskan pada penggunaan metode Faster R-CNN dengan arsitektur ResNet50 sebagai *backbone* dan XGBoost.

1.4 Tujuan Penelitian

Berdasarkan latar belakang dan rumusan masalah yang sudah dipaparkan, tujuan dari penelitian ini adalah untuk mengimplementasikan dan menganalisis performa dari metode Faster R-CNN dengan *backbone* ResNet50 dalam melakukan perhitungan polong dan XGBoost dalam melakukan seleksi tanaman terpilih dan tidak terpilih kedelai berdasarkan ciri morfologi tanaman pada fase generatif.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut:

1. Membantu peneliti tanaman untuk menghitung polong dan biji kedelai pada citra polong, melakukan seleksi tanaman terpilih kedelai sehingga proses menjadi lebih cepat, efisien, dan konsisten.
2. Mengetahui performa Faster R-CNN dan XGBoost untuk melakukan kuantifikasi polong dan klasifikasi tanaman terpilih.
3. Mendukung upaya dalam meningkatkan produktivitas kedelai untuk mengurangi kesenjangan produksi agar bisa mencukupi kebutuhan nasional.

1.6 Metodologi Penelitian

Tahapan-tahapan yang dilakukan dalam penelitian ini adalah sebagai berikut:

1. Studi Literatur

Studi literatur adalah tahapan dalam pengumpulan data dengan berbagai sumber referensi mengenai varietas kedelai, morfologi tanaman kedelai, pengolahan citra, Faster R-CNN, ResNet50, dan XGBoost yang diperoleh dari berbagai buku, jurnal, artikel, dan berbagai sumber referensi lainnya.

2. Analisis Permasalahan

Tahap selanjutnya adalah analisis permasalahan literatur yang telah dikumpulkan pada tahap sebelumnya sehingga diperoleh penalaran menggunakan metode yang akan diimplementasikan untuk melakukan perhitungan otomatis polong dan seleksi varietas terpilih kedelai berdasarkan ciri morfologi tanaman pada fase generatif.

3. Perancangan Sistem

Setelah melakukan tahap analisis permasalahan, penelitian dilanjutkan dengan tahap perancangan sistem yang mencakup perancangan arsitektur umum sistem, pengumpulan data input dan perancangan desain antarmuka sistem.

4. Implementasi

Pada tahap ini, perancangan sistem yang sudah dibuat pada tahap sebelumnya diimplementasikan sehingga dapat menghasilkan sebuah sistem yang sesuai dengan tujuan penelitian.

5. Pengujian Sistem

Tahap pengujian sistem merupakan tahap pengujian terhadap sistem yang telah dibuat untuk memastikan bahwa sistem yang telah dibuat tersebut dapat beroperasi dengan baik serta dapat digunakan sebagai alat untuk menghitung jumlah polong dan seleksi tanaman terpilih atau tidak terpilih.

6. Penyusunan Laporan

Pada tahap ini peneliti akan melakukan penyusunan laporan dan dokumentasi proses penelitian sebagai bentuk representasi dari hasil penelitian yang telah dilakukan.

1.7 Sistematika Penulisan

Sistematika penulisan skripsi ini terdiri dari 5 bab yaitu pendahuluan, landasan teori, analisis dan perancangan sistem, implementasi dan pengujian sistem, kesimpulan dan saran.

BAB 1 : PENDAHULUAN

Bab satu memuat penjelasan mengenai latar belakang penelitian, perumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

BAB 2 : LANDASAN TEORI

Bab dua mencakup teori-teori yang mendasari penelitian ini, meliputi konsep dasar dan teori yang terkait seperti pemuliaan tanaman, *high-throughput phenotyping* (HTP), pengolahan citra digital, serta metode Faster R-CNN, ResNet50, dan XGBoost yang digunakan dalam penelitian ini.

BAB 3 : ANALISIS DAN PERANCANGAN SISTEM

Bab tiga mencakup arsitektur umum dari penelitian yang dilakukan yang terdiri dari pengambilan data gambar, langkah-langkah pada tahap *pre-processing*, proses *training* model Faster R-CNN ResNet50 dan XGBoost, evaluasi model, *testing*, evaluasi model, *testing*, dan perancangan antarmuka sistem.

BAB 4 : IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab empat, dijelaskan tentang pelaksanaan analisis dan perancangan sistem yang telah dijelaskan pada bab sebelumnya, serta menunjukkan hasil dari pengujian sistem yang telah dibangun.

BAB 5 : KESIMPULAN DAN SARAN

Bab lima berisi tentang kesimpulan dan saran dari hasil penelitian yang telah dilakukan guna pengembangan pada penelitian selanjutnya.

BAB II

LANDASAN TEORITIS

2.1 Kedelai (*Glycine max* (L.) Merr)

Kedelai (*Glycine max* (L.) Merr) merupakan salah satu tanaman pangan terpenting di dunia dan memegang peran strategis dalam ketahanan pangan, terutama di Indonesia. Tanaman ini mengandung rata-rata 40% protein, 35% karbohidrat, 20% minyak, dan 5% abu (Liu *et al.*, 2004), menjadikannya sumber utama protein dan lemak bagi manusia maupun hewan. Minyak kedelai sendiri menyumbang 56% dari produksi minyak nabati global, menempatkannya sebagai minyak nabati dengan tingkat konsumsi tertinggi kedua setelah minyak kelapa sawit. Selain nilai ekonominya yang tinggi, kedelai juga memberikan berbagai manfaat kesehatan. Konsumsi kedelai terbukti dapat mengurangi risiko penyakit kronis seperti kanker, penyakit jantung, dan osteoporosis (Messina *et al.*, 2010). Hal ini disebabkan oleh kandungan nutrisinya yang kaya, termasuk vitamin, mineral, isoflavon, dan asam folat, yang berperan penting dalam menjaga kesehatan tubuh.

Sebagai tanaman yang berasal dari Asia Timur, kedelai telah dibudidayakan secara luas di berbagai belahan dunia, termasuk Indonesia. Di Indonesia, kedelai menjadi sumber protein nabati utama bagi masyarakat, terutama melalui produk olahan seperti tahu, tempe, dan kecap. Tempe, sebagai hasil inovasi lokal, bahkan dianggap sebagai makanan bernutrisi tinggi dan banyak diminati oleh masyarakat. Namun, meskipun memiliki potensi besar, produksi kedelai dalam negeri hanya memenuhi sekitar 6,5% dari total konsumsi nasional. Dari sisi impor, Indonesia mengimpor sebagian besar dari Brazil, Amerika Serikat, dan Argentina. Impor pada tahun 2023 terbanyak yaitu dari Brazil dengan nilai impor mencapai USD 1,99 miliar atau 43,58% dari total impor kedelai Indonesia (Pusat Data dan Sistem Informasi Pertanian, 2024). Ketergantungan pada impor ini tidak hanya memengaruhi stabilitas harga kedelai tetapi juga mengancam keberlangsungan industri pengolahan kedelai lokal, seperti tahu dan tempe, yang melibatkan jutaan tenaga kerja.

2.2 Varietas Unggul Kedelai

Varietas unggul menjadi salah satu teknologi pendongkrak produktivitas, dimana pemilihan varietas yang tepat akan berdampak pada pertumbuhan dan hasil yang akan diperoleh. Hasil yang rendah atau sedikit di tingkat petani salah satunya bisa disebabkan oleh penggunaan varietas yang tidak sesuai dengan kondisi lingkungan dan varietas yang digunakan (Hamdani *et al.*, 2023). Beberapa varietas unggul yang disilangkan pada penelitian ini yaitu:

2.2.1 *Anjasmoro*

Anjasmoro adalah salah satu varietas unggul nasional dan kian diminati oleh para petani. Varietas Anjasmoro memiliki kualitas fisik yang setara dengan kedelai impor, ciri-ciri keunggulan varietas Anjasmoro yaitu memiliki ukuran biji yang besar kisaran 14,8 hingga 15,3 gram per 100 biji dengan daya hasil yang cukup tinggi berkisar 2,03 sampai 2,25 ton/ha (Kementerian Pertanian Republik Indonesia, 2022).

2.2.2 *Dega*

Dega si kedelai genjah merupakan keturunan dari hasil perkawinan silang antara varietas Grobogan dan Malabar. Persilangan buatan dilakukan pada tahun 2008 dan selanjutnya dilakukan penggaluran pada tahun 2010 sampai 2012 hingga diperoleh galur Dega. Umur genjah mengurangi resiko tanaman terhadap cekaman, baik biotik maupun abiotik khususnya kekeringan, mudah ditanam pada lahan sawah, serta mengurangi konsumsi air dan biaya pengairan.

Dega mempunyai keunggulan seperti umur genjah (± 71 hari), potensi hasil tinggi mencapai 3,82 ton/ha, ukuran biji besar sekitar 22,98 ton per 100 biji, dan dapat beradaptasi dengan baik di segala lokasi (Kementerian Pertanian Republik Indonesia, 2021).

2.2.3 *Grobogan*

Grobogan merupakan salah satu komoditi pangan unggulan di Kabupaten Grobogan yang dilepas pemerintah pada tahun 2008. Kedelai Grobogan mempunyai data tarik tersendiri karena varietasnya yang terbaik di Indonesia, yakni memiliki warna biji yang putih kekuningan, ukuran biji yang besar sekitar 18 gram per 100 biji, tingkat produktivitas tanamannya yang tergolong cukup

tinggi yaitu berkisar 2,77 ton/ha hingga 3,4 ton/ha, genjah, tidak mudah pecah, serta memiliki kandungan protein yang tinggi (Gebyar Perbenihan Tanaman Pangan, 2023).

Berdasarkan umur tanaman, varietas-varietas unggul kedelai diklasifikasikan menjadi 3 kelompok yaitu varietas yang berumur kurang dari 75 hari (genjah), varietas yang berumur 75-90 hari (sedang), dan varietas yang berumur lebih dari 90 hari (tinggi) (Widyawati, 2008). Potensi hasil suatu varietas unggul yang ditanam menentukan produktivitas tanaman tersebut dan merupakan salah satu komponen penting dalam upaya meningkatkan produksi, produktivitas kedelai, dan terbukti memberikan keuntungan secara finansial dan layak sehingga berdampak terhadap peningkatan pendapatan dan kesejahteraan petani (Krisdiana, 2021).

2.3 Citra Polong Kedelai

Polong kedelai merupakan data citra visual yang penting dalam proses perhitungan pada deteksi objek. Citra polong yang dihasilkan berguna untuk mengidentifikasi atau mengenali ciri-ciri dari polong tersebut, seperti bentuk, jumlah biji pada polong yang dapat menjadi parameter atau indikator keberhasilan varietas.

Metode yang digunakan untuk memproses citra ini adalah Faster R-CNN dengan arsitektur ResNet50 yang merupakan metode *deep learning* yang dapat melakukan klasifikasi objek pada citra. Penelitian ini menggunakan teknologi dengan metode tersebut untuk menjadikan citra polong kedelai sebagai alat yang efisien dan akurat dalam mengenali polong sehingga mengurangi ketergantungan pada perhitungan manual serta akan meningkatkan hasil pemuliaan tanaman kedelai.



Gambar 2.1 Citra Polong Kedelai

2.4 Morfologi Tanaman Kedelai

Morfologi tanaman kedelai memegang peranan penting dalam menentukan apakah suatu varietas dapat dikategorikan unggul atau tidak. Penentuan parameter merupakan langkah kunci, yang dilakukan dengan mengidentifikasi istilah-istilah umum yang digunakan oleh petani, yang kemudian akan dijadikan parameter kunci dalam seleksi varietas unggul kedelai (Kumalasari, 2017). Beberapa parameter digunakan yaitu :

1. Nomor tanaman

Parameter ini menunjukkan identitas dari tanaman tersebut, seperti jenis varietas, fase, nomor, dan baris tanaman.

2. Tinggi Tanaman (cm)

Parameter ini mengukur panjang batang utama dari leher akar hingga titik tumbuh paling atas. Tinggi tanaman dipengaruhi oleh faktor genetik dan lingkungan (cahaya dan nutrisi). Tanaman dengan tinggi yang optimal biasanya memiliki struktur yang baik untuk penyerapan cahaya.

3. Umur Berbunga (hari)

Parameter ini berisi nilai tanaman tersebut berbunga di hari ke-berapa, dimana cepat lambatnya umur berbunga akan menentukan umur panen. Rata-rata umur berbunga varietas anjasmoro, grobogan, dan dega berkisar 30 sampai 35 hari.

4. Jumlah Cabang

Parameter ini berisi jumlah cabang pada satu tanaman kedelai, dimana jumlah cabang produktif berpengaruh sangat nyata terhadap jumlah polong per tanaman (Tanjung *et al.*, 2022).

5. Umur Panen (hari)

Parameter ini berisi nilai atau hari seberapa tanaman tersebut dapat dipanen.

6. Jumlah Polong per Tanaman

Jumlah polong kedelai dipengaruhi oleh kondisi lingkungan (Sjamsijah *et al.*, 2018). Ketersediaan air yang cukup pada saat pertumbuhan generatif juga dapat meningkatkan jumlah polong.

7. Jumlah Polong Berbiji 1, 2, 3

Parameter ini berisi jumlah polong yang berbiji satu, dua, tiga, pada suatu tanaman kedelai.

8. Jumlah Biji/Tanaman

Parameter ini bersisi total dari jumlah biji yang terdapat pada polong dihasilkan dari tanaman kedelai tersebut. Ukuran biji kedelai dapat diklasifikasikan menjadi 3 kelas yaitu biji kecil (6-10 gr/100 biji), biji sedang (11-12 gr/100 biji), dan biji besar (>12 gr/100 biji) (Fachruddin, 2000).

9. Bobot Biji/Tanaman (gr)

Parameter ini berisi berat total atau bobot total biji yang dihasilkan per tanaman. Bobot biji yang dihasilkan diduga dipengaruhi oleh perbedaan karakter antara varietas itu sendiri, pengaruh lingkungan, dan faktor genetik (Lakitan *et al.*, 2011).

10. Bobot 100 Biji (gr)

Parameter ini berisi hasil dari perhitungan bobot per 100 biji kedelai secara acak dari hasil biji setelah dikeringkan pada setiap petak, kemudian bobot kedelai ditimbang dengan neraca analitik untuk mengetahui besar kecilnya ukuran biji (Suryadi *et al.*, 2020).

2.5 Pengolahan Citra

Meskipun sebuah citra kaya informasi, namun seringkali citra yang kita miliki mengalami penurunan mutu (degradasi), misalnya mengandung cacat atau derau (*noise*), kurang tajam, kabur (*blur*), dan sebagainya. Tentu saja citra tersebut menjadi sulit diinterpretasi karena informasi yang disampaikan oleh citra tersebut berkurang. Agar citra yang mengalami gangguan mudah diinterpretasi, maka citra tersebut perlu dimanipulasi menjadi citra yang kualitasnya lebih baik.

Pengolahan citra digital adalah cabang ilmu teknik yang mempelajari teknik-teknik untuk mengolah citra. Tujuan dilakukannya pengolahan citra digital adalah untuk mengolah citra agar lebih mudah diinterpretasikan sehingga informasi dalam gambar dapat diolah (Sumijan *et al.*, 2021).

2.6 Labeling

Labeling merupakan tahap dalam pemberian label atau identifikasi pada objek dalam citra untuk memudahkan analisis lebih lanjut. Label ini biasanya diberikan oleh manusia atau ahli *domain* yang mengetahui informasi yang sebenarnya dari setiap contoh data. Pemberian label ini dilakukan dengan tujuan untuk melatih model *machine learning* agar dapat memahami pola atau karakteristik dari setiap kelas. Dalam data

berlabel, kita tahu apa yang menjadi "jawaban benar" untuk setiap contoh data, sehingga model dapat belajar dari data tersebut dan diuji dengan akurasi yang tinggi.

2.7 Resizing

Resizing adalah langkah penting dalam *preprocessing* citra untuk *machine learning*. Dengan mempertahankan *region of interest* (ROI), meningkatkan efisiensi komputasi, dan memastikan konsistensi data input, *resizing* membantu meningkatkan performa dan akurasi model. Teknik-teknik canggih seperti *content-aware resizing* juga memastikan kualitas visual citra tetap terjaga, sehingga model dapat fokus pada informasi yang relevan. Dengan demikian, *resizing* tidak hanya mempercepat proses komputasi tetapi juga meningkatkan kualitas hasil analisis dalam aplikasi *computer vision* (Saponara *et al.*, 2022).

2.8 Augmentation

Augmentation merupakan teknik dalam pengelolaan gambar yang digunakan untuk memperluas dataset *training* dengan membuat variasi pada gambar-gambar yang ada. Data biasanya diaugmentasi dengan melakukan transformasi pada data atau membuat salinan dari sumber data tanpa mengubah label yang tertera pada setiap bagian data tersebut.

Augmentasi data merupakan salah satu teknik yang penting untuk dilakukan pada *computer vision*. Manfaatnya dapat membuat mesin mampu mengenali dari berbagai citra yang berbeda, menambah data sehingga memperkecil kemungkinan adanya *overfitting*. Beberapa transformasi yang umum digunakan dalam augmentasi citra meliputi:

1. *Rotating* (Rotasi)

Gambar diputar searah jarum jam atau berlawanan arah jarum jam dengan beberapa derajat, mengubah posisi objek dalam bingkai.

2. *Flipping* (Memutarbalikkan)

Merefleksikan gambar di sekitar sumbu vertikal atau horizontalnya atau keduanya. Ini dapat membantu memaksimalkan jumlah gambar (Khalifa *et al.*, 2021)

3. *Brightness* (Kecerahan)

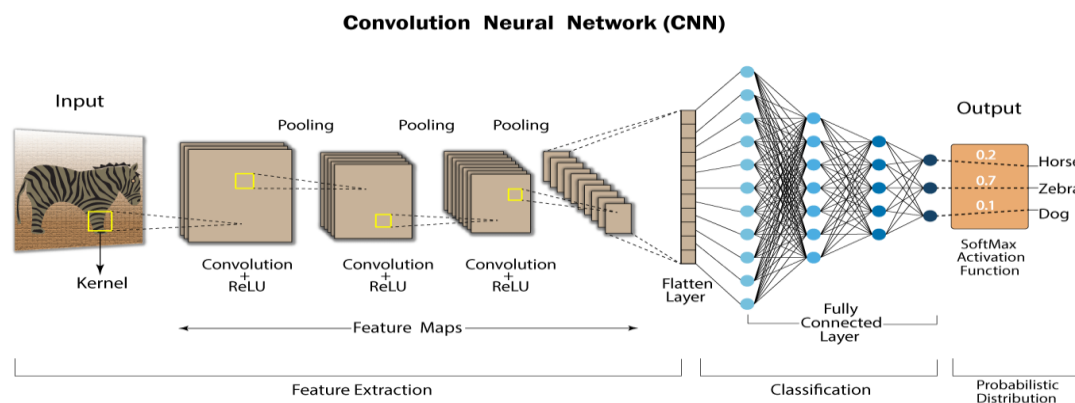
Tingkat kecerahan dapat melatih sistem dengan gambar dengan intensitas warna yang berbeda, sehingga model dapat menggeneralisasi gambar yang tidak dilatih bahkan pada kondisi pencahayaan yang berbeda.

4. *Sharpen* (Penajaman)

Memberikan penajaman atau meningkatkan kontras untuk dapat menghasilkan enkapsulasi detail yang lebih banyak dari objek (Shorten *et al.*, 2019).

2.9 Convolutional Neural Network

Convolutional Neural Network (CNN) adalah salah satu metode *machine learning* dari pengembangan *Multi Layer Perceptron* (MLP) yang didesain untuk mengolah data dua dimensi. CNN merupakan modifikasi dari *Deep Neural Network* (DNN) yang digunakan dalam pengenalan gambar karena memerlukan pra-pemrosesan (Pratama *et al.*, 2020). Struktur utama CNN terdiri dari *convolution*, *activation*, *pooling*, *fully connected*, dan *output layers*. *Convolution layers* menggunakan filter yang memindai gambar input untuk menangkap fitur-fitur penting, seperti tekstur tepi, bentuk di mana lapisan awal menangkap fitur sederhana, dan lapisan lebih dalam menangkap pola yang lebih kompleks. Berikut adalah arsitektur CNN yang dapat dilihat pada Gambar 2.2.



Gambar 2.2 Arsitektur CNN (Sumber: Developersbreach.com)

Setelah melewati *convolution layers*, CNN menggunakan *activation layers* yang berfungsi untuk menggantikan semua nilai negatif dengan nol untuk memasukkan non-linearitas ke dalam model, yang membantu jaringan mempelajari pola yang lebih kompleks.

Selain *activation layers*, CNN menggunakan *pooling layer* yang berfungsi untuk menjaga ukuran data ketika *convolution* dilakukan, yaitu dengan melakukan reduksi sampel (*down sampling*). Dengan *pooling*, kita dapat merepresentasikan data menjadi

lebih kecil, mudah untuk dikelola, dan mudah dalam mengontrol *overfitting* (Charli *et al.*, 2020).

Setelah melalui *convolution layers* dan *pooling*, data yang telah diproses diteruskan ke *fully connected layers*. Lapisan ini adalah lapisan yang digunakan untuk melakukan transformasi pada dimensi data agar dapat diklasifikasikan secara linear. Untuk menghasilkan *output* dari lapisan ini tidak dibutuhkan operasi konvolusi, tetapi menggunakan komputasi perkalian matriks yang diikuti dengan *bias offset*. Dengan operasi ini, setiap *neuron* memiliki koneksi penuh ke semua dalam lapisan sebelumnya, sehingga layer ini disebut sebagai *fully connected layer* (Alwanda *et al.*, 2020). Setelah melewati *fully connected layer*, ada *output layer* sebagai lapisan terakhir yang berfungsi untuk menghasilkan probabilitas kelas akhir dari input gambar.

2.10 ResNet 50

Residual Network (ResNet) adalah arsitektur *neural network* yang diperkenalkan oleh (He *et al.*, 2015). Arsitektur ini dirancang untuk mengatasi masalah yang sering muncul dalam pelatihan *deep learning*, seperti waktu pelatihan yang lama dan batasan pada jumlah lapisan. Solusi yang ditawarkan oleh ResNet adalah penerapan *skip connection* atau *shortcut*, yang memungkinkan arsitektur model tetap efektif meskipun semakin dalam.

ResNet memiliki kelebihan pada kinerja model yang stabil dan tidak menurun meskipun jumlah lapisannya bertambah dibandingkan arsitektur CNN lainnya. Selain itu, ResNet juga lebih efisien dalam perhitungan komputasi dan memiliki kemampuan yang lebih baik dalam melatih jaringan. Pada percobaan menggunakan *dataset* ImageNet menunjukkan bahwa arsitektur ResNet yang memiliki semakin banyak *layer* akan mendapatkan *training error* yang lebih rendah, serta akurasi yang lebih tinggi (He *et al.*, 2016).

Penelitian yang dilakukan oleh He *et al.* (2016) menguji beberapa jenis arsitektur ResNet. Mereka menemukan terdapat beberapa jenis arsitektur ResNet yang cukup optimal untuk digunakan seperti ResNet18, ResNet34, ResNet50, ResNet101, dan ResNet152. Perbedaan pada setiap jenis arsitektur ResNet terletak pada jumlah atau banyaknya *layer* yang digunakan. Penelitian ini akan berfokus untuk mengimplementasikan arsitektur ResNet50 sebagai *backbone* karena memiliki karakteristik yang berbeda dari *skip connection* dan struktur pembelajaran residual. Hal

tersebut dapat memecahkan masalah pelatihan jaringan dalam, penghilangan gradien, dan ledakan gradien dengan lebih baik (Liu *et al.*, 2021). Tabel 1 merupakan rincian lapisan yang digunakan pada arsitektur ResNet50 berdasarkan penelitian yang dilakukan oleh (He *et al.*, 2016).

Tabel 2.1 Rincian Layer Arsitektur ResNet50

<i>Layer</i>	<i>Ukuran Output</i>	<i>ResNet50</i>
Conv1	112×112	7×7 , 64, <i>stride</i> 2 <i>3x3 max pool, stride</i> 2
Conv2_x	56×56	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
Conv3_x	28×28	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
Conv4_x	14×14	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$
Conv5_x	1×7	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	<i>Average pool,</i> <i>1000-d fc, softmax</i>
FLOPs		3.8×10^9

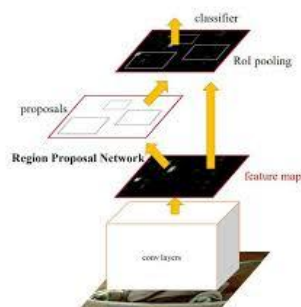
Pada penelitian yang dilakukan oleh Srinidhi *et al.* (2025) ResNet50 yang mengintegrasikan *Feature Pyramid Network* (FPN) diterapkan untuk deteksi objek berukuran kecil pada citra aerial menggunakan dataset SODA-A. Melalui rangkaian pra-pemrosesan, perancangan arsitektur, pelatihan, dan evaluasi, model ResNet50-FPN mencapai *accuracy* 84,12%, *precision* 84,52%, *recall* 85,69%, dan *F1-score* 85,10%. Hasil tersebut mengindikasikan bahwa model tersebut mampu meningkatkan akurasi dan ketangguhan deteksi pada skenario kompleks dengan objek berukuran sangat kecil, *occlusion*, kamuflase, dan *clutter*. Oleh karena itu, pendekatan ini efektif untuk otomatisasi deteksi objek skala kecil dan berpotensi dimanfaatkan pada pengawasan, pemantauan batas laut atau darat, monitoring satwa liar, manajemen bencana, serta perencanaan perkotaan.

2.11 Fast R-CNN

Fast R-CNN adalah model *deep learning* yang dirancang untuk deteksi objek dengan menggabungkan efisiensi dan akurasi. Berbeda dengan pendahulunya (R-CNN), Fast R-CNN memproses seluruh citra sekaligus menggunakan jaringan konvolusi untuk menghasilkan *feature map*, kemudian menggunakan *Region of Interest (RoI) Pooling* untuk mengekstrak fitur dari setiap *region proposal*. Fitur-fitur ini diproses oleh *fully connected layers* yang menghasilkan dua *output* yaitu probabilitas kelas objek (melalui *softmax*) dan regresi *bounding box* untuk menyempurnakan lokasi objek. Keunggulan utama Fast R-CNN adalah kemampuannya untuk melatih seluruh jaringan secara *end-to-end*, yang mengurangi waktu komputasi dan meningkatkan akurasi. Model ini menjadi fondasi untuk pengembangan Faster R-CNN, yang lebih lanjut meningkatkan kecepatan dengan menggantikan *selective search* dengan *Region Proposal Network* (RPN) (Girshick *et al.*, 2015).

2.12 Faster R-CNN

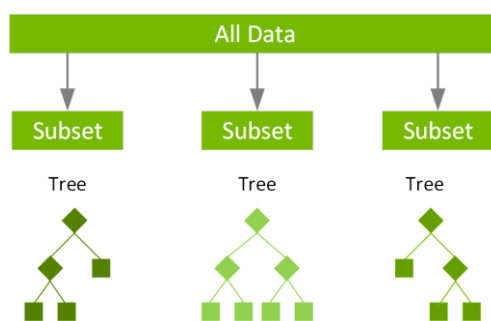
Faster Region-Based Convolutional Neural Network (Faster R-CNN) adalah metode yang digunakan untuk mendeteksi objek dalam sebuah gambar. Faster R-CNN merupakan pengembangan dari algoritma Fast R-CNN. Pada algoritma Fast R-CNN, *selective search* digunakan untuk mengusulkan *region*, namun metode ini terlalu lambat karena *selective search* membutuhkan waktu setidaknya dua detik untuk memproses setiap gambar, sehingga tidak efektif untuk menerapkannya. Oleh karena itu, dikembangkanlah Faster R-CNN yang menggunakan *Region Proposal Network* (RPN). *Region Proposal Network* (RPN) yaitu tugas cepat yang berguna untuk mencari pada input gambar kemungkinan lokasi dari objek. Posisi dari objek yang ada pada gambar memiliki kemungkinan objek dibatasi dari wilayah yang diketahui sebagai *region of interest (ROI)* (Charli *et al.*, 2020)



Gambar 2.3 Faster R-CNN

2.13 XGBoost

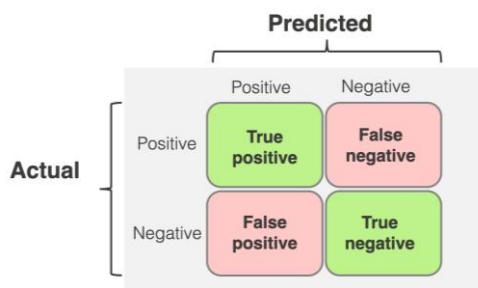
Extreme Gradient Boosting merupakan algoritma *machine learning* yang termasuk dalam kategori pembelajaran *ensemble*, yang menggabungkan prediksi dari beberapa model untuk meningkatkan kinerja keseluruhan. Pendekatan ini berbasis pada teknik *boosting*, di mana model-model lemah (*weak learners*), biasanya berupa pohon keputusan (*decision trees*) yang sederhana, digabungkan menjadi model yang kuat (*strong learner*). XGBoost merupakan metode implementasi peningkatan gradien yang dapat diskalakan dan sangat akurat yang mendorong batas daya komputasi untuk algoritma pohon yang ditingkatkan, yang sebagian besar dibangun untuk memberi energi pada kinerja model pembelajaran mesin dan kecepatan komputasi.



Gambar 2.4 XGBoost (Sumber: Nvidia.com)

2.14 Confusion Matrix

Confusion matrix atau matrik evaluasi merupakan salah satu teknik yang dapat digunakan untuk mengukur kinerja suatu model dengan hasil klasifikasi pada *machine learning*. *Confusion matrix* berupa tabel matrik yang menggambarkan kinerja model klasifikasi pada data uji.



Gambar 2.5 Confusion Matrix

Confusion matrix terdiri dari empat nilai yang digunakan untuk mengevaluasi performa dari model, yaitu:

1. TP (*True Positive*): Diprediksi positif oleh model dan sebenarnya positif
2. TN (*True Negative*): Diprediksi negatif oleh model dan sebenarnya negatif.
3. FP (*False Positive*): Diprediksi positif oleh model namun sebenarnya negatif.
4. FN (*False Negative*): Diprediksi negatif oleh model namun sebenarnya positif.

Matrik evaluasi adalah parameter yang dipakai untuk mengevaluasi kinerja dari model. Seluruh matrik evaluasi dihitung berdasarkan TP, TN, FP, dan FN yang telah disebutkan:

1. Recall

Sensitivitas (*Recall*) adalah metrik evaluasi yang menggambarkan seberapa baik suatu model dalam mengidentifikasi kelas positif dengan benar. *Recall* akan menunjukkan kemampuan model kita untuk menganali review positif yang sebenarnya.

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

2. Precision

Presisi mengukur sejauh mana hasil positif yang diklasifikasikan oleh model benar. Presisi dihitung dengan membagi jumlah hasil positif yang benar dengan jumlah total hasil positif yang diklasifikasikan oleh model.

$$Precision = \frac{TP}{TP + FP}$$

3. Intersection over Union (IoU)

IoU menghitung rasio area tumpang tindih (*intersection*) antara hasil prediksi dan *ground truth* dengan area gabungan (*union*) antara keduanya. Matrik ini memberikan ukuran sejauh mana area prediksi mencakup atau cocok dengan area *ground truth*. Semakin tinggi nilai IoU, semakin baik hasil prediksi atau deteksi objeknya.

$$IoU = \frac{TP}{TP + FP + FN}$$

2.15 Penelitian Terdahulu

Penelitian yang dilakukan oleh Tonael *et al.* (2021) berfokus pada seleksi fitur benih padi unggul dengan tujuan pengoptimalan hasil panen padi berkualitas tinggi. Proses ini dilakukan dengan mengekstraksi fitur menggunakan metode *Color Moment* dan

GLCM untuk menganalisis karakteristik visual benih. Klasifikasi dilakukan dengan menggunakan algoritma KNN untuk menentukan kelas benih dan SVM untuk memisahkan dua kelas dengan *hyperplane* terbaik. Evaluasi sistem dilakukan menggunakan matriks. Penelitian ini berhasil dilakukan menggunakan metode *deep learning* dengan akurasi tertinggi mencapai 92,85%.

Penelitian yang dilakukan oleh Hanafiah *et al.* (2023) bertujuan untuk mengembangkan varietas unggul dengan melakukan teknik persilangan tiga arah menggunakan varietas Grobogan, Anjasmoro, dan Dega-1. Tujuannya adalah menciptakan kedelai yang tangguh, produktif, dan adaptif terhadap kondisi lingkungan. Berbagai parameter yang diukur pada penelitian ini adalah tinggi tanaman, jumlah polong, dan tingkat keberhasilan persilangan. Hasil penelitian menunjukkan tingkat keberhasilan persilangan antara Grobogan dan Anjasmoro sebesar 57,1%, sementara persilangan 3 arah dengan Dega-1 mencapai 35,37%. Dega-1 menunjukkan potensi pertumbuhan terbaik dengan tanaman lebih tinggi dan jumlah polong yang lebih banyak. Teknik persilangan tiga arah ini berhasil meningkatkan keragaman genetik dan potensi hasil panen kedelai dan menawarkan solusi untuk tantangan pertanian di masa depan.

Penelitian yang dilakukan oleh Zhao *et al.* (2023) melakukan pengembangan alat penghitungan otomatis biji kedelai (P2PNet-Soy) bertujuan untuk memprediksi hasil panen dan meningkatkan efisiensi seleksi dalam program pemuliaan. Metode tradisional yang manual seringkali memakan waktu, rentan error, dan kurang akurat dalam pelokalisasian. Untuk mengatasi hal ini, P2PNet-Soy menggunakan beberapa strategi, seperti penggabungan hasil penghitungan berlebih dengan *unsupervised clustering*, penggabungan fitur level rendah dan tinggi untuk informasi lebih lengkap, serta penggunaan *atrous convolution* dengan ukuran kernel berbeda agar model dapat mengenali variasi ukuran biji kedelai. Selain itu, *channel* dan *spatial attention* membantu memisahkan latar belakang dan biji untuk memudahkan penghitungan dan pelokalisasian. Model diuji pada 24 aksesori kedelai, dengan hasil *mean absolute error* (MAE) turun dari 105,55 (P2PNet asli) menjadi 12,94. Model ini juga efektif bekerja pada gambar langsung dari lapangan tanpa gangguan latar belakang.

Penelitian yang dilakukan oleh Leite *et al.* (2018) bertujuan memilih genotipe kedelai unggul yang membawa gen RR dengan performa agronomi baik menggunakan analisis multivariat dan indeks seleksi, serta mengidentifikasi karakter agronomi yang

berpengaruh. Eksperimen dilakukan pada 227 genotipe generasi F5, dengan 85 di antaranya tahan glifosat berdasarkan uji PCR. Karakter yang dievaluasi meliputi umur panen, tinggi tanaman, ketahanan rebah, nilai agronomi, jumlah cabang, jumlah polong, berat 100 biji, dan hasil biji. Analisis komponen utama berhasil memilih 16 genotipe dengan hasil biji tinggi, di mana karakter terkait komponen produksi sangat berpengaruh. Pengelompokan dengan metode *K-means* dan *Ward* menunjukkan kesamaan dalam mengelompokkan genotipe terpilih. Hasil analisis multivariat juga sesuai dengan indeks seleksi Mulamba dan Mock. Metode yang digunakan terbukti efektif untuk seleksi genotipe.

Penelitian yang dilakukan oleh Abdullah dan Oothariasamy (2020), dimana penelitian ini mengembangkan sistem pada deteksi dan perhitungan kendaraan berbasis lalu lintas perkotaan. Penelitian ini menggunakan *TensorFlow* dengan pendekatan *transfer learning*. Metode yang digunakan yaitu YOLOv3, DarkNetv19, Faster R-CNN, ResNet 101, dan SSD Inception. Hasil kinerja dari beberapa model tersebut mencapai 80,90%.

Penelitian yang dilakukan oleh Zhang *et al.* (2021) mengusulkan pendekatan prediktif untuk memperkirakan volume penjualan di toko fisik menggunakan analisis deret waktu dan rekayasa fitur. Dengan mempertimbangkan variabel eksternal seperti cuaca dan suhu, model *XGBoost* dipilih sebagai algoritma utama karena keunggulannya dalam menangani data non-linear serta efisiensi dalam iterasi dibandingkan dengan model lain seperti GBDT, LSTM, Prophet, dan ARIMA. Hasil eksperimen pada dua dataset menunjukkan bahwa *XGBoost* mencapai performa terbaik dengan RMSE dan MAE terendah yaitu 0,22 pada dataset pertama dan 0,06 pada dataset kedua, sekaligus membutuhkan iterasi yang lebih sedikit dibandingkan GBDT, menjadikannya metode yang optimal untuk prediksi volume penjualan di industri ritel.

Penelitian yang dilakukan oleh Li *et al.* (2020) ini mengusulkan penggunaan algoritma *XGBoost* dengan kombinasi fitur numerik dan fitur teks yang diekstraksi dari data eksperimen. Dengan perkembangan teknologi internet, jumlah serta kompleksitas data medis terus meningkat, sehingga pendekatan *machine learning* menjadi solusi yang lebih efektif dalam mendeteksi pola tersembunyi dalam data kesehatan. Penelitian ini melakukan perbandingan beberapa model seperti Random Forest, Support Vector Machine (SVM), Decision Tree (DT), Naïve Bayes (NB), Logistic Regression (LR), dan BP Neural Network untuk prediksi penyakit, tetapi masih memiliki keterbatasan

dalam akurasi dan efisiensi komputasi. Dalam penelitian ini, model XGBoost terbukti memiliki keunggulan signifikan dibandingkan metode lain, dengan akurasi mencapai 81,2% dan waktu operasi yang jauh lebih singkat dibandingkan algoritma lainnya. Dengan demikian, metode ini menjadi pendekatan yang efisien dan andal dalam membantu deteksi dini diabetes serta meningkatkan efektivitas pencegahan penyakit kronis.

Penelitian yang dilakukan oleh Kurniawan *et al.* (2025) melakukan klasifikasi terhadap makanan tradisional Indonesia dengan melakukan perbandingan kepada dua arsitektur *convolutional network* yaitu AlexNet dan ResNet50. Hasil eksperimen menunjukkan bahwa ResNet50 secara konsisten mengungguli AlexNet di semua matrik evaluasi. ResNet50 mendapatkan rata-rata akurasi sebesar 92% sementara AlexNet sebesar 86% sehingga ResNet50 lebih efektif digunakan dalam mempelajari pola visual untuk gambar makanan tradisional yang beragam. Penelitian ini menyimpulkan bahwa ResNet50 adalah pilihan yang lebih optimal untuk tugas klasifikasi citra makanan tradisional Indonesia.

Penelitian yang dilakukan oleh Datta *et al.* (2020) membahas tentang pentingnya deteksi objek dalam kehidupan sehari-hari, terutama dalam mobil otonom, sistem bantuan pengemudi cerdas, dan analisis lalu lintas. Di Bangladesh, deteksi objek yang akurat dari video *real-time* sangat penting untuk memberikan informasi yang jelas tentang lingkungan sekitar kendaraan. Namun, algoritma yang lebih cepat dari Faster R-CNN seringkali kurang akurat dalam mendeteksi objek kecil hingga sedang. Untuk mengatasi masalah ini, peneliti menggunakan Faster R-CNN yang dilatih untuk 19 kelas objek. Hasilnya, model ini mencapai akurasi deteksi sebesar 86,42% dalam analisis video *real-time*. Selain itu, *False Positive Rate* (FPR) dan *False Negative Rate* (FNR) dihitung untuk evaluasi, dengan hasil *FPR* sebesar 15,97% dan *FNR* sebesar 12,2%. Hasil dari penelitian ini menunjukkan bahwa Faster R-CNN efektif untuk deteksi objek dalam sistem bantuan pengemudi cerdas dan analisis lalu lintas.

Penelitian yang dilakukan oleh Hakim *et al.* (2024) membahas tentang penerapan Faster R-CNN untuk mendeteksi enam kategori kendaraan seperti bus, mobil, motor, mobil *pick up*, truk, dan truk kotak serta membandingkan empat arsitektur *backbone* yaitu ResNet50, ResNet50V2, MobileNetV2 Large, dan MobileNetV3 Large 320. Hasil menunjukkan ResNet50 memberikan kinerja keseluruhan yang terbaik dengan mAP 0,966 yang menawarkan *trade-off* yang seimbang antara *precision* dan

recall daripada metode lainnya. Temuan ini memberikan wawasan mengenai kesesuaian arsitektur *backbone* untuk deteksi kendaraan, khususnya pada aplikasi pemantauan lalu lintas.

Tabel 2.2 Penelitian Terdahulu

No	Penulis	Judul	Tahun	Keterangan
1.	Tonael, Kaesmetan, Lamabelawa,	Selection of Superior Rice Seed Features Using Deep Learning Method	2021	Berfokus pada seleksi fitur benih padi unggul menggunakan metode <i>Color Moment</i> dan GLCM untuk menganalisis karakteristik visual benih menggunakan algoritma KNN untuk menentukan kelas benih dan SVM untuk memisahkan dua kelas dengan <i>hyperplane</i> terbaik dan menggunakan <i>deep learning</i> sehingga mendapatkan hasil 92.85%.
2.	Hanafiah, Lubis, Haryanti, Setiado, Damanik, Limbong, Silaen, Joshua, Lestami	Assembly of Soybean Genotypes Developed Through Three-Way Cross	2023	Penelitian ini mengembangkan varietas unggul dengan menyilangkan tiga varietas Grobogan, Anjasmoro, dan Dega-1 dengan tujuan untuk menciptakan kedelai yang tangguh, produktif, dan adaptif terhadap kondisi lingkungan.
3.	Zhao, Kaga, Yamada, Komatsu, Hirata, Kikuchi, Hirafuji, Ninomiya, Guo	Improved Field-Based Soybean Seed Counting and Localization with Feature Level Considered	2023	Penelitian ini mengembangkan metode P2PNet-Soy untuk menghitung dan melokalisasi biji kedelai secara otomatis, menggantikan cara manual yang lambat dan tidak akurat. Dengan menggabungkan fitur <i>level</i> rendah dan tinggi, <i>atrous convolution</i> , serta <i>attention mechanism</i> , model ini mengurangi <i>mAE</i> dari 105,55

				menjadi 12,94 dan bekerja efektif di lapangan.
4.	Leite, Silva, Uneda-Trevisoli, Silva	Identification of superior genotypes and soybean traits by multivariate analysis and selection index	2018	Penelitian ini bertujuan untuk menyeleksi genotipe kedelai RR unggul menggunakan analisis multivariat (PCA, K-means, Ward) dan indeks seleksi Mulamba & Mock. Dari 227 genotipe F5 (85 tahan glifosat), terpilih 16 genotipe berproduksi tinggi dengan karakter utama seperti jumlah polong dan berat biji. Metode analisis terbukti efektif dan konsisten dalam seleksi.
5.	Abdullah, Oothariasamy	Vehicle Counting Using Deep Learning Models: A Comparative Study	2023	Penelitian ini mengembangkan sistem dalam melakukan deteksi dan perhitungan kendaraan berbasis untuk lalu lintas perkotaan. Penelitian ini menggunakan <i>TensorFlow</i> dengan pendekatan <i>transfer learning</i> . Metode yang digunakan yaitu YOLOv3, DarkNetv19, Faster R-CNN, ResNet 101, dan SSD Inception. Hasil kinerja dari beberapa model tersebut mencapai 80,90%.
6.	Zhang, Bian, Qu, Tuo, Wang	Time Series Forecast of Sales Volume Based on XGBoost	2021	Penelitian ini mengusulkan pendekatan prediktif untuk memperkirakan <i>volume</i> penjualan di toko fisik menggunakan analisis deret waktu dan rekayasa fitur. Dengan mempertimbangkan variabel eksternal seperti cuaca dan suhu, model XGBoost dipilih sebagai algoritma utama karena

				keunggulannya dalam menangani data non-linear serta efisiensi dalam iterasi. Hasil eksperimen pada dua dataset menunjukkan bahwa XGBoost mencapai performa terbaik dengan <i>RMSE</i> dan <i>MAE</i> terendah yaitu 0,22 pada dataset pertama dan 0,06 pada dataset kedua.
7.	Migqi Li, Fu, Dondong Li	Migqi Li, Fu, Dondong Li	2020	Penelitian ini mengusulkan penggunaan algoritma XGBoost dengan kombinasi fitur numerik dan fitur teks yang diekstraksi dari data eksperimen. Pendekatan <i>machine learning</i> menjadi solusi yang lebih efektif dalam mendeteksi pola tersembunyi dalam data kesehatan. Penelitian ini melakukan perbandingan beberapa model seperti Random Forest, Support Vector Machine, Decision Tree, Naïve Bayes, Logistic Regression, dan BP Neural Network untuk prediksi penyakit. Dalam penelitian ini, model XGBoost terbukti memiliki keunggulan signifikan dibandingkan metode lain, dengan akurasi mencapai 81,2% dan waktu operasi yang jauh lebih singkat dibandingkan algoritma lainnya.
8.	Kurniawan	Comparison of AlexNet and ResNet50 Model	2025	Penelitian ini membahas penggunaan dua arsitektur CNN, AlexNet dan ResNet50, untuk klasifikasi citra makanan

		Performance in Classifying Images of Indonesian Traditional Food		tradisional Indonesia. Hasil menunjukkan ResNet50 unggul di semua metrik dengan akurasi rata-rata 92%, melampaui AlexNet (86%), sehingga ResNet50 dinilai lebih optimal untuk tugas ini.
9.	Datta, Meghla, Khatun, Bhuiya, Shuvo, Rahman	Road Object Detection in Bangladesh Using Faster R-CNN: A Deep Learning Approach	2020	Penelitian ini membahas tentang pentingnya deteksi objek dalam kehidupan sehari-hari, terutama dalam mobil otonom, sistem bantuan pengemudi cerdas, dan analisis lalu lintas. Peneliti menggunakan Faster R-CNN yang dilatih untuk 19 kelas objek. Hasilnya, model ini mencapai akurasi deteksi sebesar 86,42% dalam analisis video <i>real-time</i> . Selain itu, <i>False Positive Rate</i> (FPR) dan <i>False Negative Rate</i> (FNR) dihitung untuk evaluasi, dengan hasil <i>FPR</i> sebesar 15,97% dan <i>FNR</i> sebesar 12,2%. Hasil ini menunjukkan bahwa Faster R-CNN efektif untuk deteksi objek dalam sistem bantuan pengemudi cerdas dan analisis lalu lintas.
10.	Hakim, Hendrawan, dan Khoiriyah	Traffic Vehicle Detection Using Faster R-CNN: A Comparative Analysis of Backbone Architectures	2024	Penelitian ini menerapkan Faster R-CNN untuk deteksi enam jenis kendaraan dan membandingkan beberapa <i>backbone</i> , ResNet50 memberikan kinerja terbaik dengan mAP 96,6%, melampaui ResNet50V2 dan MobileNetV3, sementara MobileNetV3 Large 320 menghasilkan nilai terendah.

2.16 Perbedaan Penelitian

Penelitian ini memiliki perbedaan dengan penelitian-penelitian sebelumnya dalam beberapa aspek, terutama dalam metode yang digunakan dan fokus penelitian. Pada penelitian yang dilakukan oleh Tonael *et al.* (2021) yaitu berfokus pada seleksi fitur benih padi unggul dengan tujuan pengoptimalan hasil panen padi berkualitas tinggi, metode yang digunakan adalah *Color Moment* dan GLCM untuk menganalisis karakteristik visual benih. Klasifikasi menggunakan algoritma KNN untuk menentukan kelas benih dan SVM untuk memisahkan dua kelas dengan *hyperplane* terbaik.

Penelitian yang dilakukan oleh Hanafiah *et al.* (2023) bertujuan untuk mengembangkan varietas unggul kedelai dengan melakukan teknik persilangan tiga arah menggunakan varietas Grobogan, Anjasmoro, dan Dega-1 yang lebih berfokus pada pengamatan manual terhadap hasil jumlah polong/biji, bobot biji tanpa penerapan model *AI*.

Penelitian yang dilakukan oleh Zhao *et al.* (2023) melakukan pengembangan alat penghitungan otomatis biji kedelai (P2PNet-Soy) bertujuan untuk memprediksi hasil panen dan meningkatkan efisiensi seleksi dalam program pemuliaan. P2PNet-Soy menggunakan beberapa strategi, seperti penggabungan hasil penghitungan berlebih dengan *unsupervised*.

Penelitian yang dilakukan oleh Leite *et al.* (2018) ini bertujuan untuk menyeleksi genotipe kedelai RR unggul menggunakan analisis multivariat (PCA, K-means, Ward) dan indeks seleksi Mulamba & Mock. Dari 227 genotipe F5 (85 tahan glifosat), dengan menggunakan karakter atau parameter evaluasi yaitu umur panen, tinggi tanaman, ketahanan rebah, nilai argonomi, jumlah cabang, jumlah polong, berat 100 biji, dan hasil biji yang dihasilkan, terpilih 16 genotipe berproduksi tinggi dengan karakter utama seperti jumlah polong dan berat biji.

Penelitian yang dilakukan oleh Abdullah dan Oothariasamy (2020) dimana penelitian ini berfokus untuk mengembangkan sistem pada melakukan deteksi dan perhitungan kendaraan berbasis untuk lalu lintas perkotaan. Penelitian ini menggunakan metode YOLOv3, DarkNetv19, Faster R-CNN, ResNet 101, dan SSD Inception dengan *TensorFlow* dengan pendekatan *transfer learning*.

Penelitian yang dilakukan oleh Zhang *et al.* (2021) mengusulkan pendekatan prediktif untuk memperkirakan *volume* penjualan di toko fisik menggunakan analisis deret waktu dan rekayasa fitur. Dengan mempertimbangkan variabel eksternal seperti

cuaca dan suhu dengan membandingkan model XGBoost, GBDT, LSTM, Prophet, dan ARIMA.

Penelitian yang dilakukan oleh Li *et al.* (2020) ini mengusulkan penggunaan algoritma *XGBoost* dengan kombinasi fitur numerik dan fitur teks yang diekstraksi dari data eksperimen untuk membantu deteksi dini diabetes untuk meningkatkan efektivitas pencegahan penyakit kronis.

Pada Penelitian yang dilakukan oleh Kurniawan *et al.* (2025) membandingkan dua arsitektur *convolutional neural network* yaitu AlexNet dan ResNet50 yang berfokus untuk melakukan klasifikasi terhadap makanan tradisional.

Pada peneliain ini memiliki kemiripan metode yang digunakan yaitu pada penelitian yang dilakukan oleh Datta *et al.* (2020) dimana pada penelitian tersebut menggunakan metode Faster R-CNN untuk melakukan deteksi objek dalam kehidupan sehari-hari, terutama dalam mobil otonom, sistem bantuan pengemudi cerdas, dan analisis lalu lintas dengan menggunakan video *real-time*.

Penelitian yang dilakukan oleh Hakim *et al.* (2024) membahas tentang penerapan metode Faster R-CNN dengan membandingkan arsitektur ResNet50, ResNet50V2, MobileNetV3 Large, dan MobileNetV3 large 320 untuk mendeteksi enam kategori kendaraan seperti bus, mobil, motor, mobil *pick up*, truk, dan truk kotak di jalan.

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Data yang digunakan

Data yang digunakan dalam penelitian ini terdiri atas citra polong kedelai dan data morfologi tanaman. Citra diperoleh dari tanaman kedelai yang ditanam di lahan pertanian Universitas Sumatera Utara (Pasar 8, Jl. Barokah, Medan Selayang) pada Januari 2025, serta dilengkapi citra dari penelitian yang dilakukan oleh Jiang *et al.* (2025) untuk memperkaya variasi data. Data morfologi yang dikumpulkan oleh peneliti di Fakultas Pertanian USU. Seluruh sampel merupakan hasil persilangan tiga varietas unggul kedelai, yaitu Grobogan, Dega-1, dan Anjasmoro.

3.1.1 Data Citra Polong Kedelai

Citra polong kedelai yang dikumpulkan secara langsung di lahan pertanian Universitas Sumatera Utara tersebut diambil menggunakan iPhone 11 dengan resolusi kamera 12 MP. Pengambilan citra tersebut dilakukan di dalam dan diluar ruangan, untuk menambah variasi sudut dan mengurangi *overfitting* setiap citra polong yang sudah dipetik diambil sebanyak empat kali dari berbagai sisi (He *et al.*, 2023) pada pencahayaan yang seragam, citra polong pada tanaman dan citra polong yang sudah dipetik diambil dari jarak 30 cm (Zhang *et al.*, 2023), dan polong per biji diambil dari jarak 10 cm.



(a)



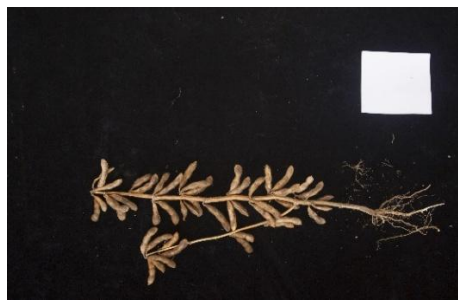
(b)



(c)

Gambar 3.1 (a) Polong pada tanaman, (b) Polong yang sudah dipetik, (c) Polong per biji (1, 2, dan 3)

Untuk memperkaya variasi data, penelitian ini turut memanfaatkan 130 citra dari penelitian yang dilakukan oleh Jiang *et al.* (2025). Citra tersebut menampilkan polong kedelai dengan pencahayaan yang sama dan latar belakang gelap untuk meningkatkan kontras antara objek polong dan latar belakang.



(a)



(b)

Gambar 3.2 (a) Polong pada Tanaman, (b) Polong yang sudah dipetik

Citra polong kedelai yang digunakan dalam penelitian ini berfokus pada penelitian kedelai di Universitas Sumatera Utara. Setiap citra yang digunakan telah melalui proses anotasi untuk mengidentifikasi polong. Data yang terkumpul mencakup jumlah biji pada polong kedelai. Rincian jumlah data yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.1 dan Tabel 3.2.

Tabel 3.1 Rincian Jumlah Instance tiap Biji pada Polong

Kelas Polong	Jumlah
Polong Biji 1	1812
Polong Biji 2	4150
Polong Biji 3	4833
Total	10795

Tabel 3.2 Rincian Jumlah Data Citra Polong per-tipe Pengambilan Gambar

Tipe Pengambilan	Jumlah
Polong pada tanaman	159
Polong setelah dipetik	111
Polong per-biji (1, 2 dan 3)	165, 142, 138
Total	715

3.1.2 Data Morfologi Kedelai

Data kuantitatif atau data tabular berisi tentang data morfologi dan produksi yang dihasilkan dari tanaman kedelai yang diambil dari Lab Pertanian Universitas Sumatera Utara.

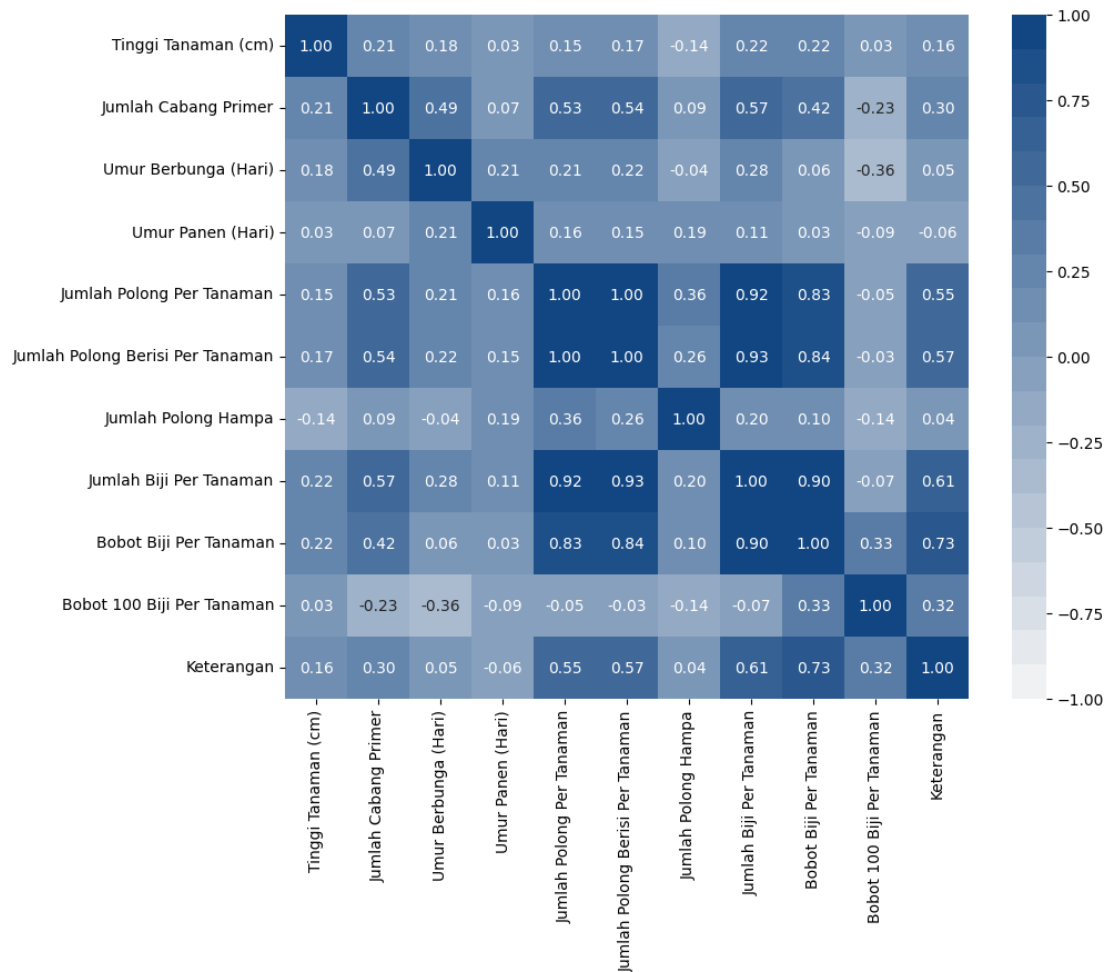
Kode Tanaman	Tinggi Tanaman	Jumlah Cabang Primer	Umur Berbunga	Umur Panen	Jumlah Polong Per Tanaman	Jumlah Polong Berisi Per	Jumlah Polong Hampa	Jumlah Biji Per Tanaman	Bobot Biji Per Tanaman	Bobot 100 Biji Per Tanaman	Keterangan
F5.2.1/1	37.5	1	29	77	38	36	2	54	12.76	23.62962963	Tidak Terpilih
F5.2.1/4	55.4	2	29	77	38	37	1	62	11.6	18.70967742	Tidak Terpilih
F5.2.1/5	50.2	2	29	77	37	37	0	54	11.1	20.55555556	Tidak Terpilih
F5.2.2/1	54.8	2	28	78	41	41	0	78	16.94	21.71794872	Tidak Terpilih
F5.2.2/3	64.3	1	28	78	40	39	1	90	20.35	22.61111111	Tidak Terpilih
F5.9.3/2	50.1	2	29	77	37	37	0	85	18.85	22.17647059	Tidak Terpilih
F5.16.3/1	52	2	28	77	46	45	1	100	23.42	23.42	Tidak Terpilih
F5.16.3/2	53.2	2	29	77	40	40	0	83	19.7	23.73493976	Tidak Terpilih
F5.16.3/4	48.3	2	28	77	49	49	0	100	19.96	19.96	Tidak Terpilih
F5.16.3/5	52.5	2	28	77	58	58	0	114	23.82	21.6	Tidak Terpilih
F5.16.4/3	61.1	1	29	78	36	36	0	77	19.49	25.31168831	Tidak Terpilih
F5.16.4/4	53.9	3	28	77	51	51	0	111	23.9	21.2	Tidak Terpilih
F5.16.4/5	59.4	1	28	78	46	46	0	94	22.6	24.04255319	Tidak Terpilih
F5.2.3/4	54.3	1	28	77	40	40	0	74	18.5	25	Tidak Terpilih
F5.2.4/2	74.2	1	28	77	42	42	0	88	22.03	25.03409091	Tidak Terpilih
F5.2.4/3	61.4	3	28	78	45	45	0	54	12.41	22.98148148	Tidak Terpilih
F5.2.4/4	58.8	3	29	77	43	40	3	79	18.34	23.21518987	Tidak Terpilih
F5.9.5/1	49.3	3	27	77	53	53	0	101	22.53	22.43	Tidak Terpilih
F5.9.5/2	58.1	2	28	78	47	47	0	101	22.6	22.43	Tidak Terpilih
F5.9.5/4	55.2	1	29	78	32	31	1	64	13.5	21.09375	Tidak Terpilih
F5.9.5/5	48	1	28	77	40	40	0	76	15.38	20.23684211	Tidak Terpilih
F5.9.6/1	45.3	2	29	78	35	35	0	67	13.98	20.86567164	Tidak Terpilih
F5.9.6/3	48	1	28	78	27	27	0	54	11.54	21.37037037	Tidak Terpilih
F5.16.5/1	45	3	29	77	43	43	0	100	23.47	23.47	Tidak Terpilih
F5.16.5/2	52.7	2	29	78	43	42	1	92	19.6	21.30434783	Tidak Terpilih
F5.16.5/4	54.2	1	29	77	48	48	0	100	19.46	19.46	Tidak Terpilih

Gambar 3.3 Data Kuantitatif Morfologi Kedelai

Data kuantitatif yang berisi morfologi tanaman yang digunakan dalam penelitian kedelai untuk menentukan tanaman tersebut tergolong terpilih atau tidak terpilih. Setiap data yang digunakan melalui proses *labelling* oleh peneliti pertanian di Universitas Sumatera Utara. Rincian jumlah data yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.3.

Tabel 3.3 Rincian Jumlah Data

Kategori	Jumlah
Terpilih	281
Tidak Terpilih	272
Total	553



Gambar 3.4 *Correlation Matriks*

Terlihat pada gambar diatas, setiap parameter memiliki tingkat keterkaitan yang berbeda terhadap keterangan (Terpilih atau Tidak Terpilih). Bobot Biji per Tanaman menunjukkan korelasi tinggi $r=0,73$, sehingga menjadi indikator paling kuat bahwa tanaman tersebut Terpilih saat bobot bijinya besar. Jumlah Biji per Tanaman juga kuat ($r=0,61$), diikuti Jumlah Polong Berisi per Tanaman ($r=0,57$), dan Jumlah Polong per Tanaman ($r=0,55$). Keempatnya mencerminkan komponen hasil panen untuk menentukan status seleksi. Bobot 100 Biji per Tanaman berkorelasi $r=0,32$, Jumlah Cabang Primer $r=0,30$, dan Tinggi Tanaman $r=0,16$ sebagai tambahan informasi, sementara Jumlah Polong Hampa $r=0,04$ serta Umur Berbunga $r=0,05$ hampir tidak berpengaruh. Umur Panen bahkan sedikit negatif ($r=-0,06$), menandakan kecenderungan sangat kecil bahwa tanaman yang penennya lebih cepat menjadi tanaman terpilih. Secara keseluruhan, parameter yang paling informatif adalah yang berhubungan langsung dengan kuantitas dan bobot hasil seperti Jumlah Polong per

Tanaman, Jumlah Polong Berisi per Tanaman, Jumlah Biji per Tanaman, dan Bobot Biji per Tanaman.

3.2 Analisis Sistem

Pada penelitian ini terdapat beberapa tahapan yang harus dilalui, yang mencakup *data acquisition*, *image processing*, *data counting*, *classification*, *model evaluation*, dan pada tahap akhir akan menghasilkan *output* berupa jumlah polong berbiji 1, 2, 3, jumlah polong, jumlah biji, dan prediksi tanaman tersebut merupakan tanaman terpilih atau tidak terpilih. Tahap pertama penelitian dimulai dari *data acquisition* yang merupakan proses pengambilan data citra polong kedelai dan data tabular morfologi tanaman kedelai yang akan dipakai dalam penelitian. Data tersebut dikumpulkan dari lahan Lahan Pertanian Universitas Sumatera Utara.

Pada data *counting*, proses yang dilakukan adalah *image pre-processing* yang meliputi *labeling*, *resizing*, dan *augmentation*. *Labeling* bertujuan untuk memberi label pada gambar sesuai dengan kategori seharusnya. Setelah itu, proses *resizing* bertujuan untuk mengubah ukuran gambar menjadi seragam dalam satuan piksel. Terakhir, *augmentation* digunakan untuk memperbanyak variasi data dengan melakukan transformasi pada gambar.

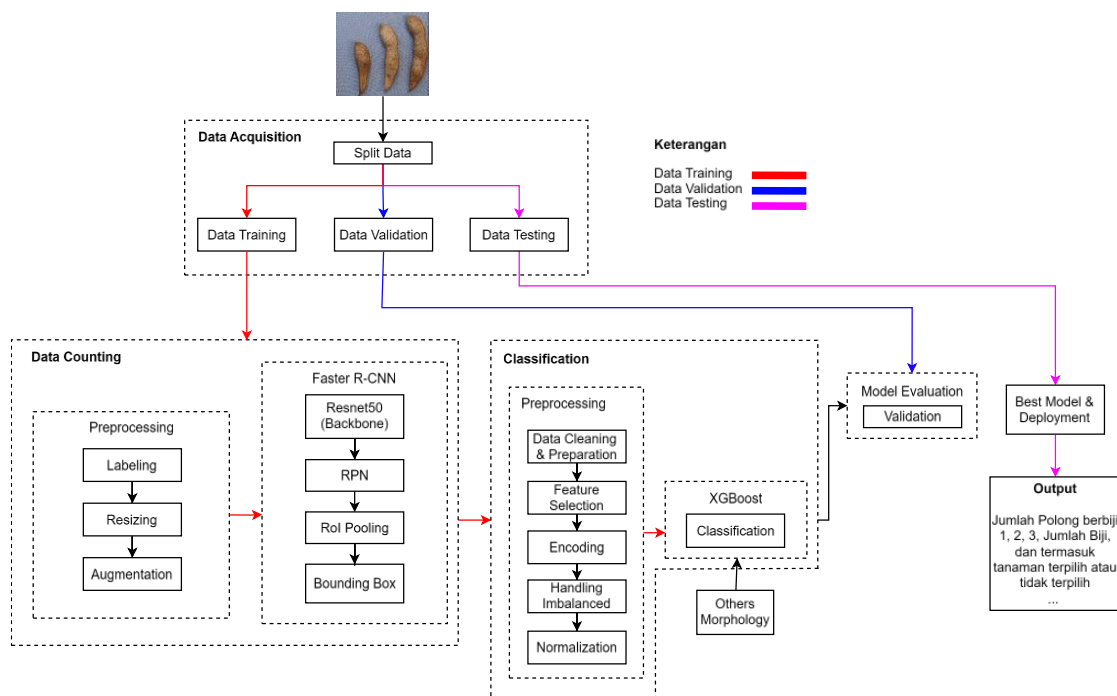
Setelah tahap *pre-processing* selesai, tahap berikutnya adalah objek deteksi, yaitu pengambilan citra polong kedelai dari berbagai varietas yang diteliti pada fase generatif. Data citra yang digunakan dikumpulkan baik dari lahan pertanian Universitas Sumatera Utara. Hasil dari tahap ini adalah informasi jumlah polong berbiji 1, 2, 3 dan jumlah biji.

Selanjutnya adalah *classification*, proses yang dilakukan adalah data *pre-processing* yang meliputi *data cleaning and preparation*, *feature selection*, *encoding*, *handling imbalanced*, dan *normalization*. *Data cleaning and preparation* bertujuan untuk membersihkan data, melakukan penamaan, dan konversi tipe data agar sesuai dengan data. *Feature selection* bertujuan untuk memilih fitur atau parameter morfologi mana yang benar-benar penting. *Encoding* bertujuan untuk mengubah label target menjadi format numerik agar dapat dipahami oleh algoritma *machine learning*. *Handling imbalanced* bertujuan untuk mengatasi ketidakseimbangan data pada setiap kelas dalam dataset. Setelah itu, proses *normalization* bertujuan untuk mengubah skala

fitur agar model dapat belajar lebih efisien sehingga tidak ada fitur yang mendominasi karena memiliki skala nilai yang lebih besar.

Setelah tahap *pre-processing* selesai, tahap berikutnya adalah klasifikasi yaitu melakukan klasifikasi dari parameter morfologi tanaman kedelai yaitu: Tinggi Tanaman (cm), Jumlah Cabang Primer, Jumlah Polong, Jumlah Polong Berisi, Jumlah Biji, Bobot Biji, dan Bobot 100 Biji. Hasil dari tahap klasifikasi ini adalah prediksi tanaman tersebut merupakan tanaman yang terpilih atau tidak terpilih.

Tahap selanjutnya adalah *model evaluation*. Dalam tahap ini, dilakukan evaluasi terhadap hasil yang dihasilkan oleh model untuk memastikan bahwa model dapat melakukan deteksi dan kuantifikasi polong kedelai seleksi tanaman kedelai dengan baik. Evaluasi dilakukan menggunakan metrik seperti *Mean Average Precision (mAP)*, *Intesection over Union (IoU)* untuk objek deteksi dan *Accuracy*, *Precision*, dan *Recall* untuk klasifikasi, metrik tersebut berfungsi untuk menilai seberapa baik model dapat memberikan prediksi. Hasil dari semua proses yang telah dilakukan akan menghasilkan informasi jumlah polong berbiji 1, 2, 3, jumlah polong, jumlah biji, dan prediksi tanaman tersebut merupakan tanaman terpilih atau tidak terpilih. Arsitektur umum penelitian dapat dilihat pada Gambar 3.5.



Gambar 3.5 Arsitektur Umum Penelitian

3.3 Data Acquisition

Image acquisition merupakan tahap pertama yaitu tahap pengumpulan data berupa citra polong kedelai dan parameter morfologi tanaman kedelai pada fase generatif.

3.3.1 *Faster R-CNN*

Data citra yang digunakan adalah citra polong pada tanaman, polong setelah dipetik, dan polong per biji (1, 2 dan 3). Data dipisahkan menjadi tiga kelompok yaitu *data training* sebanyak 561 data, *data validation* sebanyak 76 data, dan *data testing* sebanyak 76 data. Data Polong kedelai dikumpulkan menggunakan iPhone 11 dengan kamera 12 MP yang menangkap gambar. Seluruh citra yang digunakan dalam penelitian ini memiliki format file .png.

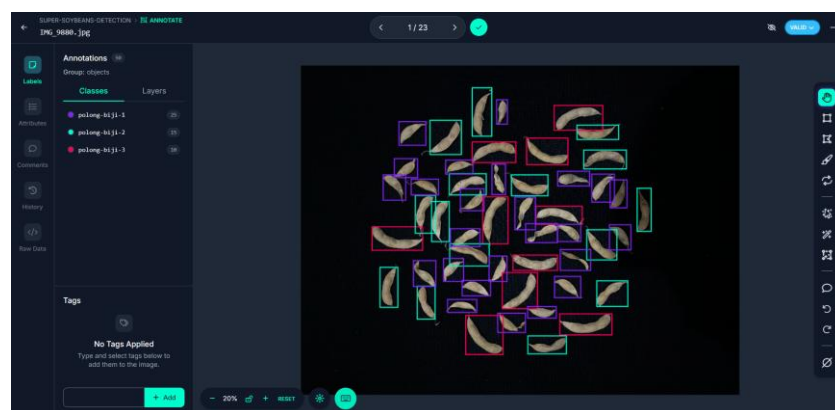
3.3.2 *XGBoost*

Data morfologi dikumpulkan oleh peneliti pertanian di Lab Pertanian Universitas Sumatera. Data morfologi yang digunakan adalah tinggi tanaman, jumlah cabang primer, jumlah polong, jumlah polong berisi, jumlah biji, bobot biji, bobot 100 biji. Data dipisahkan menjadi dua kelompok yaitu *data training* sebanyak 442 data dan *data testing* sebanyak 111 data. Data morfologi yang digunakan dalam penelitian ini memiliki format .csv.

3.4 Image Pre-preprocessing

1. Labelling

Tahap ini bertujuan untuk mengelompokkan polong seperti (Polong biji 1, 2, dan 3). Dengan melakukannya, dapat memberikan informasi kepada model untuk mengidentifikasi objek target pada gambar. Proses pembuatan anotasi pada citra dapat dilihat pada Gambar 3.6.



Gambar 3.6 Proses Anotasi Pada Citra

Sementara itu file anotasi akan mengandung informasi yang diperlukan untuk proses training dan validation pada objek, yang berupa koordinat anotasi dalam bentuk *bounding box* yang terdiri dari titik x1, x2, y1, dan y2. Format yang digunakan pada penelitian ini yaitu *file* dengan ekstensi json. Informasi yang terdapat file .json pada Gambar 3.7.

```
{
  "id": 560,
  "license": 1,
  "file_name": "Biji-3-Derap-1-2-.jpg.rf.b7ae042496cf45534a294b7d72c413dc.jpg",
  "height": 4624,
  "width": 3468,
  "date_captured": "2025-10-17T10:50:59+00:00",
  "extra": { "name": "Biji-3-Derap-1-2-.jpg" }
},
"annotations": [
{
  "id": 1,
  "image_id": 0,
  "category_id": 3,
  "bbox": [560, 1604, 276.43, 143.15],
  "area": 39570.955,
  "segmentation": [],
  "iscrowd": 0
},

```

Gambar 3.7 Struktur File Json dari Hasil Anotasi

2. Resizing

Mengubah ukuran gambar ke dimensi yang konsisten untuk memastikan bahwa semua gambar memiliki dimensi yang sama. Melalui tahapan ini citra akan memiliki ukuran yang sama sehingga dapat diproses dengan baik. Dalam penelitian ini gambar diatur agar menjadi ukuran 800 x 1333.

```
FUNCTION resize_images_in_folder(input_folder, output_folder, target_size)
  IF not exists(output_folder)DO
    CREATE_DIRECTORY(output_folder)
  files = GET_FILES_IN_FOLDER(input_folder)
  FOR EACH file IN files DO
    image = READ_IMAGE(file)
    IF image is not NULL DO
      resized_image = RESIZE_IMAGE(image,
        target_size)
      SAVE_IMAGE(resized_image, output_folder,file)
    ENDIF
  END FOR
END FUNCTION
input_folder = "path/to/input/folder"
output_folder = "path/to/output/folder"
```



```
target_size = (800, 1333)
resize_images_in_folder(input_folder, output_folder, target_size)
```

3. Augmentation

Menerapkan teknik seperti *flip*, *brightness*, *sharpen* untuk memperkaya dataset dan mengurangi risiko *overfitting* dengan memberikan variasi data yang banyak sehingga dapat membantu model memahami dan belajar informasi penting dalam gambar. Berikut adalah *pseudocode* untuk *augmentation flip horizontal*.

```
FUNCTION flip_horizontal(image)
    height, width = get_dimensions(image)
    flipped_image = create_empty_image(height, width)
    FOR y = 0 TO height - 1 DO
        FOR x = 0 TO width - 1 DO
            flipped_image[y][width - 1 - x] = image[y][x]
        END FOR
    END FOR
    RETURN flipped_image
END FUNCTION

original_image = load_image('path/to/xray_image.jpg')
flipped_image = flip_horizontal(original_image)
SAVE_IMAGE(flipped_image, 'path/to/save/flipped_image.jpg') target_size =
    (800, 1333)
resize_images_in_folder(input_folder, output_folder, target_size)
```

Pada penelitian ini menggunakan *vertical flip*, berikut adalah *pseudocode vertical flip*.

```
FUNCTION flip_vertical(image)
    height, width = get_dimensions(image)
    flipped_image = create_empty_image(height, width)
    FOR y = 0 TO height - 1 DO
        FOR x = 0 TO width - 1 DO
            flipped_image[height - 1 - y][x] = image[y][x]
        END FOR
    END FOR
    RETURN flipped_image
```

```
END FUNCTION
```

```
original_image = load_image('path/to/polong_image.jpg')
flipped_image = flip_vertical(original_image)
SAVE_IMAGE(flipped_image, 'path/to/save/flipped_image_vertical.jpg')
target_size = (800, 1333)
resize_images_in_folder(input_folder, output_folder, target_size)
```

Pada penelitian ini menggunakan *sharpen* untuk meningkatkan kontras tepi pada gambar, berikut adalah *pseudocode sharpen*.

```
FUNCTION Sharpen(p, T):
  FOR EACH probability p_i IN p:
    p_i_sharpened = p_i ^ (1 / T)
  END FOR

  sum_p_sharpened = SUM(p_i_sharpened EACH p_i_sharpened IN p_sharpened)

  FOR EACH probability p_i_sharpened IN p_sharpened:
    p_i_sharpened = p_i_sharpened / sum_p_sharpened
  END FOR

  RETURN p_sharpened
END FUNCTION
```

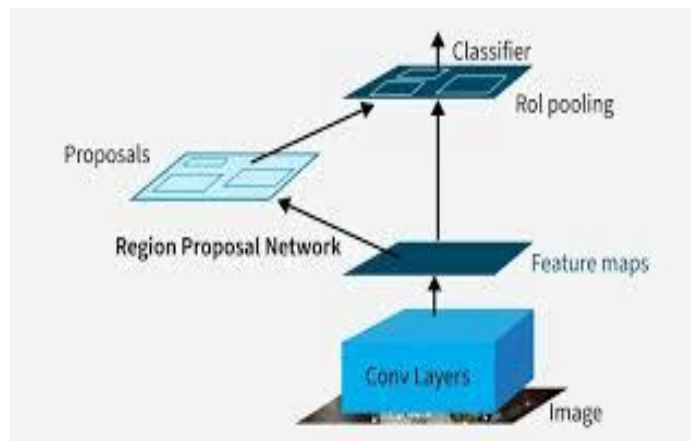
Pada penelitian ini juga menggunakan *random brightness* untuk meningkatkan kontras tepi pada gambar, Berikut adalah *pseudocode random brightness*.

```
FUNCTION RandomBrightnessAugmentation(Image input_image, Float min_factor,
  Float max_factor)
  brightness_factor = RandomUniform(min_factor, max_factor)
  IF using_library_enhancer THEN
    augmented_image = ApplyBrightnessEnhancement(input_image, brightness_factor)
  ELSE
    augmented_image = ConvertImageToFloat(input_image)
    augmented_image = augmented_image * brightness_factor
    augmented_image = Clip(augmented_image, min_value, max_value)
    augmented_image = ConvertImageBackToOriginalType(augmented_image)
  END IF
  RETURN augmented_image
```

END FUNCTION

3.5 Faster R-CNN

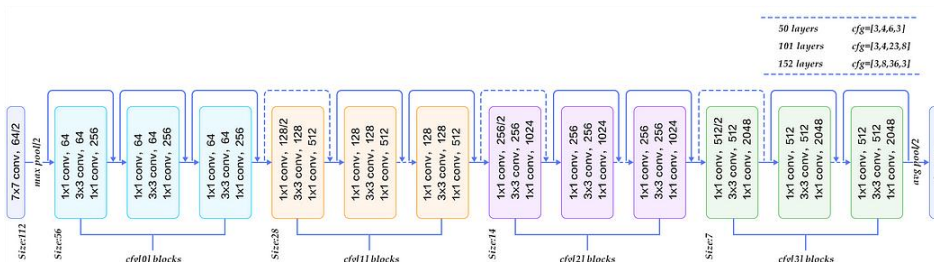
Setelah data citra melalui *pre-processing*, data tersebut akan diproses agar model dapat mengenali polong dengan menggunakan algoritma objek deteksi yang diajukan pada penelitian ini yakni Faster R-CNN. Tahapan pada proses objek deteksi dapat dilihat pada Gambar 3.8.



Gambar 3.8 Arsitektur Faster R-CNN

1. Resnet 50 (Backbone)

Pada tahapan ini, citra dikonvolusi menggunakan jaringan *backbone* ResNet-50 untuk mengekstraksi fitur yang informatif. ResNet-50 terdiri dari 50 lapisan dengan struktur bertingkat yang terbagi dalam beberapa tahap utama, antara lain conv1, conv2_x, conv3_x, conv4_x, dan conv5_x. Setiap tahap dibangun dari blok residual bertipe *bottleneck* (1x1, 3x3, 1x1) dengan koneksi *shortcut* yang menjaga aliran gradien sehingga pelatihan jaringan tetap stabil. Rangkaian konvolusi tersebut menghasilkan *feature map multi-level* yang memadukan informasi spasial dan semantik. Arsitektur ResNet-50 dapat dilihat pada Gambar 3.9.

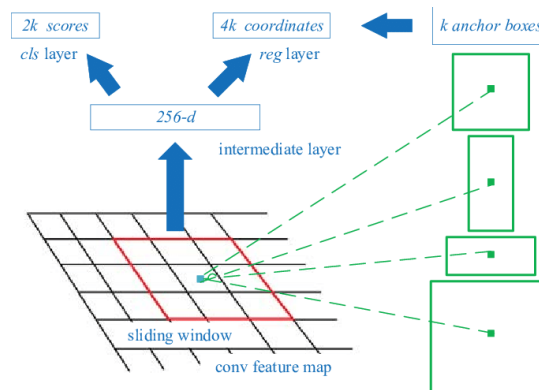


Gambar 3.9 Arsitektur ResNet-50

Seperti yang terlihat pada Gambar 3.9, setelah konvolusi awal 7x7 (stride 2) dan max-pooling 3x3 (stride 2), fitur melewati empat tahap blok residual dengan konfigurasi jumlah blok 3-4-6-3. Di awal setiap tahap (kecuali conv2_x) diterapkan *downsampling* (stride 2) untuk menurunkan resolusi spasial, sementara jumlah stride atau kanal meningkat guna untuk memperkaya konten semantik. Hasil akhirnya adalah peta fitur yang lebih padat secara semantik.

2. RPN (Region Proposal Network)

Region Proposal Network adalah komponen *neural network* yang berfungsi untuk mengidentifikasi dan mengusulkan *region-region* potensial dalam gambar yang kemungkinan mengandung objek target. RPN berkeja dengan membuat *anchor boxes* pada setiap lokasi dalam *feature map* dengan berbagai ukuran aspek, kemudian melakukan klasifikasi yaitu memprediksi ada atau tidaknya objek dan regresi *bounding box* (menyesuaikan koordinat *anchor*) untuk setiap anchor secara paralel dan akhirnya menerapkan *Non-Maximum Suppression* (NMS) untuk mengeliminasi kandidat yang saling tumpang tindih sehingga menghasilkan proposal yang berkualitas yang akan diproses lanjut oleh deteksi berikutnya. Arsitektur RPN dapat dilihat pada Gambar 3.10.



Gambar 3.10 Arsitektur *Region Proposal Network* (RPN)

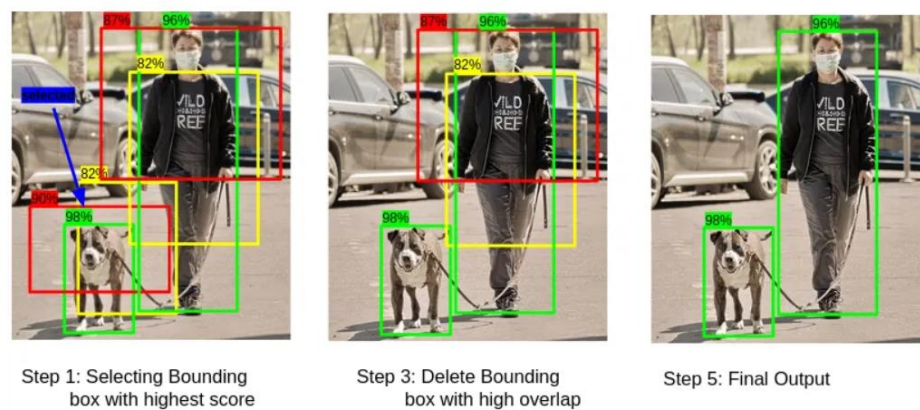
3. RoI Align

Setelah itu, *Region of Interest* (RoI) *Align* yang berfungsi untuk menyempurnakan ekstraksi fitur dari *region* yang diusulkan untuk meningkatkan akurasi model. Proses ini berfungsi untuk menjaga informasi

penting pada gambar dengan baik dari *input feature map* untuk setiap *region proposal*.

3. Bounding Box

Bounding box atau juga dikenal sebagai *bounding volume* atau daerah pembatas adalah bentuk geometris yang melingkupi atau mengelilingi suatu objek atau sekelompok objek dalam citra digital. *Bounding box* dalam gambar 2D umumnya direpresentasikan sebagai kotak persegi panjang dengan sisi-sisi panjangnya sejajar dengan sumbu x dan y gambar. Kotak persegi panjang tersebut menandai dan menunjukkan lokasi serta ukuran suatu objek dalam gambar.



Gambar 3.11 *Bounding Box*

3.5.1 Hyperparameter Tuning

Setelah model selesai dibuat, selanjutnya dilakukan *training* model menggunakan data *training* yang sudah disiapkan. Terdapat beberapa konfigurasi model dan optimasi *hyperparameter* lainnya agar dapat meminimalkan *loss* dan meningkatkan performa dari model. Setelah proses *training* selesai, model yang diproses akan diekspor dan nantinya akan digunakan untuk melakukan prediksi objek. Parameter yang akan dilibatkan dalam proses *hyperparameter tuning* antara lain *batch size*, *epoch*, *learning rate*, dan *weight decay*.

1. Batch size

Batch size merupakan jumlah sampel yang diproses sebelum bobot model diperbaharui selama pelatihan. *Batch size* yang kecil menggunakan memori lebih sedikit, sedangkan *batch size* besar membutuhkan lebih banyak memori. Dalam penelitian yang dilakukan oleh Cai *et al.* (2018), menggunakan *batch*

size tertinggi yaitu sebesar 32. Oleh karena itu pada penelitian ini, akan dilakukan *hyperparameter tuning* dengan nilai batch size sebesar 1, 2, 4, 8, 16 dan 32 untuk melihat *batch size* yang paling sesuai dengan data.

2. *Learning rate*

Learning rate merupakan parameter dalam *machine learning* yang digunakan untuk menghitung nilai koreksi bobot dalam proses pelatihan. Nilai *learning rate* antara 0 hingga 1, dimana semakin tinggi nilainya maka model dapat konvergen dengan cepat dan hasilnya tidak akurat dan bila nilainya terlalu rendah maka proses latihan akan berlangsung lama sebelum konvergen dan menghasilkan *overfitting*. Berdasarkan penelitian yang dilakukan oleh Panda (2019), melakukan penelitian tentang *object detection tracking* menggunakan Faster R-CNN dengan menggunakan *learning rate* sebesar 0,005 dan penelitian yang dilakukan oleh Chen *et al.* (2018) tentang *object detection in the wild* menggunakan Faster R-CNN dengan menggunakan *learning rate* sebesar 0,001.

3. *Weight decay*

Weight decay merupakan teknik regularisasi yang digunakan untuk mencegah *overfitting* dengan menambahkan penalti terhadap bobot model yang terlalu besar. *Weight decay* bekerja dengan menambahkan nilai bobot ke dalam fungsi *loss*, sehingga model terdorong untuk menghasilkan bobot yang lebih kecil dan tidak terlalu kompleks. Nilai *weight decay* yang terlalu besar dapat menyebabkan model mengalami *underfitting*, sedangkan nilai yang terlalu kecil dapat menyebabkan *overfitting*. Dalam penelitian yang dilakukan oleh Cheng *et al.* (2018), digunakan nilai *weight decay* sebesar 0,0001 untuk pelatihan Faster R-CNN. Oleh karena itu, dalam penelitian ini akan dilakukan *hyperparameter tuning* dengan mencoba nilai *weight decay* sebesar 0,0001 untuk mengevaluasi pengaruhnya terhadap performa model dalam tugas segmentasi.

2. *Momentum*

Momentum adalah teknik optimasi dalam *machine learning* yang membantu model belajar lebih cepat dan stabil dengan cara mengingat arah pergerakan sebelumnya. Dalam *training neural network*, tanpa momentum, *optimizer* akan mengupdate parameter (*weights*) hanya berdasarkan gradient saat ini, yang bisa

menyebabkan pergerakan zig-zag atau terjebak di lokal minimal (titik minimum lokal yang bukan solusi terbaik). Berdasarkan penelitian yang dilakukan oleh Bejo *et al.* (2024), melakukan penelitian tentang *planthopper detection and counting* menggunakan Faster R-CNN menggunakan *momentum* sebesar 0,9. Oleh karena itu, dalam penelitian ini akan dilakukan *hyperparameter tuning* dengan nilai *momentum* 0,9.

3.5.2 Training Process

Tahap berikutnya adalah melatih model menggunakan data yang sudah disiapkan. Proses *training* data dilakukan dengan menggunakan variasi *batch size* 1, 2, 4, 8 dan 16, *learning rate* 0,001 dan 0,005, *momentum* 0,9, dan *weight decay* 0.0001. Keseluruhan skenario pelatihan dapat dilihat pada Tabel 3.4.

Tabel 3.4 Skenario *Training*

No	Nama	Batch Size	Learning rate	Weight decay
1	Model 1	1	0.001	0.0001
2	Model 2	1	0.005	0.0001
3	Model 3	2	0.001	0.0001
4	Model 4	2	0.005	0.0001
5	Model 5	4	0.001	0.0001
6	Model 6	4	0.005	0.0001
7	Model 7	8	0.001	0.0001
8	Model 8	8	0.005	0.0001
9	Model 9	16	0.001	0.0001
10	Model 10	16	0.005	0.0001
11	Model 11	32	0.001	0.0001
12	Model 12	32	0.005	0.0001

Hasil dari penelitian tersebut akan menjadi model berbentuk file dengan format .pth dan akan terbentuk beberapa model terbaik. Kinerja model dilihat berdasarkan seberapa baik model tersebut memprediksi objek sesuai dengan anotasi sebenarnya dengan data uji. Berikut merupakan proses pelatihan pada penelitian ini.

1. Install Requirement

Tahap ini adalah proses menginstall *requirement* yang dibutuhkan untuk melatih model. Dalam model ini *requirement* yang dipakai adalah *matplotlib* versi 3.10.0, *opencv-python* versi 4.12.0.00, *pycocotools* versi 2.0.10, *albumentations* versi 2.0.8, *numpy* versi 2.0.2, *torch* versi 2.8.0+cu126, dan *torchvision* versi 0.23.0+cu126. Proses instalasi *requirement* dapat dilihat pada Gambar 3.12.

```
!pip install matplotlib==3.10.0
!pip install opencv-python==4.12.0.88
!pip install pycocotools==2.0.10
!pip install --user albumentations==2.0.8
!pip install numpy=2.0.2
```

Gambar 3.12 *Install Requirement*

2. Model Installation

Langkah berikutnya adalah menginstall algoritma yang akan digunakan dalam penelitian, yaitu Faster R-CNN. Proses model *installation* dapat dilihat pada Gambar 3.13 dan Gambar 3.14

```
!pip torch==2.8.0+cu126
!pip install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu121
!git clone https://github.com/pytorch/vision.git
```

Gambar 3.13 *Model Installation*

```
import torch
from torch.utils.data import Dataset
from torch.utils.data import DataLoader

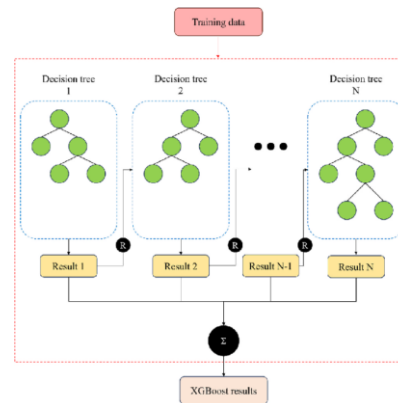
import torchvision
import torchvision.transforms.functional as f
from torchvision.models.detection.faster_rcnn import FastRCNNPredictor
from torchvision.models.detection import FasterRCNN_ResNet50_FPN_Weights
from torchvision.models.detection.rpn import AnchorGenerator
from torchvision.ops import box_iou
from torchvision.ops import nms

import coco_eval
from engine import train_one_epoch
```

Gambar 3. 14 *Import Library*

3.6 XGBoost

Setelah data morfologi melalui proses *pre-processing*, data tersebut akan diproses agar model dapat mengenali tanaman terpilih dan tidak terpilih dengan menggunakan algoritma klasifikasi yang diajukan pada penelitian ini yakni XGBoost.



Gambar 3.15 Arsitektur XGBoost

3.6.1 Hyperparameter Tuning

Setelah model selesai dibuat, selanjutnya dilakukan *training* model menggunakan data *training* yang sudah disiapkan. Terdapat beberapa konfigurasi model dan optimasi *hyperparameter* lainnya agar dapat meningkatkan performa model. Kemudian setelah proses *training* selesai, model yang diproses akan diekspor dan nantinya akan digunakan untuk melakukan klasifikasi tanaman terpilih dan tidak terpilih. Parameter yang akan dilibatkan dalam proses *hyperparameter tuning* antara lain *learning rate*, *n_estimators*, dan *max depth*.

1. Learning rate

Learning rate merupakan parameter dalam *machine learning* peningkat gradien yang penting. Nilai *learning rate* antara 0 hingga 1. Laju pembelajaran menentukan tingkat dimana algoritma belajar dari setiap iterasi. Nilai *learning rate* yang lebih rendah berarti pembelajaran menjadi lebih lambat, karna mengurangi kontribusi setiap pohon dalam *ensemble*, sehingga membantu mencegah *overfitting*. Sebaliknya, nilai *learning rate* yang lebih tinggi mempercepat pembelajaran, tetapi dapat menyebabkan *overfitting* jika tidak diatur dengan baik.

2. N_estimator

N_estimator adalah teknik untuk menentukan jumlah pohon yang akan dibangun dalam *ensemble*. Tiap putaran peningkatan menambahkan pohon baru kedalam keseluruhan dan model perlahan belajar memperbaiki kesalahan yang dibuat oleh pohon sebelumnya. *N_estimator* mengarahkan kompleksitas model, mempengaruhi waktu pelatihan, dan kemampuan model untuk digeneralisasi ke data yang tidak terlihat. Meningkatkan nilai *n_estimators* akan meningkatkan

kompleksitas model, karena memungkinkan model untuk menangkap pola yang lebih rumit dalam data. Akan tetapi, jika menambahkan terlalu banyak pohon dapat menyebabkan *overfitting*.

3. *Max depth*

Max depth mempresetasikan seberapa dalam setiap pohon dalam proses peningkatan dapat tumbuh selama pelatihan. Meningkatkan nilai ini akan membuat model menjadi lebih kompleks dan lebih cenderung *overfit*. Pada XGBoost, *max_depth* default adalah 6, yang berarti bahwa setiap pohon dalam model diizinkan untuk tumbuh hingga kedalaman 6 level. Pada penelitian yang dilakukan oleh Firdaus *et al.* (2025), melakukan penelitian tentang optimalisasi *hyperparameter* XGBoost menggunakan *grid search* untuk prediksi. *Hyperparameter* yang dilakukan dapat dilihat pada Tabel 3.5.

Tabel 3.5 *Hyperparameter* XGBoost

Parameter	Rentang
<i>Learning_rate</i>	0.01, 0.02, 0.03
<i>N_estimators</i>	100, 200, 300, 400, 500
<i>Max_depth</i>	3, 4, 5, 6, 7, 8, 9, 10

3.6.2 *Grid Search*

Proses optimasi *grid search* bertujuan untuk mengidentifikasi secara sistematis kombinasi optimal *hyperparameter* yang memaksimalkan kinerja prediktif model. Dalam pendekatan ini, algoritma mengevaluasi secara menyeluruh semua kemungkinan kombinasi dalam *grid* nilai *hyperparameter* yang telah ditentukan sebelumnya menggunakan validasi silang k-fold untuk memastikan estimasi kinerja yang andal dan tidak bias. Meskipun membutuhkan komputasi intensif, pencarian menyeluruh ini menjamin bahwa optimum global dalam ruang pencarian ditemukan, asalkan resolusi *grid* cukup detail.

3.6.3 *Training Process*

Tahap berikutnya adalah melatih model menggunakan data yang sudah disiapkan. Proses melatih data dilakukan dengan menggunakan variasi *learning rate*, *n_estimators*, dan *max_depth*. Keseluruhan rentang untuk nilai pada setiap parameter pelatihan dapat dilihat pada Tabel 3.5.

1. Install Requirement

Tahap ini adalah proses menginstall *requirement* yang dibutuhkan untuk model. Dalam model ini *requirement* yang dipakai adalah *pandas* versi 2.2.2, *numpy* versi 2.0.2, *matplotlib* versi 3.10.0, *seaborn* versi 0.13.2, *scikit-learn* versi 1.6.1, *xgboost* versi 3.1.1, *imblearn* versi 0.14.0, dan *joblib* versi 1.5.2. Proses instalasi *requirement* dapat dilihat pada Gambar 3.16.

```
!pip install pandas==2.2.2
!pip install numpy==2.0.2
!pip install matplotlib==3.10.0
!pip install seaborn==0.13.2
!pip install scikit-learn==1.6.1
!pip install xgboost==3.1.1
!pip install imblearn==0.14.0
!pip install joblib==1.5.2
```

Gambar 3.16 *Install Requirement*

2. Library

Langkah berikutnya adalah mengimport library yang akan digunakan dalam penelitian, yaitu XGBoost. Proses *import library* dapat dilihat pada Gambar 3.17.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import joblib

import sklearn
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import confusion_matrix, classification_report
from sklearn.model_selection import GridSearchCV

import xgboost as xgb
import imblearn.over_sampling
from imblearn.over_sampling import SMOTE
```

Gambar 3.17 *Import Library*

3.7 Model Evaluation

Setelah melakukan *training* pada model, data validasi digunakan untuk memastikan akurasi atau ketepatan kinerja pada model dalam melakukan prediksi dan membandingkan hasil prediksi dengan nilai yang sebenarnya. Perbedaan antara prediksi dan nilai sebenarnya akan diukur menggunakan 4 metrik evaluasi seperti : *Recall* dan *Precision* untuk model XGBoost serta *Precision* dan *Intersection over Union* (IoU) untuk model Faster R-CNN.

3.8 Best Model Selection & Deployment

Model terbaik dipilih berdasarkan model evaluasi. Model kemudian ditetapkan untuk memproses data baru.

3.9 Output

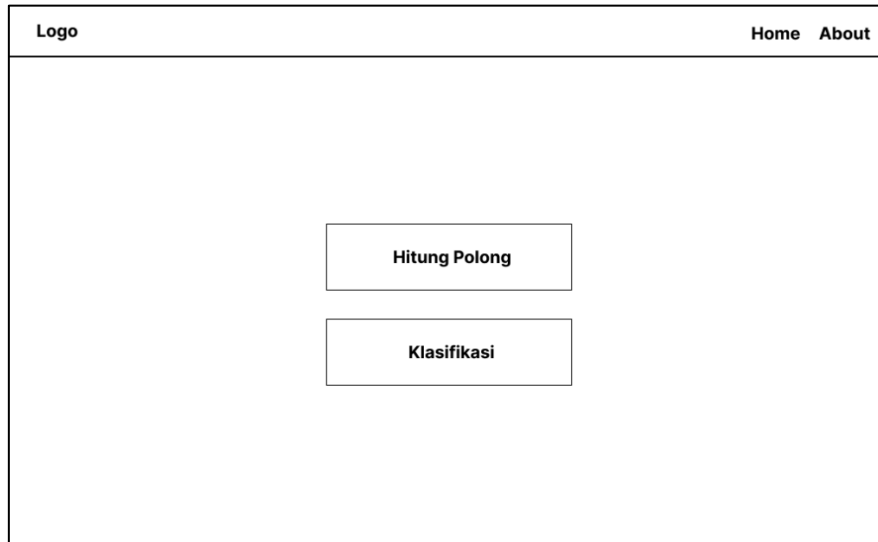
Hasil dari semua proses yang telah dilakukan akan menghasilkan informasi jumlah polong berbiji 1, 2, 3, jumlah polong, jumlah biji, dan klasifikasi tanaman tersebut merupakan tanaman yang terpilih atau tidak terpilih.

3.10 Perancangan Antarmuka Sistem

Langkah berikutnya adalah merancang antarmuka sistem berbasis web yang berfungsi sebagai representasi visual dari sistem deteksi, kuantifikasi polong kedelai dan klasifikasi tanaman kedelai. Tujuannya adalah untuk memudahkan pengguna dalam melihat jumlah polong, biji, dan prediksi tanaman tersebut termasuk dalam kategori terpilih atau tidak terpilih. Pada web ini, terdapat dua menu yaitu : hitung polong dan klasifikasi. Pada menu hitung polong, terdapat *input* gambar dan hasil dari sistem ini adalah jumlah polong biji 1, 2, 3, jumlah keseluruhan polong. Pada menu klasifikasi, terdapat *input* morfologi tanaman seperti tinggi tanaman, jumlah cabang primer, jumlah polong, jumlah polong berisi, jumlah biji, bobot biji, dan bobot 100 biji per-tanaman yang hasilnya adalah kategori tanaman terpilih atau tidak terpilih..

3.10.1 Rancangan Tampilan Halaman Beranda

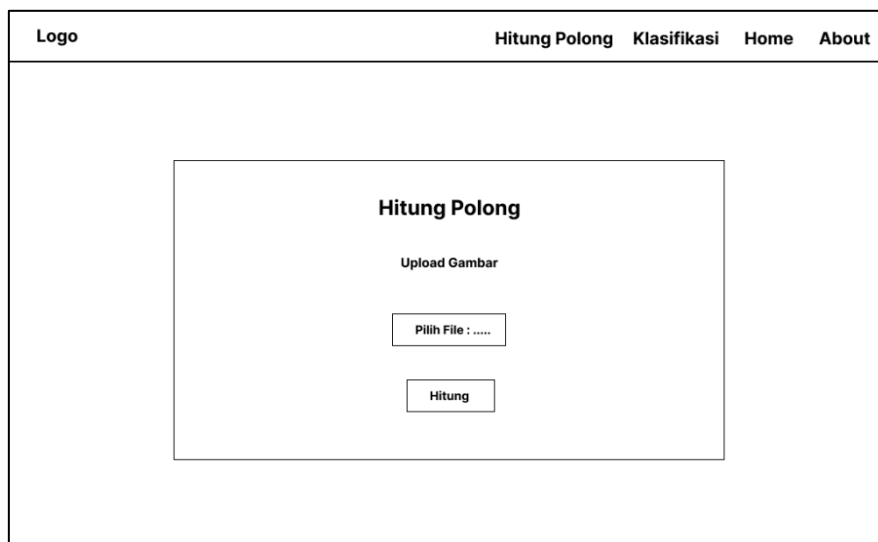
Halaman beranda merupakan halaman pertama yang dilihat oleh pengguna saat membuka sistem. Halaman ini terdiri dari judul aplikasi, menu hitung polong, dan klasifikasi. Di setiap halaman terdapat *navbar* yang terdapat tombol *home* yang mengarah ke halaman beranda awal ini. Rancangan tampilan halaman beranda dapat dilihat pada Gambar 3.18.



Gambar 3.18 Rancangan Halaman Beranda

3.10.2 Rancangan Tampilan Halaman Hitung Polong

Pada halaman Hitung Polong merupakan halaman dimana pengguna mengunggah data citra polong kedelai yang nantinya akan diidentifikasi oleh sistem. Halaman ini memiliki bagian untuk memasukkan citra polong kedelai dan tombol deteksi untuk melakukan proses perhitungan polong yang ada pada gambar. Rancangan tampilan halaman hitung polong dapat dilihat pada gambar 3.19.

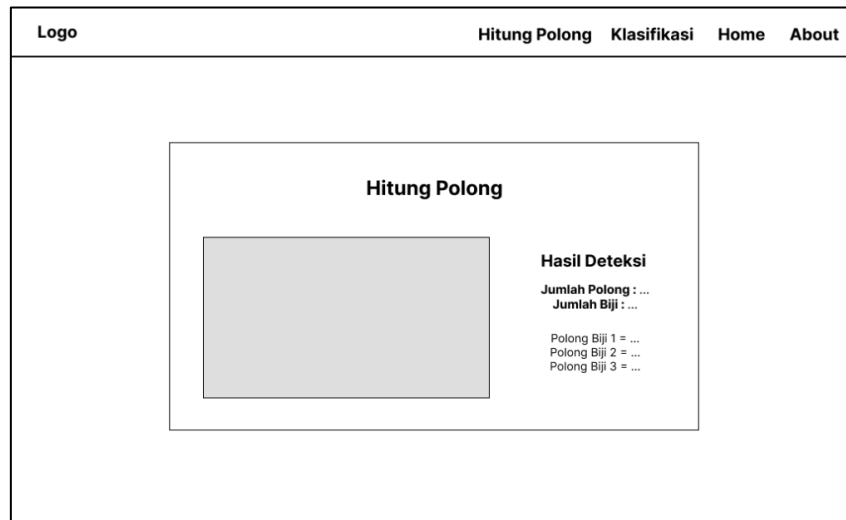


Gambar 3.19 Rancangan Halaman Hitung Polong

3.10.3 Rancangan Halaman Hasil Hitung Polong

Pada halaman Hitung Polong merupakan halaman yang menyajikan *output* dari sistem setelah pengguna mengunggah citra polong kedelai dan menjalankan

proses perhitungan. Pada halaman ini, ditampilkan satu gambar citra yang sudah dibounding box per-kelas. Pada sisi kanan menampilkan hasil deteksi seperti jumlah polong, jumlah biji, banyaknya polong biji 1, 2, dan 3. Rancangan halaman hasil hitung polong dapat dilihat pada Gambar 3.20.

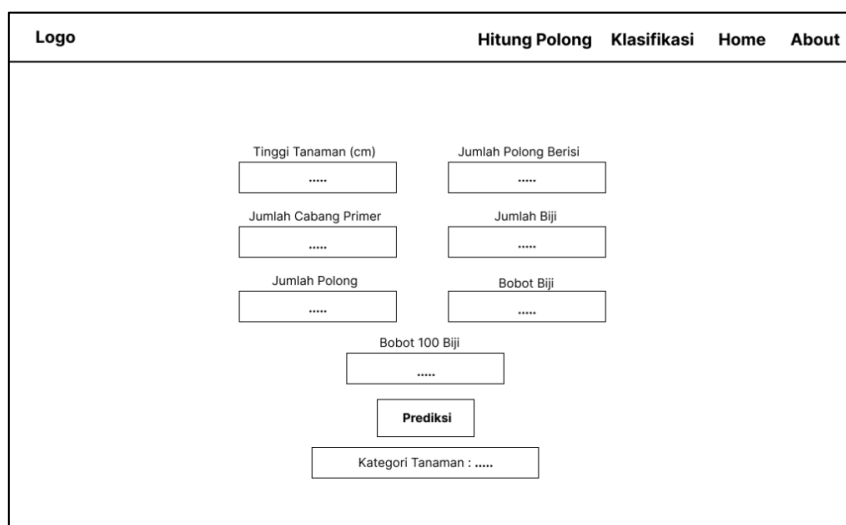


The image shows a web page layout for 'Hitung Polong'. At the top is a navigation bar with 'Logo' on the left and 'Hitung Polong', 'Klasifikasi', 'Home', and 'About' on the right. The main content area contains a central box titled 'Hitung Polong'. Inside this box, on the left, is a large gray rectangle representing an image placeholder. On the right, under the heading 'Hasil Deteksi', there are labels for 'Jumlah Polong : ...', 'Jumlah Biji : ...', 'Polong Biji 1 = ...', 'Polong Biji 2 = ...', and 'Polong Biji 3 = ...'.

Gambar 3.20 Rancangan Halaman Hasil Hitung Polong

3.10.4 Rancangan Tampilan Halaman Klasifikasi

Pada halaman Klasifikasi merupakan halaman dimana pengguna memasukkan nilai ciri morfologi tanaman kedelai lalu menekan tombol Prediksi. Sistem akan menganalisis data dan menentukan apakah tanaman tersebut Terpilih atau Tidak Terpilih. Rancangan tampilan halaman hitung polong dapat dilihat pada Gambar 3.21.

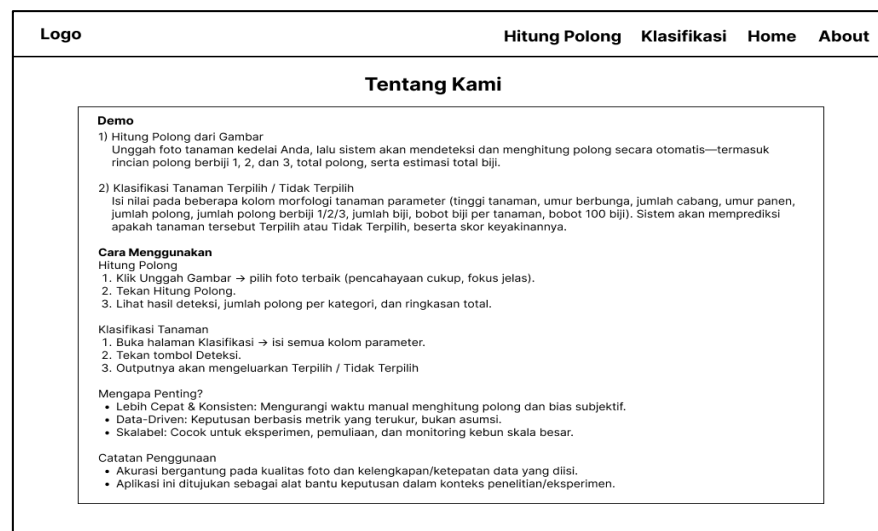


The image shows a web page layout for 'Klasifikasi'. At the top is a navigation bar with 'Logo' on the left and 'Hitung Polong', 'Klasifikasi', 'Home', and 'About' on the right. The main content area contains several input fields for morphological data: 'Tinggi Tanaman (cm)', 'Jumlah Polong Berisi', 'Jumlah Cabang Primer', 'Jumlah Biji', 'Jumlah Polong', 'Bobot Biji', and 'Bobot 100 Biji'. Each field has a text label above it and a text input box with five asterisks. Below these fields is a 'Prediksi' button. At the bottom, there is a label 'Kategori Tanaman :'.

Gambar 3.21 Rancangan Halaman Klasifikasi

3.10.5 Rancangan Tampilan Halaman *About*

Halaman *about* ini bertujuan untuk memberikan informasi kepada pengguna mengenai fungsi, tujuan, dan cara penggunaan aplikasi. Pada halaman ini, pengguna akan diperkenalkan dengan latar belakang aplikasi serta kenapa adanya aplikasi ini. Informasi yang ditampilkan disajikan secara ringkas namun informatif agar mudah untuk dipahami dan dilakukan oleh pengguna. Rancangan tampilan halaman hitung polong dapat dilihat pada Gambar 3.22.



Gambar 3.22 Rancangan Halaman *About*

BAB IV

IMPLEMENTASI DAN PENGUJIAN

Bab ini akan membahas bagaimana metode Faster R-CNN dan XGBoost diimplementasikan dalam melakukan deteksi, kuantifikasi polong kedelai dan klasifikasi tanaman kedelai yang sebelumnya dirancang pada bab 3.

4.1 Implementasi Sistem

Pada tahap ini, dilakukan implementasi sistem untuk melakukan deteksi, kuantifikasi polong kedelai menggunakan Faster R-CNN dan klasifikasi menggunakan XGBoost. Implementasi dilakukan menggunakan bahasa pemrograman Python dan menghasilkan sebuah sistem berbasis web dengan menggunakan *framework* Flask.

4.1.1 Perangkat Keras

Adapun dalam pembangunan sistem ini, penulis menggunakan perangkat keras yang memiliki spesifikasi seperti berikut :

- | | |
|---------------------|--|
| 1. Device | : ASUS |
| 2. Operating System | : Windows 11 64-bit |
| 3. Processor | : Intel(R) Core(TM) i5-10300H @2.50GHz |
| 4. RAM | : 60 GB dan 15 GB |
| 5. Capacity | : 512 GB |
| 6. GPU | : A100 dan T4 |

4.1.2 Perangkat Lunak

Dalam pembangunan sistem ini juga memerlukan perangkat lunak seperti berikut :

1. Python
2. Flask
3. Roboflow
4. Google Colab

4.2 Implementasi Data

Data yang digunakan dalam penelitian ini diambil langsung oleh penulis di Lab Pertanian Universitas Sumatera Utara dan diperoleh dari penelitian Jiang *et al.* (2025). Data citra yang digunakan merupakan citra polong tanaman kedelai. Seluruh gambar yang digunakan berformat .png dengan total 715 gambar setelah dilakukan anotasi.

4.3 Training Pada Metode Faster R-CNN

Untuk melakukan *training* pada model, penulis terlebih dahulu melakukan konfigurasi *hyperparameter* sesuai dengan skenario *training* yang dapat dilihat pada Tabel 4.1 guna memperoleh performa terbaik dari model yang dilatih. Model yang memiliki performa terbaik akan diimplementasikan ke dalam sistem. Model yang akan dihasilkan akan memiliki ekstensi .pth.

4.3.1 Evaluasi pada citra polong kedelai

Dalam proses *training* model prediksi polong kedelai, dilakukan beberapa percobaan menggunakan kombinasi *batch size*, *learning rate*, dan *weight decay* serta melakukan perbandingan dua strategi yaitu *fine-tuning*, yaitu untuk melatih seluruh jaringan (*full layers*), melatih lapisan kepala (*head layers*), dan melatih seluruh jaringan (*full layers* dengan *dropout*) untuk mendapatkan hasil yang terbaik. Eksperimen pertama di fokuskan pada pencarian *hyperparameter* terbaik untuk melatih *full layers* untuk memungkinkan model dapat beradaptasi secara mendalam dengan dataset citra polong kedelai. Hasil validasi model prediksi polong kedelai menggunakan *full layers* dengan berbagai kombinasi *hyperparameter* dapat dilihat pada Tabel 4.1.

Tabel 4.1 Hasil Evaluasi Model Polong Kedelai menggunakan *Full Layers*

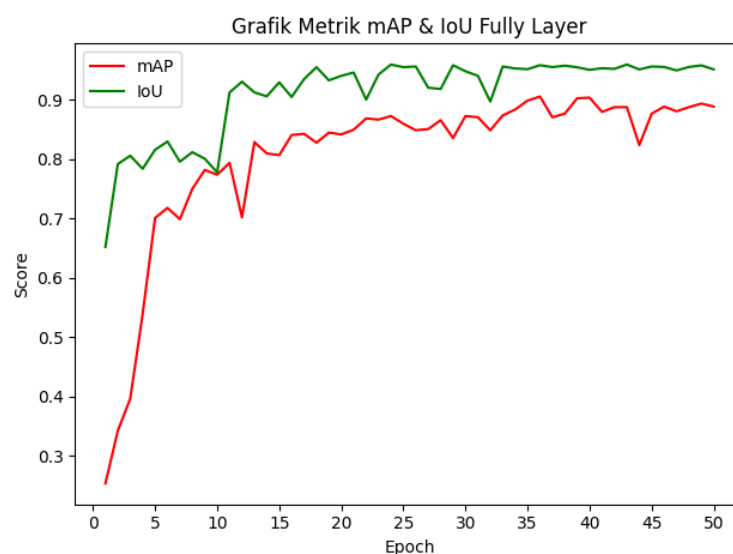
Batch Size	Learning Rate	Weight Decay	Epoch	mAP	IoU
1	0.001	0.0001	26	0.898	0.9617
1	0.005	0.0001	21	0.905	0.9574
2	0.001	0.0001	16	0.869	0.9574
2	0.005	0.0001	40	0.896	0.9596
4	0.001	0.0001	46	0.787	0.9592

4	0.005	0.0001	36	0.906	0.9588
8	0.001	0.0001	47	0.705	0.960
8	0.005	0.0001	33	0.888	0.9625
16	0.001	0.0001	48	0.824	0.959
16	0.005	0.0001	49	0.894	0.958
32	OUT OF MEMORY				

Berdasarkan Tabel 4.1, disajikan hasil dari beberapa eksperimen untuk mengevaluasi dan menemukan kombinasi *hyperparameter* terbaik untuk model deteksi polong kedelai saat melakukan pelatihan pada *full layers*. Eksperimen ini menggunakan variasi *hyperparameter* berupa *batch size* sebesar 1, 2, 4, 8, dan 16, *learning rate* 0,001 dan 0,005, dan *weight decay* sebesar 0,0001.

Dari hasil yang ditampilkan, dapat dilihat bahwa penggunaan *batch size* sebesar 32 menyebabkan kegagalan proses pelatihan karena keterbatasan memori (*Out of Memory*). Hal ini mengindikasikan bahwa dengan sumber daya perangkat keras yang digunakan, *batch size* 32 terlalu besar untuk ditampung. Sehingga, proses latihan hanya bisa menggunakan *batch size* 1, 2, 4, 8, dan 16.

Dengan konfigurasi *hyperparameter* tersebut. Model berhasil mencapai hasil terbaik saat menggunakan *batch size* 4, *learning rate* 0,005, *weight decay* 0,0001, yang menghasilkan mAP sebesar 0,906, dan IoU 0,9588 pada epoch ke-36.



Gambar 4.1 Grafik Matrik *Fully Layer*

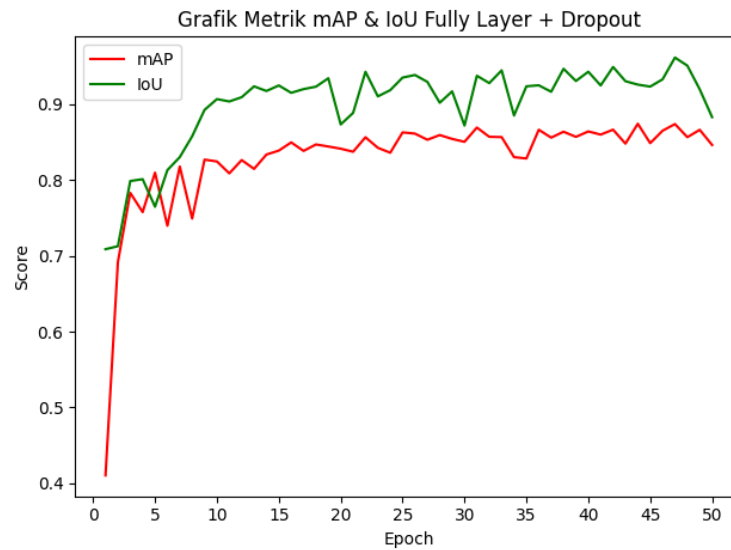
Selanjutnya, untuk menentukan strategi *fine-tuning* yang paling efektif, konfigurasi terbaik ini kemudian diuji kembali dengan strategi kedua, yaitu hanya melatih *head layer*. Perbandingan ini bertujuan untuk mengevaluasi apakah efisiensi pelatihan dapat dicapai tanpa mengorbankan akurasi secara signifikan.

Setelah melakukan strategi *fine tuning* khusus pada *head layer*, kemudian dilakukan uji kembali dengan strategi ketiga, yaitu dengan menggunakan *full layer* dengan *dropout*. Perbandingan langsung antara ketiga *layer* menggunakan *hyperparameter* terbaik dapat dilihat pada Tabel 4.2.

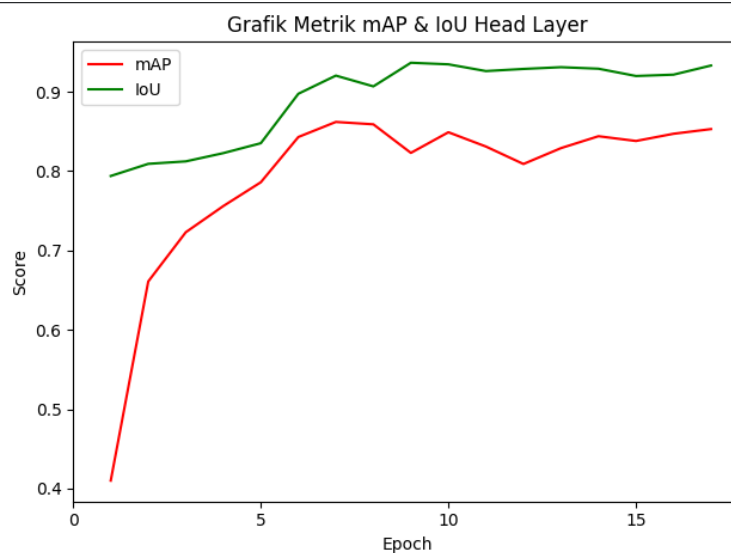
Tabel 4.2 Perbandingan Hasil Evaluasi *Fine-Tuning* dengan Parameter Terbaik

Layer	Learning Rate	Weight Decay	Epoch	mAP	IoU
<i>Full Layers</i>	0.005	0.0001	36	0.906	0.9588
<i>Head Layers</i>	0.005	0.0001	7	0.862	0.9592
<i>Full Layers + Dropout</i>	0.005	0.0001	47	0.874	0.9615

Dari perbandingan pada Tabel 4.2, dapat disimpulkan bahwa strategi *training* menggunakan *full layers* menghasilkan performa yang paling baik. Meskipun *training* menggunakan *head layers* yang memberikan hasil terbaik pada *epoch* ke-7 yang lebih awal dari *full layers* yang memberikan hasil terbaik pada *epoch* ke-36 dan *full layers* dengan *dropout* yang memberikan hasil terbaik pada *epoch* ke-47 dengan metrik IoU yang lebih tinggi sekitar 0,0027 dari *full layers* tetapi mAP lebih rendah sekitar 0,031, tetapi *full layers* yang mampu mencapai nilai yang lebih baik. Hal ini mengindikasikan bahwa menggunakan *full layers* terbukti lebih efektif untuk mencapai hasil yang terbaik. Oleh karena itu, model yang dilatih *full layers* dengan konfigurasi *hyperparameter batch size* 4, *learning rate* 0,005, dan *weight decay* 0,0001 dipilih sebagai model *final* untuk digunakan pada tahap pengujian selanjutnya.

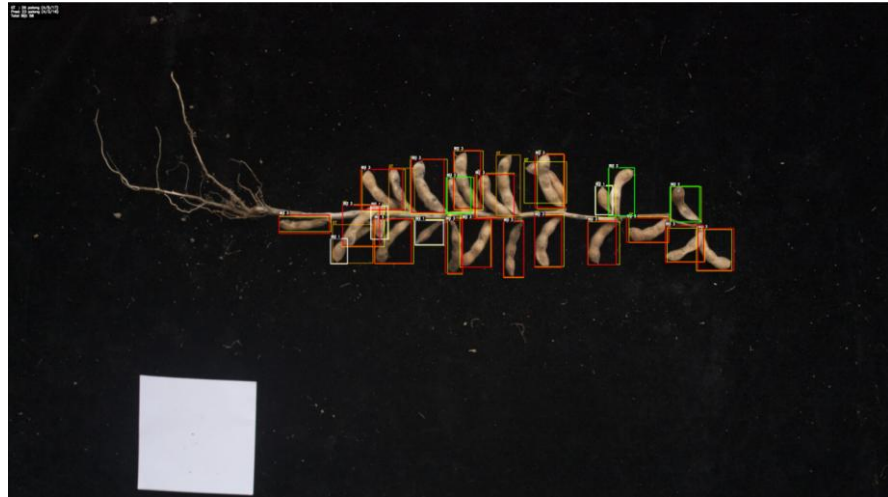


Gambar 4.2 Grafik Metrik *Fully Layer* dengan *Dropout*



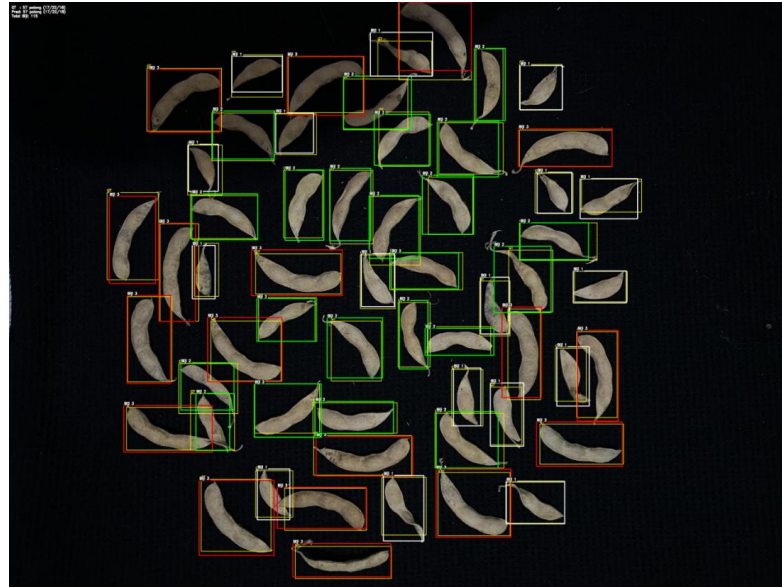
Gambar 4.3 Grafik Metrik *Head Layer*

Model dengan konfigurasi *batch size* 4, *learning rate* 0,005, dan *weight decay* 0,001 dengan menggunakan *full layers* menghasilkan performa terbaik dengan nilai mAP sebesar 0,906 dan IoU sebesar 0.9588 pada *epoch* ke-36 dengan waktu pelatihan sekitar 285 menit. Nilai mAP sebesar 0,906 menandakan bahwa model memiliki kemampuan yang baik untuk mendeteksi seluruh objek target. Nilai IoU sebesar 0.9588 menandakan bahwa model dapat memprediksi *bounding box* dengan benar atau tidak.



Gambar 4.4 Deteksi Polong pada Tanaman

Pada Gambar 4.4, ditunjukkan hasil deteksi polong pada tanaman utuh. Sampel ini memiliki nilai aktual 26 polong yang terdiri dari 4 polong biji-1, 5 polong biji-2, dan 17 polong biji-3. Model memprediksi dengan jumlah 23 polong yang terdiri dari 4 polong biji-1, 3 polong biji-2, dan 16 polong biji-3, sehingga terjadi selisih -3 polong (*false negative*) dimana 2 polong biji-2 dan 1 polong biji-3 tidak terdeteksi. Kesalahan ini terutama diakibatkan oleh tumpang tindih antar polong, sehingga proposal yang saling bertumpuk cenderung digabung atau disupresi oleh NMS, sehingga satu polong menghapus polong yang tumpang tindih dengannya. Tumpang tindih mengakibatkan bentuk polong dan kemiripan tekstur membuat pemisahan antar objek menjadi lebih sulit. Ini mengakibatkan bahwa objek yang tumpang tindih dengan objek lainnya, akurasi deteksi menjadi menurun. Oleh karena itu, untuk memaksimalkan kinerja model dalam melakukan deteksi, posisikan polong agar tidak saling tumpang tindih dengan polong lainnya agar model dapat bekerja dengan lebih maksimal.



Gambar 4.5 Deteksi Polong setelah dipetik

Pada Gambar 4.5, ditampilkan hasil deteksi polong setelah dipetik dan ditata diatas alas berwarna hitam tanpa saling tumpang tindih. Sampel ini memiliki nilai aktual 57 polong yang terdiri dari 17 polong biji-1, 22 polong biji-2, dan 18 polong biji-3. Sistem deteksi menghasilkan jumlah dan komposisi kelas identik dengan nilai aktual. Kesesuaian ini menunjukkan bahwa, ketika objek dipisahkan dan tidak saling menutup atau tumpang tindih dengan objek lainnya, model mampu melakukan lokalisasi dan perhitungan dengan akurat. Dengan demikian, pemisahan polong dari tanaman, penataan tanpa tumpang tindih, serta latar berkontras tinggi, dan pencahayaan yang baik, merupakan cara yang direkomendasikan untuk memaksimalkan kinerja dari model.

4.4 Training Pada Metode XGBoost

Untuk melakukan *training* model klasifikasi, penulis menggunakan XGBoost pada data tabular hasil *pre-processing* (Tabel 3.3) dengan skema *train* 80% dan *test* 20%. Ketidakseimbangan kelas diatasi dengan *SMOTE* pada data latih, sedangkan normalisasi skala fitur dilakukan melalui *scaler* yang di-fit hanya pada data latih dan kemudian diaplikasikan pada *test* untuk memberikan kontribusi yang seimbang terhadap hasil model, model akhir akan memiliki ekstensi .pkl.

4.4.1 Evaluasi pada parameter morfologi kedelai

Kombinasi terbaik yang diperoleh adalah $learning_rate = 0,01$, $n_estimators = 200$, $max_depth = 3$. Kombinasi ini dipilih karena menghasilkan akurasi pengujian tertinggi sekaligus mampu menjaga keseimbangan *accuracy*, *precision*, dan *recall* pada dua kelas.

Tabel 4.3 Hasil Model dengan *Hyperparameter* Terbaik

Kelas	Precision	Recall	F1-Score
Tidak Terpilih (0)	1.00	0.98	0.99
Terpilih (1)	0.98	1.00	0.99
Accuracy			0.99

Berdasarkan Tabel 4.3, model menunjukkan kinerja sangat baik dan seimbang pada kedua kelas. Untuk kelas 0, model meraih *precision* 1.00 dan *recall* 0.98 (*F1-Score* = 0.99). Artinya, hampir semua prediksi kelas 0 tepat sasaran, dan hanya sebagian kecil sampel kelas 0 yang keliru diprediksi sebagai kelas lain (sedikit *false negative*). Untuk kelas 1, model mendapatkan *precision* 0.98 dan *recall* 1.00 (*F1-Score* = 0.99). Ini berarti tidak ada sampel kelas 1 yang terlewat (*false negative* = 0), namun masih ada sedikit kasus ketika sampel kelas 0 diprediksi sebagai kelas 1 (*false positive* kecil). Jika dalam konteks kelas 1 = “Terpilih”, maka *recall* 1.00 sangat menguntungkan karena tidak ada tanaman terpilih yang terlewat.

Tinggi Tanaman (cm)	Jumlah Cabang Primer	Jumlah Polong Per Tanaman	Jumlah Polong Berisi Per Tanaman	Jumlah Biji Per Tanaman	Bobot Biji Per Tanaman	Bobot 100 Biji Per Tanaman	Aktual	Prediksi
43.8	3	65	62	131	25.44	20.54	Tidak Terpilih	Terpilih

Gambar 4.6 Klasifikasi Pada Parameter Morfologi Kedelai

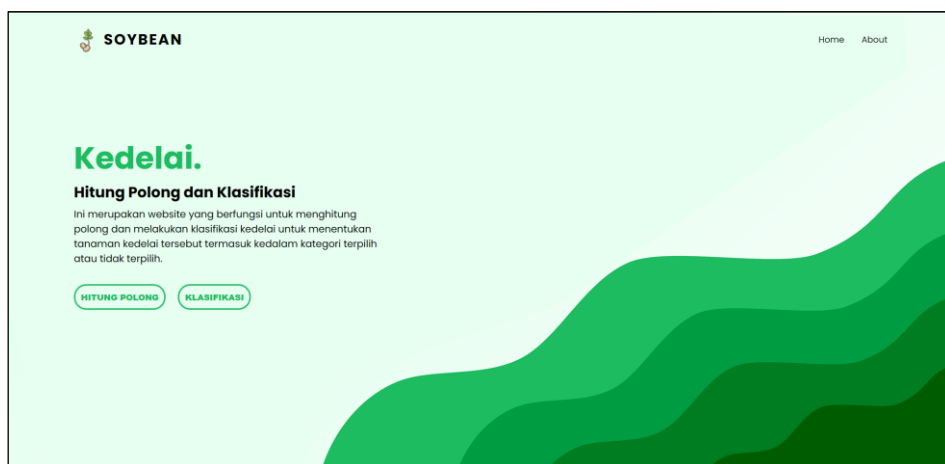
Pada Gambar 4.6, ditampilkan hasil klasifikasi pada parameter morfologi tanaman kedelai, di mana sampel dengan nilai aktual “Tidak Terpilih” justru diprediksi sebagai “Terpilih” oleh model. Kesalahan tersebut kemungkinan terjadi karena model XGBoost mempelajari pola yang dominan dari mayoritas data, sehingga pada sampel yang memiliki karakteristik tidak umum misalnya dengan nilai produktivitas tinggi namun tetap dikategorikan “Terpilih” oleh model.

4.5 Implementasi Desain Antarmuka

Tampilan antarmuka dari sistem ini memiliki satu halaman yang terdiri dari 4 *section*, yaitu halaman depan, pengenalan *website*, hitung polong, dan klasifikasi.

4.5.1 Tampilan Halaman Depan

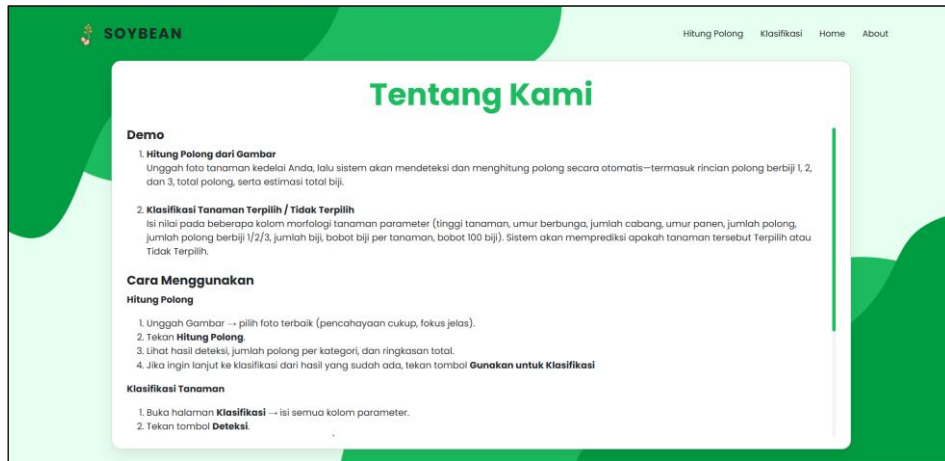
Tampilan ini akan muncul ketika pengguna masuk kedalam *website*. Pada bagian ini, terdapat *navbar* yang berisi tautan ke *section Home*, *About*, serta sebuah tombol “Hitung Polong dan Klasifikasi”. Tampilan halaman depan dapat dilihat pada Gambar 4.7.



Gambar 4.7 Tampilan Halaman Depan

4.5.2 Tampilan Pengenalan Website

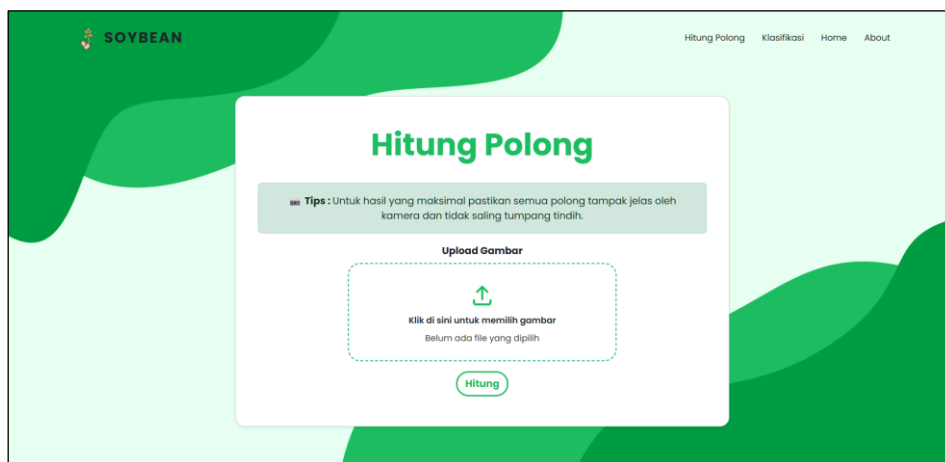
Halaman ini akan muncul ketika pengguna menekan *navbar About*. *Section* ini merupakan tampilan yang memberi pengguna penjelasan mengenai cara penggunaan dan fungsi dari *website* ini. Tampilan pengenalan *website* dapat dilihat pada Gambar 4.8.



Gambar 4.8 Tampilan Halaman Pengenalan Website

4.5.3 Tampilan Halaman Hitung Polong

Halaman ini akan muncul ketika pengguna menekan tombol “Hitung Polong” pada halaman depan atau pada *navbar*. *Section* ini berguna untuk pengguna mengunggah citra yang ingin dihitung polongnya. Tampilan pada *section* ini dapat dilihat pada Gambar 4.9.

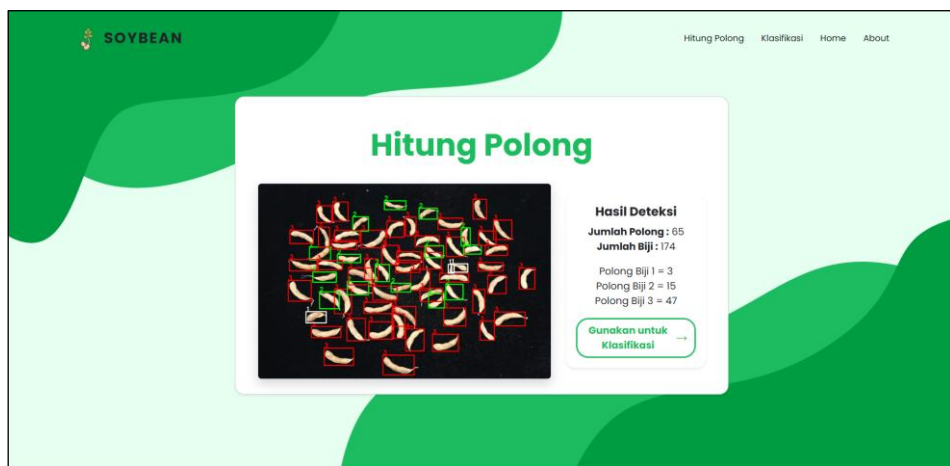


Gambar 4.9 Tampilan Halaman Hitung Polong

Sebagaimana ditunjukkan pada Gambar 4.3, terdapat area khusus ditengah halaman yang memungkinkan pengguna untuk memilih file dari perangkat. Sebuah teks status umpan balik yang memberitahu file sudah dipilih atau belum. Tombol “Hitung” yang berada pada bagian bawah akan menjadi berfungsi jika gambar sudah dipilih dan dapat melakukan perhitungan polong.

4.5.4 Tampilan Hasil Hitung Polong

Halaman ini akan muncul ketika setelah pengguna mengunggah citra yang ingin dihitung polongnya dan menekan tombol hitung. *Section* ini berguna untuk melihat jumlah polong yang dideteksi. Pada halaman ini terdapat tombol “Gunakan untuk Klasifikasi” yang berfungsi untuk mengisi kolom pada halaman klasifikasi untuk dari hasil gambar yang diunggah untuk mempercepat waktu seleksi tanaman terpilih dan tidak terpilih. Tampilan pada *section* ini dapat dilihat pada Gambar 4.10.



Gambar 4.10 Tampilan Hasil Hitung Polong

Sebagaimana yang ditunjukkan pada Gambar 4.4, pada sisi sebelah kiri akan memperlihatkan gambar unggahan yang sudah diberikan *bounding box* per-kelas yaitu polong biji 1, polong biji 2, dan polong biji 3 dengan warna yang berbeda. Pada sisi sebelah kanan akan memperlihatkan hasil deteksi yang terdiri dari jumlah polong, jumlah biji, dan keterangan polong setiap biji. Terdapat juga tombol “Gunakan untuk klasifikasi” yang berfungsi untuk langsung beralih ke-halaman klasifikasi dengan data polong yang dihitung.

4.5.5 Tampilan Halaman Klasifikasi

Halaman ini akan muncul ketika pengguna menekan tombol “Klasifikasi” pada halaman depan atau pada *navbar*. *Section* ini berguna untuk pengguna memasukkan data morfologi untuk menentukan tanaman tersebut termasuk tanaman terpilih atau tidak terpilih. Tampilan pada *section* ini dapat dilihat pada Gambar 4.11.

Gambar 4.11 Tampilan Halaman Klasifikasi

Setelah pengguna mengisi semua kolom morfologi atau parameter kedelai dan menekan tombol “Prediksi”, pada bagian bawah akan muncul kategori tanaman tersebut terpilih atau tidak terpilih.

4.6 Prosedur Operasional

Prosedur operasional menjelaskan bagaimana menjalankan sistem yang sudah dibangun. Tampilan utama dari sistem ini berisi informasi mengenai sistem dan input berupa gambar yang nantinya akan menentukan jumlah polong, jumlah biji, jumlah biji berisi, dan memprediksi tanaman tersebut termasuk kategori terpilih atau tidak terpilih. Untuk menjalankan aplikasi ini, pengguna terlebih dahulu mengunduh dan melakukan instalasi *environment* yang dibutuhkan dalam aplikasi ini dan *framework* Flask. Untuk menjalankan website-nya, pengguna harus mengetik “python app.py” pada terminal atau *prompt*.

Setelah menjalankan Flask, maka sebuah alamat akan diberikan. Alamat ini digunakan pada *web browser* untuk mengakses aplikasi ini. Pada gambar diatas, pengguna diberikan alamat <http://127.0.0.1:5000/> untuk mengakses aplikasi. Pada aplikasi terdapat tombol “Hitung Polong” yang digunakan untuk menuju *section upload* gambar. Selanjutnya pengguna memasukkan gambar yang ingin dihitung polongnya dengan melakukan klik pada tombol “Hitung” dan terdapat juga tombol “Klasifikasi” yang digunakan untuk menuju *section* mengisi data morfologi tanaman kedelai yang ingin diketahui kategori tanaman tersebut terpilih atau tidak terpilih dengan melakukan klik pada tombol “Prediksi”. Setelah itu sistem akan mulai berjalan untuk menghitung polong dan prediksi kategori tanaman terpilih atau tidak terpilih.




4.7 Pengujian Sistem

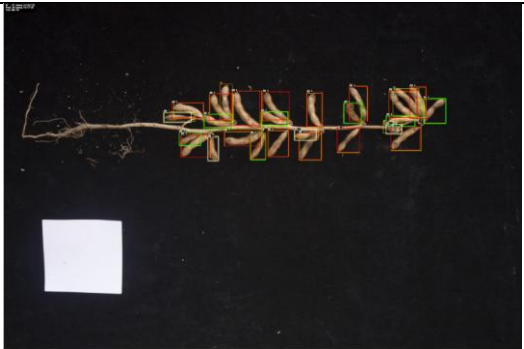
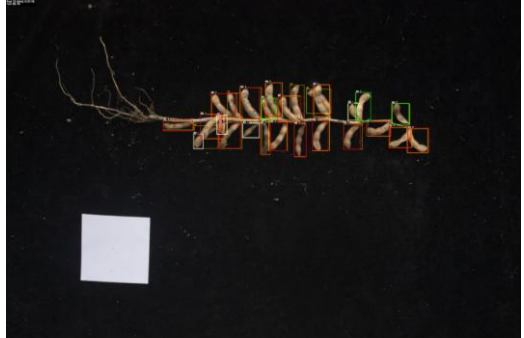

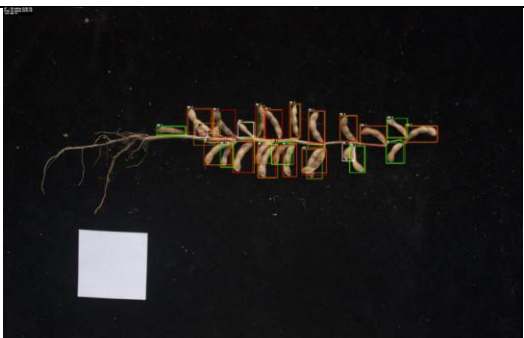
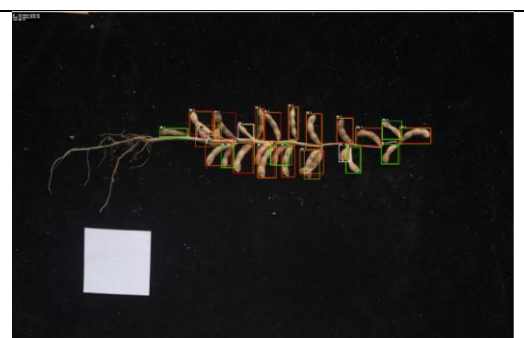
Pada tahap ini, sistem yang telah dibangun akan diuji dengan tujuan menganalisa kemampuan dan kinerja sistem dalam mendeteksi polong kedelai biji 1, biji 2, dan biji 3 dengan menggunakan Faster R-CNN dan melakukan klasifikasi tanaman terpilih dan tidak terpilih menggunakan XGBoost. Pengujian dilakukan terhadap 20 data *testing*.

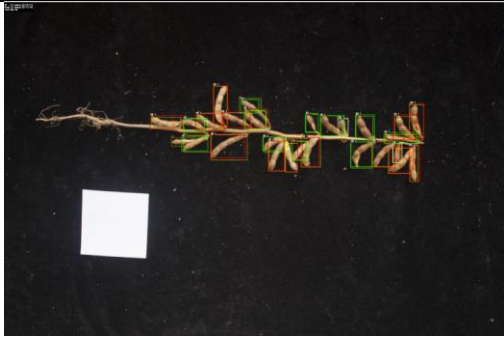



4.7.1 Pengujian Model Deteksi Objek Polong Kedelai



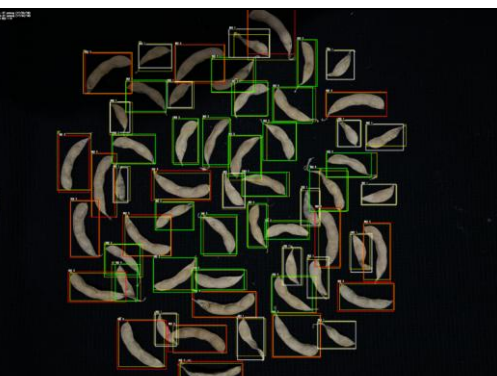
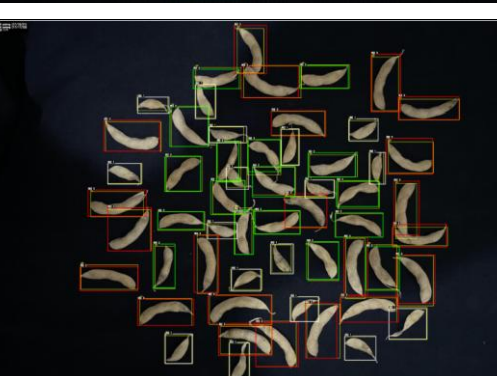
Hasil pengujian model deteksi polong dapat dilihat pada Tabel 4.4.





Tabel 4.4 Hasil Pengujian Model Deteksi Polong Kedelai

No	Deteksi Polong	Prediksi Polong	IoU	mAP
1		Polong Biji 1 : 1 Polong Biji 2 : 13 Polong Biji 3 : 19 Total Polong : 33	0.971	0.921
2		Polong Biji 1 : 5 Polong Biji 2 : 6 Polong Biji 3 : 15 Total Polong : 26	0.970	0.963
3		Polong Biji 1 : 0 Polong Biji 2 : 12 Polong Biji 3 : 19 Total Polong : 31	0.952	0.939

4		<p>Polong Biji 1 : 4 Polong Biji 2 : 7 Polong Biji 3 : 15 Total Polong : 26</p>	0.966	0.926
5		<p>Polong Biji 1 : 4 Polong Biji 2 : 3 Polong Biji 3 : 16 Total Polong : 23</p>	0.962	0.846
6		<p>Polong Biji 1 : 1 Polong Biji 2 : 13 Polong Biji 3 : 18 Total Polong : 32</p>	0.901	0.934
7		<p>Polong Biji 1 : 3 Polong Biji 2 : 6 Polong Biji 3 : 14 Total Polong : 23</p>	0.959	0.841
8		<p>Polong Biji 1 : 3 Polong Biji 2 : 6 Polong Biji 3 : 14 Total Polong : 23</p>	0.959	0.841

9		<p>Polong Biji 1 : 2</p> <p>Polong Biji 2 : 11</p> <p>Polong Biji 3 : 14</p> <p>Total Polong : 27</p>	0.869	0.959
10		<p>Polong Biji 1 : 4</p> <p>Polong Biji 2 : 9</p> <p>Polong Biji 3 : 19</p> <p>Total Polong : 32</p>	0.898	0.886
11		<p>Polong Biji 1 : 18</p> <p>Polong Biji 2 : 16</p> <p>Polong Biji 3 : 22</p> <p>Total Polong : 56</p>	0.958	0.849
12		<p>Polong Biji 1 : 18</p> <p>Polong Biji 2 : 16</p> <p>Polong Biji 3 : 22</p> <p>Total Polong : 56</p>	0.958	0.897

13		Polong Biji 1 : 24 Polong Biji 2 : 16 Polong Biji 3 : 10 Total Polong : 50	0.966	0.901
14		Polong Biji 1 : 20 Polong Biji 2 : 21 Polong Biji 3 : 21 Total Polong : 62	0.962	0.984
15		Polong Biji 1 : 17 Polong Biji 2 : 22 Polong Biji 3 : 18 Total Polong : 57	0.954	0.908
16		Polong Biji 1 : 17 Polong Biji 2 : 17 Polong Biji 3 : 22 Total Polong : 56	0.941	0.881

17		Polong Biji 1 : 17 Polong Biji 2 : 25 Polong Biji 3 : 15 Total Polong : 57	0.956	0.890
18		Polong Biji 1 : 17 Polong Biji 2 : 17 Polong Biji 3 : 22 Total Polong : 56	0.960	0.911
19		Polong Biji 1 : 20 Polong Biji 2 : 22 Polong Biji 3 : 20 Total Polong : 62	0.984	0.891
20		Polong Biji 1 : 25 Polong Biji 2 : 14 Polong Biji 3 : 11 Total Polong : 50	0.963	0.931

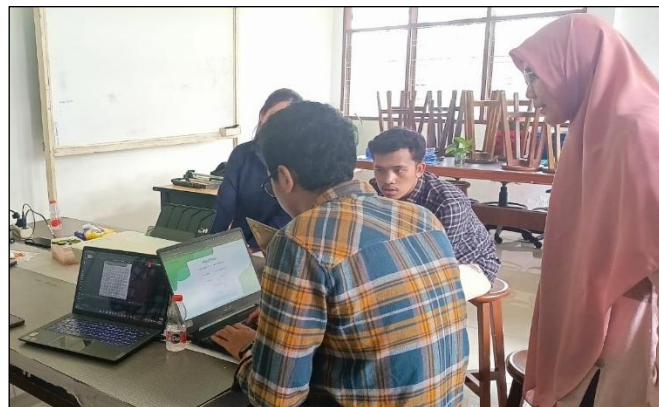
Tabel 4.4 menunjukkan hasil pengujian model deteksi pada citra polong kedelai menggunakan Faster R-CNN dengan ResNet50 sebagai *backbone*. Sistem diuji menggunakan data deteksi uji sebanyak 20 data. Berdasarkan hasil pengujian

menunjukkan nilai deteksi terbaik polong pada tanaman ada pada data nomor 2 dimana model mampu melakukan deteksi dengan nilai mAP sebesar 0.970 dan IoU sebesar 0.963 serta polong yang sudah dipisah memiliki hasil terbaik pada data nomor 14 dimana model melakukan deteksi dengan nilai mAP sebesar 0.962 dan IoU sebesar 0.984.

4.7.2 Validasi dengan Pakar

Untuk menilai performa aplikasi, maka dilakukan validasi oleh Prof. Dr. Diana Sofia Hanafiah SP., MP. Berdasarkan hasil validasi tersebut, diperoleh sejumlah masukan dan rekomendasi pengembangan, antara lain:

1. Pakar menilai sistem ini sangat potensial untuk mempercepat dan mempermudah perhitungan polong, biji, serta melakukan seleksi tanaman terpilih dan tidak terpilih, sehingga menghemat waktu dan tenaga peneliti di lapangan maupun laboratorium serta mendukung percepatan pengembangan varietas unggul kedelai.
2. Karena banyak tanaman yang harus diklasifikasi, disarankan sistem tersebut dapat melakukan klasifikasi massal agar banyak tanaman yang bisa diproses sekaligus.



Gambar 4.12 Validasi dengan Pakar

BAB V

KESIMPULAN DAN SARAN

Bab ini berisi rangkuman hasil dan penelitian objek deteksi dan klasifikasi kedelai menggunakan metode Faster R-CNN pada citra polong, XGBoost pada data kuantitatif kedelai dan memberikan saran sebagai acuan penelitian selanjutnya.

5.1 Kesimpulan

Berikut adalah kesimpulan yang dapat diambil dari penelitian ini:

3. Metode *Faster Region-based Convolutional Neural Network* (Faster R-CNN) dengan *backbone* ResNet50 dan *eXtreme Gradient Boosting* (XGBoost) berhasil diimplementasikan untuk melakukan objek deteksi polong kedelai dan klasifikasi tanaman terpilih dan tidak terpilih.
4. Model objek deteksi polong kedelai mencapai performa terbaik dengan nilai mAP sebesar 0,906 dan IoU sebesar 0,9588. Kinerja ini membuktikan kemampuan model untuk mendeteksi polong kedelai secara efektif.
5. Konfigurasi *hyperparameter* terbaik untuk model objek deteksi polong kedelai diperoleh dengan melatih seluruh jaringan (*full layers*) menggunakan *batch size* 4, *learning rate* 0,005, *momentum* 0,9, dan *weight decay* 0,0001.
6. Model klasifikasi mencapai hasil terbaik untuk kelas “Tidak Terpilih” dengan *precision* 1,00, *recall* 0,98, dan *F1-Score* 0,99, untuk kelas ”Terpilh” dengan *precision* 0,98, *recall* 1,00, dan *F1-Score* 0,99, dan akurasi sebesar 0,99.
7. Konfigurasi *hyperparameter* terbaik untuk model klasifikasi kedelai diperoleh dengan menggunakan *learning rate* = 0,01, *n_estimators* = 200, *max depth* = 3.
8. Sistem mampu melakukan deteksi dan menghitung objek polong dan klasifikasi kedelai dengan baik.

5.2 Saran

Berikut adalah saran dari penulis untuk pengembangan penelitian selanjutnya yang berkaitan dengan penelitian ini.

1. Penelitian selanjutnya disarankan untuk menambah jumlah dan variasi data latih, terutama untuk citra polong kedelai.
2. Perlu untuk mengeksplor *backbone* lain selain ResNet50, untuk melihat potensi peningkatan performa deteksi objek kecil yang lebih baik.
3. Disarankan untuk melakukan *tuning hyperparameter* yang lebih ekstensif menggunakan teknik optimasi otomatis pada objek deteksi untuk menemukan kombinasi parameter yang lebih optimal bagi model.
4. Penelitian mendatang dapat membandingkan performa Faster R-CNN dengan metode objek deteksi lain atau terbaru untuk memberikan hasil yang lebih baik lagi.
5. Sistem klasifikasi dapat mengklasifikasikan banyak tanaman sekaligus.

DAFTAR PUSTAKA

- Sjamsijah, N., Varisa, N., Kunci, K., Marshal, I., Dithane, F., & Gandasil, B. D. (2018). *Uji Daya Hasil Beberapa Genotipe Tanaman Kedelai (Glycine max (L .) Merrill) Produksi Tinggi dan Umur Genjah Generasi F6 Masyarakat Indonesia menjadikan*. 2(2), 106–116. <https://doi.org/10.25047/agriprima.v2i2.79>
- Ichwan, M., & Siti Syifa, R. (2023). MIND (Multimedia Artificial Intelligent Networking Database Klasifikasi Citra Bibit Tanaman Menggunakan Convolutional Neural Network dan Improved Feature Pyramid Network. *Journal MIND Journal | ISSN*, 8(1), 1–13. <https://doi.org/10.26760/mindjournal.v8i1.1-13>
- Firdaus, E. A., Manurung, J., Saragih, H., & Azhar, M. (2025). *Optimization of XGBoost hyperparameters using grid search and random search for credit card default prediction*. 14(2), 269–280.
- Roswita, R., Yohana, & Abdullah, S. (2020). Pertumbuhan dan Hasil Tiga Varietas Unggul Kedelai pada Lahan Sawah Tadah Hujan di Kabupaten Pasaman Barat. *Jurnal Pembangunan Nagari*, 5(2), 225–234.
- Charli, F., Syaputra, H., Akbar, M., Sauda, S., & Panjaitan, F. (2020). Implementasi Metode Faster Region Convolutional Neural Network (Faster R-CNN) Untuk Pengenalan Jenis Burung Lovebird. *Journal of Information Technology Ampera*, 1(3), 185–197. <https://doi.org/10.51519/journalita.volume1.issue3.year2020.page185-197>
- Zhang, C., Lu, X., Ma, H., Hu, Y., Zhang, S., Ning, X., Hu, J., & Jiao, J. (2023). *High-Throughput Classification and Counting of Vegetable Soybean Pods Based on Deep Learning*. <https://doi.org/10.3390/agronomy13041154>
- Kumalasari, T. (2017). *TEKNIK PEMILIHAN VARIETAS UNGGUL KEDELAI BERBASIS WEB Technique of Choosing The Soybeans Superior Varieties Based on The Web*. 6(2), 93–102. <https://doi.org/doi.org/10.31850/jgt.v6i2.214>
- Messina, M., & Messina, V. (2010). *The Role of Soy in Vegetarian Diets*. 855–888. <https://doi.org/10.3390/nu2080855>
- Pratama, I. Y., Wahab, A., & Alaydrus, M. (2021). *Deep Learning for Assessing Unhealthy Lettuce Hydroponic Using Convolutional Neural Network based on Faster R-CNN with Inception V2*. 4–9.
- Hanafiah, D. S., Lubis, K., Haryati, Setiado, H., Damanik, G. M., Limbong, M. S., Silaen, F. R., Joshua, & Lestami, A. (2023). Assembly of Soybean Genotypes Developed Through Three-Way Cross. *Sabrao Journal of Breeding and Genetics*, 55(3), 940–950. <https://doi.org/10.54910/sabrao2023.55.3.28>

- Mechanism, S. (2024). *A Crowded Object Counting System with Self-Attention Mechanism*. <https://doi.org/10.3390/s24206612>
- Chen, M., Jin, C., Ni, Y., Yang, T., & Xu, J. (2024). *A dataset of the quality of soybean harvested by mechanization for deep-learning-based monitoring and analysis*. 52. <https://doi.org/10.1016/j.dib.2023.109833>
- Bantacut, T. (2017). Pengembangan Kedelai untuk Kemandirian Pangan, Energi, Industri, dan Ekonomi. *Pangan*, 26(1), 81–96. <http://www.jurnalpangan.com/index.php/pangan/article/download/346/299>
- Ji, B., Zhou, H., Pan, L., Xie, F., Xie, Y., Li, Y., & Bai, J. (2025). *Smart Agricultural Technology*. 12(September), 0–10. <https://doi.org/10.1016/j.atech.2025.101473>
- Hamdani, K. K., & Haryati, Y. (2023). Keragaan dan hasil varietas unggul kedelai di lahan sawah tadah hujan setelah padi di Kabupaten Majalengka. *Agromix*, 14(1), 60–66. <https://doi.org/10.35891/agx.v14i1.3098>
- Algoritme, J., Alwanda, M. R., Putra, R., Ramadhan, K., Alamsyah, D., Studi, P., & Informatika, T. (2020). *Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle*. 1(1). <https://doi.org/10.35957/algoritme.v1i1.434>
- Li, H., Zhang, L., Sun, H., Rao, Z., & Ji, H. (2021). Identification of soybean varieties based on hyperspectral imaging technology and one-dimensional convolutional neural network. *Journal of Food Process Engineering*, 44(8), 1–14. <https://doi.org/10.1111/jfpe.13767>
- Yin, J., Li, W., Shen, J., Zhou, C., Li, S., Suo, J., Yang, J., Jia, R., & Lv, C. (2025). *A Diffusion-Based Detection Model for Accurate Soybean Disease Identification in Smart Agricultural Environments*. <https://doi.org/10.3390/plants14050675>
- Liew, X. Y., Hameed, N., & Clos, J. (2021). An investigation of XGBoost-based algorithm for breast cancer classification. *Machine Learning with Applications*, 6(April), 100154. <https://doi.org/10.1016/j.mlwa.2021.100154>
- Cai, W., Li, J., Xie, Z., Zhao, T., & Lu, K. (2018). Street Object Detection Based on Faster R-CNN. *2018 37th Chinese Control Conference (CCC)*, 9500–9503. <https://doi.org/10.23919/ChiCC.2018.8482613>
- Jia, X., Hua, Z., Shi, H., Zhu, D., Han, Z., Wu, G., & Deng, L. (2025). *A Soybean Pod Accuracy Detection and Counting Model Based on Improved YOLOv8*. <https://doi.org/10.3390/agriculture15060617>
- Krisdiana, R., Prasetyaswati, N., Sutrisno, I., Rozi, F., Harsono, A., & Mejaya, M. J. (2021). *Financial Feasibility and Competitiveness Levels of Soybean Varieties in Rice-Based Cropping System of Indonesia*. <https://doi.org/10.3390/su13158334>

- He, H., Ma, X., Guan, H., & Wang, F. (2023). *Recognition of soybean pods and yield prediction based on improved deep learning model*. January, 1–20. <http://doi.org/10.3389/fpls.2022.1096619>
- Qin, P., Wang, T., & Luo, Y. (2022). A review on plant-based proteins from soybean : Health benefits and soy product development. *Journal of Agriculture and Food Research*, 7, 100265. <https://doi.org/10.1016/j.jafr.2021.100265>
- Shmuel, A., Glickman, O., & Lazebnik, T. (2024). *A Comprehensive Benchmark of Machine and Deep Learning Across Diverse Tabular Datasets*. <http://arxiv.org/abs/2408.14817>
- Li, X., Wang, S., Zhao, Y., & Zhao, C. (2019). *Object Detection Using Faster R-CNN*. December, 3–7. http://noiselab.ucsd.edu/ECE228_2019/Reports/Report3.pdf
- Choi, S., & Kim, S. (2025). *Optimized Faster R-CNN with Swintransformer for Robust Multi-Class Wildfire Detection*. <https://doi.org/10.3390/fire8050180>
- Stefia, E. (2017). Struktur Anatomi Tanaman Kedelai (*Glycine max L.*). *Tesis Departemen Biologi, Fakultas Matematika Dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh November*, 109.
- Saponara, S., & Elhanashi, A. (n.d.). *Impact of Image Resizing on Deep Learning Detectors for Training Time and Model Performance*.
- Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*. <https://doi.org/10.1186/s40537-019-0197-0>
- Rahman, A., Sari, R. P., & Prawira, D. (2023). *KOMODITI HASIL PANEN BERBASIS WEBSITE (Studi Kasus : Dinas Pertanian Kabupaten Sanggau)*. 11(01), 83–91.
- Chaudhuri, A. (2024). Smart traffic management of vehicles using faster R-CNN based deep learning method. *Scientific Reports*, 14(1), 1–11. <https://doi.org/10.1038/s41598-024-60596-4>
- Putri, P. P., Adisyahputra, A., & Asadi, A. (2014). Keragaman Karakter Morfologi, Komponen Hasil, dan Hasil Plasma Nutfah Kedelai (*Glycine max L.*). *Bioma*, 10(2), 41. [https://doi.org/10.21009/bioma10\(2\).7](https://doi.org/10.21009/bioma10(2).7)
- Sumarmi, S., & Triyono, K. (2023). Pengamatan Morfologi Bagian Tanaman Lima Kultivar Kedelai [*Glycine max (L.) Merrill*]. *Bioma : Berkala Ilmiah Biologi*, 24(2), 130–137. <https://doi.org/10.14710/bioma.24.2.130-137>
- Kohli, V., & Singha, S. (2024). Protein digestibility of soybean : how processing affects seed structure , protein and non - protein components. *Discover Food*. <https://doi.org/10.1007/s44187-024-00076-w>

- Eldeen, N., Mohamed, K., & Seyedali, L. (2022). A comprehensive survey of recent trends in deep learning for digital images augmentation. *Artificial Intelligence Review*, 55(3), 2351–2377. <https://doi.org/10.1007/s10462-021-10066-4>
- Pertumbuhan, R., Glycine, K., Merrill, L., Pemberian, T., & Sampah, K. (2015). *Respons Pertumbuhan dan Produksi Kedelai (Glycine max (L.) Merrill) Terhadap Pemberian Kompos Sampah Kota dan Pupuk P*. 4(1), 1838–1848.
- Cheng, B., Wei, Y., & Shi, H. (n.d.). *Revisiting RCNN: On Awakening the Classification Power of Faster RCNN*. <https://doi.org/10.48550/arXiv.1803.0679>
- Mosahab, R., Mahamad, O., Ramayah, T., RA Nur Amalina, Ekonomi, F., Diponegoro, U., Citraluki, J., Studi, P., Fakultas, A., Dan, E., Surakarta, U. M., Efendi, P., Mandala, K., عبدالله،ماهر، Fayzollahi, S., Shirmohammadi, A., Latifian, B., 崔宇红, 楚恒亚, ... Akuntansi, J. R. (2011). Budidaya Tanaman Kedelai di Lahan Pasir Pantai. In *图书情报工作* (Vol. 4, Issue 3).
- Liu, K. (n.d.). *Soybeans as Functional Foods and Ingredients*. <https://doi.org/10.1201/9781003040286>
- Li, W. (2021). Analysis of Object Detection Performance Based on Faster R-CNN. *Journal of Physics: Conference Series*, 1827(1). <https://doi.org/10.1088/1742-6596/1827/1/012085>
- Kurniawan, M. R., Christanto, Y. H., Abdillah, G., Jenderal, U., & Yani, A. (2025). *Journal la multiapp*. 06(04), 921–933. <https://doi.org/10.37899/journallamultiapp.v6i4.2269>
- Abdullah, A., & Oothariasamy, J. (2020). Vehicle counting using deep learning models: A comparative study. *International Journal of Advanced Computer Science and Applications*, 11(7), 697–703. <https://doi.org/10.14569/IJACSA.2020.0110784>
- Leite, W. de S., Unêda-Trevisoli, S. H., da Silva, F. M., da Silva, A. J., & Di Mauro, A. O. (2018). Identification of superior genotypes and soybean traits by multivariate analysis and selection index. *Revista Ciencia Agronomica*, 49(3), 491–500. <https://doi.org/10.5935/1806-6690.20180056>
- Tonael, D. A., Kaesmetan, Y. R., & Lamabelawa, M. I. J. (2021). Selection of Superior Rice Seed Features Using Deep Learning Method. *Indonesian Journal of Artificial Intelligence and Data Mining (IJAIDM)*, 4(2), 78–88.
- Zhang, L., Bian, W., Qu, W., Tuo, L., & Wang, Y. (2021). Time series forecast of sales volume based on XGBoost. *Journal of Physics: Conference Series*, 1873(1). <https://doi.org/10.1088/1742-6596/1873/1/012067>
- Zhao, J., Kaga, A., Yamada, T., Komatsu, K., Hirata, K., Kikuchi, A., Hirafuji, M., Ninomiya, S., & Guo, W. (2023). Improved Field-Based Soybean Seed Counting and Localization with Feature Level Considered. *Plant Phenomics*, 5, 26. <https://doi.org/10.34133/plantphenomics.0026>

- Datta, A., Meghla, T. I., Khatun, T., Bhuiya, M. H., Shuvo, S. R., & Rahman, M. M. (2020). Road Object Detection in Bangladesh using Faster R-CNN: A Deep Learning Approach. *Proceedings of 2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering, WIECON-ECE 2020, April*, 348–351. <https://doi.org/10.1109/WIECON-ECE52138.2020.9397954>
- Hakim, L., Hendrawan, A., & Khoiriyah, R. (2024). *Traffic Vehicle Detection Using Faster R-CNN : A Comparative Analysis of Backbone Architectures*. 1(1), 50–62. <https://doi.org/10.63158/IJAIS.v1.i1.5>
- Panda, C. (2019). Object detection and tracking using faster R-CNN. *International Journal of Recent Technology and Engineering*, 8(3), 4894–4900. <https://doi.org/10.35940/ijrte.C5580.098319>
- Zhang, H., Wang, L., Jin, X., Bian, L., & Ge, Y. (2023). High-throughput phenotyping of plant leaf morphological, physiological, and biochemical traits on multiple scales using optical sensing. *Crop Journal*, 11(5), 1303–1318. <https://doi.org/10.1016/j.cj.2023.04.014>
- Jiang, T., Shao, M., Zhang, T., Liu, X., & Yu, Q. (2025). Soybean Pod and Seed Counting in Both Outdoor Fields and Indoor Laboratories Using Unions of Deep Neural Networks. *Lecture Notes in Computer Science, 15625 LNCS*, 17–30. https://doi.org/10.1007/978-3-031-91835-3_2
- Khairunniza-Bejo, S., Ibrahim, M. F., Hanafi, M., Jahari, M., Ahmad Saad, F. S., & Mhd Bookeri, M. A. (2024). Automatic Paddy Planthopper Detection and Counting Using Faster R-CNN. *Agriculture*, 14(9), 1567. <https://doi.org/10.3390/agriculture14091567>
- Li, M., Fu, X., & Li, D. (2020). Diabetes Prediction Based on XGBoost Algorithm. *IOP Conference Series: Materials Science and Engineering*, 768(7). <https://doi.org/10.1088/1757-899X/768/7/072093>
- Girshick, R. (2015). Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision, 2015 Inter*, 1440–1448. <https://doi.org/10.1109/ICCV.2015.169>
- Saragih, Benny Winson Maryanto Setyowati, Nanik, Prasetyo Nurjanah, U. (2019). Optimasi Lahan Pada Sistem Tumpang Sari Jagung Manis. *Jurnal Agroqua*, 17(2), 115–125. <https://doi.org/10.32663/ja.v>