

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kelapa sawit merupakan komoditas perkebunan strategis yang memiliki peran penting dalam perekonomian Indonesia. Berdasarkan data Kementerian Pertanian Republik Indonesia, luas areal perkebunan kelapa sawit nasional telah mencapai lebih dari 16,8 juta hektar dengan produksi *Crude Palm Oil* (CPO) sebesar 46,8 juta ton pada tahun 2023, menjadikan Indonesia sebagai produsen sekaligus eksportir minyak sawit terbesar di dunia [1]. Dengan luas perkebunan kelapa sawit terbesar di Indonesia, Kabupaten Rokan Hulu adalah pusat produksi utama di provinsi Riau, dengan puluhan pabrik kelapa sawit yang beroperasi secara aktif [2]. Besarnya skala produksi ini menempatkan efisiensi dan pengendalian kualitas di setiap tahapan proses, mulai dari pemanenan hingga pengolahan, sebagai faktor penentu daya saing industri kelapa sawit Indonesia di pasar global.

Peron Kamila merupakan salah satu tempat pengumpulan dan penjualan Tandan Buah Segar (TBS) kelapa sawit yang berdiri sejak tahun 2008. Peron ini berperan sebagai tempat transaksi antara petani dan pengepul sebelum TBS didistribusikan ke pabrik kelapa sawit untuk diolah lebih lanjut. Dalam kegiatan operasionalnya, Peron Kamila menerima sekitar 2 ton TBS per hari dari berbagai petani. Oleh karena itu, proses penentuan tingkat kematangan buah menjadi aspek penting karena memengaruhi kualitas TBS serta harga jual yang diterima.

Ketergantungan pada tenaga manusia dalam proses sortasi TBS berpotensi menimbulkan kesalahan, terutama ketika terjadi lonjakan volume produksi dan pekerjaan dilakukan secara berulang (*repetitif*), sehingga meningkatkan risiko terjadinya *human error* dalam menentukan tingkat kematangan buah kelapa sawit. Penelitian oleh [3] menggunakan metode *Human Error Assessment and Reduction Technique (HEART)*, sebuah metode yang diperkenalkan oleh Williams (1985) untuk mengkuantifikasi risiko kesalahan manusia berdasarkan reliabilitas kerja. Hasil penelitian tersebut menunjukkan bahwa nilai *Human Error Probability (HEP)* yang menunjukkan tingkat risiko kesalahan yang tinggi pada proses sortasi TBS dengan potensi kesalahan pengerjaan ulang (*rework*) terbesar terjadi pada Task 2 sebesar 1,10592. Tingginya angka *HEP* ini dipengaruhi oleh karakteristik pekerjaan sortasi yang memiliki tingkat kesulitan tinggi dan *repetitif*. Kesalahan klasifikasi ini berdampak langsung pada penurunan kualitas produk; buah mentah yang terolah menurunkan rendemen minyak, sementara buah lewat matang meningkatkan kadar asam lemak bebas (*ALB*) melebihi 5%. Hal ini menciptakan kerugian finansial yang signifikan bagi perusahaan akibat ketidakkonsistenan mutu *CPO*.

Berdasarkan kondisi tersebut, permasalahan utama dalam penelitian ini adalah tingginya potensi *human error* dalam proses sortasi TBS yang menyebabkan ketidakkonsistenan dalam penentuan tingkat kematangan buah kelapa sawit, sehingga berdampak pada kualitas dan nilai ekonomis *CPO* yang dihasilkan. Berdasarkan hasil wawancara, penentuan tingkat kematangan buah di Peron Kamila masih dilakukan secara visual dengan memperhatikan karakteristik

fisik buah kelapa sawit. Proses penilaian yang masih dilakukan secara manual sangat bergantung pada pengalaman dan ketelitian. Dalam pelaksanaannya, masih ditemukan ketidaksesuaian dalam penentuan tingkat kematangan buah, sehingga berpotensi menyebabkan kesalahan klasifikasi. Selain itu, proses sortasi buah kelapa sawit masih banyak dilakukan secara manual sehingga berpotensi menimbulkan kesalahan dalam penentuan tingkat kematangan buah. Oleh karena itu, diperlukan suatu sistem berbasis *Computer Vision* yang mampu mendeteksi dan mengklasifikasikan tingkat kematangan buah kelapa sawit secara otomatis serta mendukung implementasi secara *real-time* untuk meningkatkan efisiensi proses sortasi.

Menanggapi tingginya risiko kesalahan manusia tersebut, pemanfaatan metode *Object Detection* dalam bidang *Computer Vision* telah banyak diteliti sebagai solusi yang lebih objektif dan efisien. Berbeda dengan pendekatan klasifikasi citra konvensional, *Object Detection* memungkinkan sistem untuk mengidentifikasi sekaligus menentukan lokasi objek dalam suatu citra melalui *bounding box*. Penelitian oleh [4] menunjukkan bahwa model *YOLO11n* mampu memberikan performa deteksi yang baik pada proses identifikasi objek secara *real-time*. Hasil penelitian ini berupa model *YOLO11n* yang telah dilatih menggunakan dataset *Commercial Marine Fish Species*, dengan menggunakan notebook Google Colab dan bahasa pemrograman *Python*. Selanjutnya, penelitian oleh [5] membandingkan beberapa varian *YOLO11n* dan menyimpulkan bahwa *YOLO11n* memberikan keseimbangan terbaik antara akurasi deteksi dan efisiensi waktu inferensi, sehingga sesuai diterapkan pada perangkat dengan sumber daya

komputasi terbatas. Selain itu, kajian komprehensif yang dilakukan oleh [6] menjelaskan bahwa *YOLO11n* merupakan pengembangan arsitektur terbaru yang menghadirkan peningkatan pada proses ekstraksi fitur dan efisiensi model dibandingkan generasi sebelumnya, sehingga mampu meningkatkan performa deteksi objek pada berbagai skenario aplikasi. Berdasarkan berbagai penelitian terdahulu, metode *Object Detection* berbasis *YOLO11n* menunjukkan kemampuan yang baik dalam mendeteksi dan mengklasifikasikan objek pada berbagai bidang aplikasi. Oleh karena itu, penelitian ini menggunakan model *YOLO11n* sebagai metode deteksi objek karena merupakan salah satu varian *YOLO11n* yang dirancang untuk mendukung proses deteksi secara efisien dengan kebutuhan komputasi yang relatif ringan. Model ini diharapkan mampu mendeteksi dan mengklasifikasikan tingkat kematangan buah kelapa sawit ke dalam kategori mentah, matang, dan lewat matang secara otomatis pada sistem berbasis web.

Namun demikian, sebagian besar penelitian tersebut masih terbatas pada tahap pengujian model dan belum banyak diimplementasikan dalam bentuk sistem yang dapat digunakan secara langsung di lingkungan industri, khususnya untuk kebutuhan *real-time*. Oleh karena itu, penelitian ini diwujudkan melalui judul **"Sistem Klasifikasi Kematangan Buah Kelapa Sawit Berbasis *Computer Vision* Menggunakan *Object Detection*"**. Sistem ini dirancang untuk memberikan umpan balik visual secara langsung melalui visualisasi *bounding box*, sehingga diharapkan dapat membantu operator dalam memvalidasi hasil sortasi, mengurangi tingkat *human error*, serta meningkatkan konsistensi mutu *Crude Palm Oil (CPO)*.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka dapat dirumuskan suatu masalah pada penelitian ini, yaitu “Bagaimana merancang dan membangun sistem klasifikasi kematangan buah kelapa sawit berbasis *computer vision* menggunakan metode *object detection*?”

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah untuk merancang dan membangun sistem klasifikasi kematangan buah kelapa sawit berbasis *computer vision* dengan menggunakan metode *object detection*.

1.4 Batasan Masalah

Adapun batasan masalah pada penelitian ini, yaitu:

1. Metode yang digunakan dalam penelitian ini adalah metode *object detection* untuk mengklasifikasikan kematangan buah kelapa sawit berdasarkan citra.
2. Sistem yang dibangun mengklasifikasikan kematangan buah kelapa sawit ke dalam tiga kategori yaitu mentah/*Unripe* (tidak layak), matang/*Ripe* (layak), dan lewat matang/*Overripe* (perlu segera diolah).
3. Proses pelatihan model dalam penelitian ini dilakukan menggunakan bahasa pemrograman *Python*, dan hasil pelatihan tersebut selanjutnya diimplementasikan ke dalam sistem berbasis website.
4. Hasil dari sistem ditampilkan dalam bentuk *visualisasi bounding box* pada objek buah kelapa sawit, yang dilengkapi dengan informasi tingkat kematangan serta nilai *confidence* dari hasil klasifikasi.

5. Dataset yang digunakan berupa citra kematangan buah sawit terdiri dari 9000 citra sebagai data latih dan data uji.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah untuk membantu proses penentuan tingkat kematangan buah kelapa sawit secara otomatis melalui sistem berbasis *computer vision* dengan metode *object detection* yang diimplementasikan dalam bentuk website.

1.6 Sistematika Penulisan

Sistematika penulisan pada penelitian ini terdiri dari pokok-pokok permasalahan yang dibahas dan diuraikan menjadi beberapa bagian :

BAB 1 PENDAHULUAN

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

BAB 2 LANDASAN TEORI

Bab ini berisi teori-teori yang digunakan pada penelitian ini. Teori-teori yang berhubungan dengan pengembangan sistem yang akan dibangun.

BAB 3 METODOLOGI PENELITIAN

Bab ini berisi tahapan–tahapan dalam pengumpulan data, perancangan sistem, perumusan masalah dan analisa.

BAB 4 ANALISA DAN PERANCANGAN

Bab ini berisi analisa dan pengembangan sistem yang akan dibangun atau dikembangkan.

BAB 5 IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi implementasi dari analisa dan perancangan dan pengujian pada sistem yang berhasil dibangun.

BAB 6 PENUTUP

Bab ini berisi rangkuman dari hasil penelitian yang telah dilakukan dan saran-saran untuk pengembangan sistem atau penelitian selanjutnya.

BAB 2

LANDASAN TEORI

2.1 Kelapa Sawit (*Elaeis guineensis* Jacq)

Kelapa sawit (*Elaeis guineensis* Jacq) merupakan tanaman industri yang sangat vital bagi perekonomian Indonesia, khususnya di Provinsi Riau, serta banyak dibudidayakan di kawasan Asia Tenggara seperti Indonesia, Malaysia, dan Thailand. Selain itu, dibandingkan dengan tanaman lain seperti rapeseed, bunga matahari, dan kedelai, kelapa sawit memiliki keunggulan dalam menghasilkan minyak dalam jumlah yang lebih besar, sehingga kualitas hasil olahannya sangat ditentukan oleh kondisi bahan baku yang diterima oleh pabrik [7].

Industri kelapa sawit Indonesia mencakup seluruh rantai nilai dari pembibitan, budidaya, panen, pengolahan di pabrik kelapa sawit (PKS), hingga ekspor *CPO* (*Crude Palm Oil*) dan *CPKO* (*Crude Palm Kernel Oil*). Setiap tahapan dalam rantai nilai ini memiliki parameter kualitas yang saling bergantung, dengan kualitas TBS yang diterima pabrik menjadi titik kritis yang menentukan mutu keseluruhan produk akhir [8].

Aspek agronomi dan manajemen panen kelapa sawit tidak dapat dipisahkan dari aspek pengolahan karena karakteristik biokimiawi buah pada saat pemanenan secara langsung memengaruhi rendemen minyak dan mutu *CPO* yang dihasilkan. Pernyataan ini menegaskan pentingnya sistem sortasi berbasis teknologi untuk menjamin konsistensi kualitas TBS yang masuk ke jalur pengolahan pabrik [9].

2.1.1 Karakteristik Kematangan Tandan Buah Segar (TBS)

Standar Nasional Indonesia mengenai Tandan Buah Segar (SNI 01-3935-1995) menetapkan klasifikasi kematangan buah kelapa sawit berdasarkan jumlah brondolan yang terlepas secara alami dari tandan, di mana standar operasional ini diadopsi secara luas dalam proses penentuan mutu visual kelapa sawit [10], [11]. Brondolan adalah buah-buah individual yang lepas dari tandan akibat proses pematangan fisiologis.

Penentuan tingkat kematangan Tandan Buah Segar (TBS) kelapa sawit merupakan tahapan penting dalam proses pemeriksaan kualitas buah sebelum dipasarkan ke pabrik pengolahan. Penilaian kematangan umumnya dilakukan berdasarkan karakteristik visual, seperti perubahan warna kulit buah dan jumlah brondolan yang terlepas dari tandannya.

1. Buah Mentah (*Unripe*): Buah mentah ditandai dengan brondolan yang lepas kurang dari 4 buah dari tandannya. Secara kimiawi, kandungan minyak dalam mesokarp masih sangat rendah, sehingga pengolahan buah mentah akan mengakibatkan rendahnya rendemen minyak yang dihasilkan [12]. Karakteristik buah kelapa sawit mentah dapat dilihat pada Gambar 2.1 [13]



Gambar 2. 1 Buah Kelapa Sawit Mentah (*Unripe*)

Sumber: Kaggle, *Dataset PCD Klasifikasi Sawit CNN*, 2024 [13]

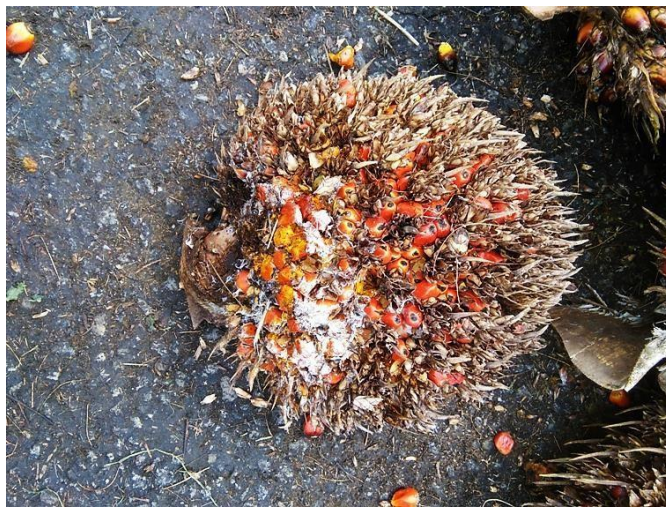
2. Buah Matang (*Ripe*): Memiliki warna permukaan merah oranye yang cerah. Pada kondisi ini, tandan buah telah memenuhi syarat untuk diolah karena kandungan minyaknya telah mencapai titik optimal dengan kadar asam lemak bebas yang masih terkontrol. Buah berwarna merah mengkilat dan lepas dari tandan sebanyak 26-50% [11]. Karakteristik buah kelapa sawit matang dapat dilihat pada Gambar 2.2



Gambar 2. 2 Buah Kelapa Sawit Matang (*Ripe*)

Sumber: Dokumentasi Pribadi, 2026

3. Buah Lewat Matang (*Overripe*): Ditandai dengan warna jingga kemerahan dan lebih dari 50 buah terlepas. Tandan buah segar pada kondisi terlalu matang masih dapat menghasilkan minyak, namun kualitas minyak dapat menurun apabila proses panen dan pengolahan tidak dilakukan secara optimal [10]. Karakteristik buah kelapa sawit lewat matang dapat dilihat pada Gambar 2.3 [13]



Gambar 2. 3 Buah Kelapa Sawit Lewat Matang (*Overripe*)

Sumber: Kaggle, *Dataset PCD Klasifikasi Sawit CNN*, 2024 [13]

2.2 *Computer Vision*

2.2.1 Definisi dan Konsep Dasar

Computer vision secara umum merupakan ilmu dan teknologi yang mempelajari bagaimana suatu *machine* atau sistem dapat “melihat” dan memahami informasi visual melalui gambar atau video, seperti pengenalan objek (*recognition*), estimasi gerak (*motion*), rekonstruksi adegan (*scene reconstruction*), serta pemulihan citra (*image restoration*) [14].

Dengan kemajuan teknologi saat ini, banyak alat yang dapat membantu pekerjaan manusia secara otomatis. *Computer vision* adalah salah satu bidang ilmu yang membantu. *Computer vision* dapat digunakan untuk mengenali gambar dari objek dan memanfaatkannya [15].

Kemajuan signifikan dalam bidang *computer vision* dalam beberapa tahun terakhir dipengaruhi oleh perkembangan *deep learning*, khususnya *YOLOv8n* yang mampu mempelajari fitur citra secara otomatis dari data gambar. Perkembangan tersebut juga mendukung penerapan *object detection* untuk meningkatkan efisiensi dan akurasi dalam proses klasifikasi citra.

2.2.2 Visualisasi *Bounding Box*

Bounding box merupakan representasi visual berupa kotak persegi panjang yang membungkus objek hasil deteksi pada citra. Metode ini umum digunakan dalam sistem deteksi objek pada *computer vision* dan terbukti efektif dalam mendeteksi elemen visual pada gambar yang kompleks [16].

1. Kotak berwarna hijau menandakan TBS yang telah terdeteksi sebagai matang (layak angkut).
2. Kotak berwarna merah menandakan TBS yang terdeteksi sebagai mentah (tidak layak angkut).

Umpan balik visual berupa *bounding box* berwarna ini membantu operator melakukan verifikasi cepat terhadap potensi kesalahan penyortiran secara realistis dan memudahkan pengamatan langsung pada hasil deteksi sistem. Pendekatan visualisasi serupa telah diterapkan pada penelitian-penelitian penerapan *computer*

vision dan *deep learning* untuk deteksi kematangan buah atau TBS kelapa sawit [17].

2.2.3 Citra Digital

Pengolahan citra digital adalah bidang ilmu yang mempelajari bagaimana membentuk, memproses, dan menganalisis gambar untuk menghasilkan data yang dapat dimengerti dan digunakan manusia [18].

Citra digital terdiri dari kumpulan titik yang disebut piksel (juga dikenal sebagai piksel atau elemen gambar) [19]. Setiap piksel diwakili oleh satu kotak kecil dengan koordinat posisinya. Citra juga dikenal sebagai gambar, adalah fungsi dua dimensi $f(x, y)$ yang diperoleh dari peralatan sensor yang mencatat harga ciri gambar pada semua titik elemen (x, y) . Sebuah ruang diskrit berdimensi dua yang berasal dari citra analog $a(x, y)$ menjadi citra kontinu dua dimensi disebut digitalisasi. Gambar dua dimensi kontinyu $a(x, y)$ dibagi menjadi N baris dan M kolom. Perpotongan baris dan kolom ini disebut piksel. Integer koordinat $[m, n]$ dengan $\{m = 0, 1, 2, \dots, M - 1\}$ dan $\{n = 0, 1, 2, \dots, N - 1\}$ memiliki nilai $[m, n]$. Dalam banyak kasus, gambar (x, y) dapat dianggap sebagai sinyal fisik dari sensor dua dimensi. Banyak variabel, seperti kedalaman (z), warna (λ), dan waktu (t), membentuk sinyal ini [20].

2.2.4 Tipe-tipe File Citra Digital

File citra memiliki format yang beragam. Berikut adalah beberapa standar format file gambar yang paling umum digunakan [20].

1. Tagged Image Format (.tif, .tiff). Jenis kompresi yang sangat luas, format tiff mendukung berbagai skema kompresi, tetapi sering digunakan untuk

gambar yang tidak dikompresi. Itu juga dapat menangani bitmap hingga gambar palet warna terkompresi. Sederhana, memiliki warna, dan populer.

2. JPEG (.jpg) adalah format standar yang paling umum untuk mengirimkan gambar. Baik JPEG maupun JPG adalah gambar raster, yang berarti mereka bukan vector. Memiliki garis dan bentuk yang sangat tidak jelas dan gradasi warna atau bayangan yang halus adalah beberapa ciri khasnya. Gambar jenis ini mengalami masalah kualitas saat ukurannya diperbesar karena sifatnya yang berbasis piksel. Ini karena gambar menjadi bergerigi dan mengekspos setiap piksel. Resolusi yang diukur dalam titik per inci (dpi) biasanya didefinisikan dan ditampilkan. Gambar raster yang lebih jernih memiliki dpi yang lebih tinggi. Gambar raster lebih mudah ditemukan di internet karena menjadi standar Web.
3. *Portable Network Graphics* (.png) adalah format file yang mudah digunakan yang memungkinkan penyimpanan gambar raster dengan kompresi yang baik dan tanpa kehilangan detail. TIFF Dengan skala abu-abu, palet warna, dan dukungan untuk gambar dengan warna asli.
4. MPEG (.mpg) khusus untuk gambar bergerak, banyak digunakan di internet. Selain itu, proses kompresi biasanya digunakan untuk jenis file ini.
5. *Graphics Interchange Format* (.gif). Gambar palet warna 8-bit yang dapat digunakan di berbagai platform dan aplikasi didukung oleh format ini. Sebagian besar digunakan untuk gambar berkualitas tinggi dan logo. Data gambar animasi dapat dimasukkan ke dalam file GIF.

6. RGB (.rgb) adalah standar Silicon Graphics untuk gambar berwarna.
7. RAS (.ras) adalah file gambar yang dihasilkan dari scan tiga pita warna untuk gambar Sun Raster tanpa kompresi.
8. *Postscript* (.ps, .eps, .epsf). Untuk mencetak dan memasukkan gambar atau gambar ke dalam buku atau catatan, format gambar ini terutama digunakan. Angka desimal atau heksadesimal yang dikodekan dalam ASCII digunakan untuk menggambarkan tingkat abu-abu dalam format postscript.
9. *Portable Image*. Termasuk *Bitmap*, *Graymap*, *Pixmap*, dan *Network Map* Portabel. Format.pbm,.pgm,.ppm, dan.pnm adalah ekstensi standar format ini, yang mendukung berbagai jenis gambar dengan kompleksitas yang meningkat, mulai dari bit hingga tingkat abu-abu hingga pixmap warna yang beragam.
10. *PPM*. File *PPM* terdiri dari dua bagian: header dan data gambar. Meskipun file jenis ini sudah jarang ditemukan akhir-akhir ini, Anda masih dapat menemukan file dengan ekstensi *PPM* di internet. Pada akhir 1980-an, format *PPM* dibuat untuk mempermudah penyebaran gambar di berbagai platform.
11. *PGM*. Format ini mirip dengan *PPM*, tetapi tidak menyimpan informasi skala abu-abu (satu nilai per piksel daripada tiga (RGB)).
12. *PBM*. format file *monokrom* paling dasar. Awalnya dibuat sebagai bahasa umum untuk keluarga besar filter konversi gambar bitmap.

2.3 Deteksi Objek (*Object Detection*)

Dalam bidang visi komputer, deteksi objek merupakan sebuah metode yang berfungsi untuk mengenali berbagai entitas, seperti buah, kendaraan, hingga manusia, baik melalui media gambar maupun video. teknik ini bekerja dengan cara memberikan kotak pembatas (*bounding box*) pada objek yang dikenali guna mengidentifikasi posisi koordinatnya. secara umum, metodologi deteksi objek diklasifikasikan ke dalam dua pendekatan utama, yaitu berbasis *machine learning* dan berbasis *deep learning* [21].

Pendeteksian objek, juga dikenal sebagai pendeteksian objek, adalah teknologi komputer yang berkaitan dengan pemrosesan gambar dan visi komputer yang berfungsi untuk mendeteksi suatu objek dalam gambar digital, yang dapat berupa warna dan bentuk objek. Pendeteksian objek baru-baru ini menjadi salah satu bidang yang paling menarik dalam visi komputer [14].

Dalam sistem ini, *bounding box* digunakan sebagai representasi visual hasil deteksi kematangan TBS kepada pekerja sortasi, di mana kotak hijau menandakan TBS matang/*Ripe* (layak), kotak merah menandakan TBS mentah/*Unripe* (tidak layak), dan kotak oranye menandakan TBS lewat matang/*Overripe* (perlu segera diolah), sehingga operator dapat langsung melakukan verifikasi secara visual terhadap potensi kesalahan penyortiran secara *real-time*.

2.4 *YOLO11n*

YOLO (*You Only Look Once*) adalah anggota keluarga algoritma deteksi objek yang terkenal sangat akurat dan cepat. *YOLO* menggunakan satu proses untuk memproses gambar untuk memprediksi bounding box dan kelas objek secara bersamaan, berbeda dengan detektor dua tahap. Komunitas mengembangkan iterasi *YOLOv11*, yang meningkatkan arsitektur dan fleksibilitas sambil mempertahankan prinsip efisiensi dari versi sebelumnya. Varian *YOLOv11n* digunakan dalam penelitian ini adalah model terkecil dan tercepat, dan cocok untuk aplikasi yang membutuhkan inferensi waktu nyata [22].

YOLO11 memiliki beberapa varian model, yaitu *YOLO11n* (*Nano*), *YOLO11s* (*Small*), *YOLO11m* (*Medium*), *YOLO11l* (*Large*), dan *YOLO11x* (*Extra Large*). Setiap varian memiliki jumlah parameter dan kebutuhan komputasi yang berbeda sehingga dapat disesuaikan dengan kebutuhan implementasi sistem [4].

YOLOv11 mengadopsi konsep dasar yang mirip dengan pendahulunya. Sistem ini memproses gambar masukan dengan membaginya menjadi kisi-kisi berukuran $S \times S$. Apabila titik tengah suatu objek terletak dalam suatu sel kisi, maka sel tersebut akan bertugas membuat prediksi terkait objek itu [23].

2.5 *Arsitektur YOLO11n*

YOLO11, yang dirilis oleh tim *Ultralytics* pada tanggal 30 September 2024, adalah penambahan terbaru dalam seri *YOLO* (*You Only Look Once*). *YOLO11* sangat cocok untuk berbagai tugas *computer vision*, seperti *object detection* dan pelacak objek, *segmentation*, *image classification*, and *pose estimation* [24].

1. *Backbone*

Backbone merupakan arsitektur *YOLO11n* adalah bagian pertama yang melakukan ekstraksi fitur dari gambar masukan. Pada tahap ini, gambar diproses untuk mengekstrak detail penting yang akan digunakan dalam fase deteksi, seperti bentuk, tekstur, pola, dan atribut objek. *Backbone* pada *YOLO11n* terdiri atas beberapa modul, yaitu *Convolution Block (CBS)*, *Cross-Stage Feature Enhancement (C3k2)*, *Spatial Pyramid Pooling Fast (SPPF)*, dan *Cross-Stage Partial Pyramid Slice Attention (C2PSA)*, modul-modul ini beroperasi secara berurutan untuk menghasilkan representasi fitur yang lebih baik [24]. Sebelum melanjutkan ke langkah *Feature Fusion* dalam studi ini, *backbone* membantu mengekstrak sifat visual dari tandan buah sawit. Model *YOLO11n* beroperasi berdasarkan hasil ekstraksi.

2. *Feature Fusion (Neck)*

Feature Fusion (Neck) merupakan komponen pada arsitektur *YOLO11n* yang berfungsi menggabungkan *feature map* dari berbagai tingkat skala sehingga hubungan antarfitur menjadi lebih optimal. Menurut penelitian [24], Neck pada *YOLO11n* terdiri atas modul Upsample, *Concat*, dan *C3k2*. Proses Upsample digunakan untuk menyesuaikan ukuran feature map beresolusi rendah dengan feature map beresolusi tinggi, sehingga informasi detail dari lapisan awal dapat digabungkan dengan informasi semantik dari lapisan yang lebih dalam. Selanjutnya, proses *Concat* menggabungkan feature map dari berbagai skala pada dimensi kanal (channel) sehingga informasi yang diperoleh menjadi lebih lengkap sebelum diteruskan ke tahap *Detection Head*.

Pada penelitian ini, *Feature Fusion* berperan dalam menggabungkan informasi fitur dari berbagai skala sehingga model *YOLO11n* mampu mendeteksi tandan buah kelapa sawit dengan karakteristik yang beragam. Hasil penggabungan fitur tersebut kemudian diteruskan ke *Detection Head* untuk proses prediksi kelas dan lokasi objek.

3. *Detection Head*

Detection Head merupakan komponen akhir pada arsitektur *YOLO11n* yang berfungsi menghasilkan prediksi objek berdasarkan fitur yang telah diproses oleh *Backbone* dan *Feature Fusion (Neck)*. *YOLO11n* menggunakan *multi-scale prediction head* untuk mendeteksi objek dengan berbagai ukuran. Prediksi dilakukan pada tiga skala feature map, yaitu P3, P4, dan P5, sehingga model mampu mendeteksi objek berukuran kecil, sedang, maupun besar secara lebih optimal [24].

2.6 Matriks Evaluasi (*Confusion Matrix*)

Confusion Matrix merupakan teknik yang digunakan untuk mengevaluasi model klasifikasi (*supervised learning*). *Confusion matrix* berbentuk tabel matriks yang menggambarkan kinerja model dengan memberikan informasi perbandingan hasil prediksi sistem terhadap hasil kelas yang sebenarnya [25]. Salah satu istilah untuk *confusion matrix* adalah *error matrix*, yang menunjukkan kinerja suatu algoritme dalam bentuk tabel tertentu [26]. Evaluasi dan validasi hasil dihitung menggunakan rumus akurasi, *precision*, *recall* dan *f measure* berikut ini:

1. Akurasi

Perhitungan akurasi dilakukan dengan membagi jumlah data yang diklasifikasi secara akurat dengan jumlah total sampel data uji coba.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.1)$$

2. Precision

Jumlah data benar yang bernilai positif (*True Positive*) dibagi dengan jumlah data salah yang bernilai positif (*False Negative*) dan dibagi dengan jumlah data benar yang bernilai positif (*True Positive*).

$$Precision = \frac{TP}{TP+FP} \quad (2.2)$$

3. Recall

Dihitung dengan membagi data benar yang bernilai positif (*True Positive*) dengan total dari data benar yang bernilai negatif (*False Negative*).

$$Recall = \frac{TP}{TP+FN} \quad (2.3)$$

4. F1-Score

Nilai *F1-Score* diperoleh dengan membagi hasil perkalian *precision* dan *recall* dengan hasil penjumlahan *precision* dan *recall*, lalu dikalikan dua.

$$F1-Score = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (1)$$

2.7 Website

Definisi website adalah bagian internet yang memungkinkan pengguna mengakses, mencari, berbagi informasi, dan gambar [27]. Website dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan

melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia. Secara umum, website dibagi menjadi 3 jenis, yaitu website statis, dinamis, dan interaktif [28]. Aplikasi berbasis web memiliki keunggulan dalam hal aksesibilitas, karena tidak memerlukan instalasi perangkat lunak khusus di sisi klien dan dapat diakses dari berbagai perangkat [29].

2.8 Unified Modeling Language (UML)

UML adalah alat/model untuk merancang perangkat lunak berorientasi objek. *UML* juga menyediakan standar penulisan untuk membuat desain sistem, termasuk konsep proses bisnis, penulisan bahasa pemrograman, diagram basis data, dan komponen-komponen yang diperlukan dalam perangkat lunak [30].

2.8.1 Use Case Diagram

Dalam *UML*, terdiri dari beberapa diagram yang digunakan untuk pemodelan terhadap sistem yang akan dirancang, salah satunya yaitu *Use case Diagram*. *Use case Diagram* ini merupakan gambaran dari interaksi antara satu atau lebih aktor dari sistem dan juga digunakan untuk mengetahui fungsi apa saja yang ada pada sistem informasi yang dibuat [31].

2.8.2 Class Diagram

Class Diagram adalah jenis diagram struktur statis dalam *UML* yang menggambarkan struktur sistem dengan menunjukkan sistem Class, atributnya, metode, dan hubungan antar objek. *Class Diagram* disebut jenis diagram struktur karena menggambarkan apa yang harus ada dalam sistem yang dimodelkan dengan berbagai komponen [32].

2.8.3 *Activity Diagram*

Setelah *use case diagram* dibuat, selanjutnya memodelkan langkah-langkah dari setiap aktivitas yang dilakukan dalam diagram aktivitas. *Activity diagram* bertujuan untuk menjelaskan bagaimana urutan-urutan kegiatan pada suatu proses sistem, baik pada proses kegiatan yang sifatnya berurutan maupun paralel [31].

2.8.3 *Sequence Diagram*

Sequence diagram dibuat berdasarkan *use case diagram* dan *activity diagram* yang telah dibuat. Kedua diagram tersebut selanjutnya akan digunakan sebagai input untuk *sequence diagram* [31].

2.9 *Flowchart*

Flowchart adalah gambaran berupa grafik yang memiliki urutan suatu proses atau langkah-langkah secara sistematis untuk menjalankan suatu program. *Flowchart* dapat memberi gambaran untuk dilakukan proses analisis, perancangan, pengkodean dalam memecahkan masalah yang lebih terperinci dalam proses operasional suatu kegiatan [33]. Penggunaan simbol standar dalam *flowchart* memudahkan pengembang dalam memahami dan menganalisis alur logika program sebelum tahap pengkodean dimulai, sehingga dapat membantu dalam proses identifikasi kesalahan (*debugging*) sejak tahap perancangan [34]. *Flowchart* juga berfungsi sebagai dokumen panduan yang membantu pengembang lain dalam memahami alur berpikir algoritma yang diimplementasikan, karena mampu merepresentasikan proses secara visual dan terstruktur [35].

2.10 Perangkat Lunak Pendukung

2.10.1 PHP (*Hypertext Preprocessor*)

PHP adalah bahasa pemrograman yang digunakan untuk membuat aplikasi web dinamis. *PHP* dapat berjalan pada berbagai platform termasuk *Linux*, *Unix*, *Windows*, dan *macOS* serta terintegrasi secara langsung dengan *HTML*. *PHP* dieksekusi di sisi server (*server-side scripting*) sehingga browser client hanya menerima hasil keluaran *HTML* yang telah diproses, meningkatkan keamanan kode sumber aplikasi [36].

2.10.2 XAMPP

XAMPP adalah web server open source yang berjalan pada sistem operasi cross-platform (*Windows*, *Linux*, *MacOS*). Semua yang diperlukan untuk mengelola website tersedia di *XAMPP* seperti *Apache*, *MySQL/MariaDB*, *PHP*, dan *Perl*. Meski program di dalamnya lengkap, *XAMPP* tetap merupakan web server yang sederhana dan ringan [37].

2.10.3 Draw.io

Draw.io adalah sebuah website yang didesain khusus untuk menggambarkan diagram *UML* secara online. Dimana punya tampilan yang sangat responsive dan terintegrasi dengan layanan penyimpanan file milik google yaitu Google Drive sehingga *draw.io* menjadi alternatif dalam pembuatan diagram *UML* dengan waktu yang lebih singkat [38].

2.11 Penelitian Terkait

No	Nama	Tahun	Judul	Hasil
1.	Akbar ddk.	2025	Implementasi Yolo11 Untuk Deteksi Penyakit Tanaman Padi Berdasarkan Citra Daun	Penelitian mengimplementasikan YOLO11 untuk mendeteksi penyakit tanaman padi berdasarkan citra daun dan memperoleh akurasi sebesar 95%, sehingga menunjukkan performa klasifikasi yang baik.
2.	Permatasari dan Putra	2025	Implementasi Algoritma YOLO11 dalam Mendeteksi Spesies Ikan Laut Komersial secara Real Time untuk Sistem Penyortiran Ikan	Penelitian mengimplementasikan algoritma YOLO11 untuk mendeteksi spesies ikan laut komersial secara real-time. Model terbaik diperoleh menggunakan YOLO11n dengan pembagian data 70% data training, 20% data validation, dan 10% data testing, menghasilkan mAP 95,1%, precision 93,2%, dan recall 91,0%, sehingga menunjukkan performa deteksi yang tinggi.
3.	Armantya dkk.	2025	Sistem Deteksi dan Pengenalan Plat Nomor Kendaraan Menggunakan YOLOv11 dan PaddleOCR	Penelitian berhasil mengembangkan sistem Automatic License Plate Recognition (ALPR) menggunakan YOLOv11 untuk mendeteksi plat nomor dan PaddleOCR untuk mengenali karakter. Model dilatih menggunakan 2.926 citra dari Roboflow Universe dan memperoleh mAP50 sebesar 88,3%, precision 86,9%, serta recall 88,6%. Sistem diimplementasikan berbasis web menggunakan Flask dan ReactJS sehingga mampu mendeteksi serta mengenali plat nomor kendaraan dengan baik.

4.	Putri, Diyasa, & Pratama	2025	Deteksi Kematangan Kelapa Sawit Menggunakan Model YOLOv8 pada Gambar Digital	Model YOLOv8 yang dilatih selama 50 epoch berhasil mengklasifikasikan 4 kelas kematangan sawit dengan sangat baik, menghasilkan rata-rata precision 96,5%, recall 84,9%, mAP@0.50 sebesar 95,8%, dan mAP@0.50-0.95 sebesar 82,8%. Model ini memiliki kecepatan inferensi yang sangat cepat yaitu 10,3 milidetik per citra.
5.	N. Fitriani	2024	Sistem Deteksi Otomatis Penggunaan Senjata Tajam Menggunakan YOLOv8n Deep Learning	Penelitian ini menggunakan YOLOv8n untuk mendeteksi tiga jenis senjata tajam (pisau, sabit, dan parang). Model dilatih selama 100 epoch menggunakan 2.160 data training. Evaluasi model menghasilkan nilai Precision sebesar 0,865, Recall 0,815, dan mAP50 sebesar 0,895. Pengujian secara real-time menunjukkan respon yang sangat responsif pada jarak 1–4 meter.
6.	Ardiansyah, K. N. Qodri, D. A. Banna, & M. Z. Al-Baihaqi	2024	Pemanfaatan SAM dan YOLOv8 untuk Deteksi dan Segmentasi pada Citra MRI Tumor Otak	Menggunakan varian YOLOv8n-seg untuk deteksi objek sekaligus instance segmentation pada citra MRI tumor otak (glioma, meningioma, dan pituitary). Integrasi ini menghasilkan akurasi koordinat (box) dengan Precision 86%, Recall 87%, dan mAP50 sebesar 89%. Pada pengujian data uji, visualisasi deteksi berhasil mencapai akurasi hingga 95% pada kelas Glioma
7.	K. D. Darajat	2024	Sistem Deteksi Tingkat Kerusakan Dinding Berbasis YOLOv8n-ORB	Menerapkan varian YOLOv8n-ORB (Oriented Bounding Box) untuk mendeteksi keretakan atau kerusakan pada dinding

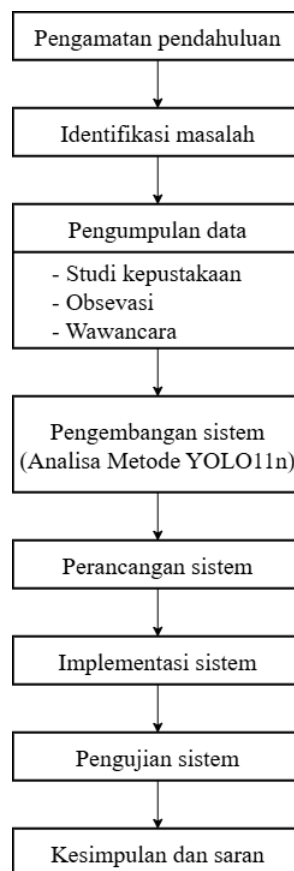
			Untuk Mendukung Pemeliharaan	bangunan. Model dilatih menggunakan 15.553 citra dinding dengan format koordinat delapan titik. Pengujian menunjukkan hasil yang impresif dengan mAP50 sebesar 90,7%, Precision 87,2%, dan Recall 84,9%, serta membuktikan efisiensi varian nano dalam pemrosesan real-time
8.	A. N. Rahmawati	2025	Deteksi Pergerakan Kepala Menggunakan Kombinasi YOLOv8N dan GhostNet untuk Navigasi Kursi Roda Pintar	Menggunakan YOLOv8n sebagai arsitektur dasar yang dimodifikasi dengan GhostNet untuk mendeteksi arah gerakan kepala secara real-time pada perangkat Jetson Nano. Model modifikasi ini sangat ringan (hanya berukuran 3,7 MB) dengan raihan mAP50 sebesar 99,4% dan waktu inferensi ekstra cepat sebesar 53,2 ms menggunakan akselerasi CUDA
9.	Fitriani	2024	Sistem Deteksi Otomatis Penggunaan Senjata Tajam Menggunakan YOLOv8n Deep Learning	Model YOLOv8n berhasil mendeteksi senjata tajam dengan nilai Precision 0,865, Recall 0,815, dan mAP50 sebesar 0,895, serta sangat responsif untuk pengujian real-time
10.	Saputro, Fatichah, & Wihayanti	2026	Deteksi Kecurangan Ujian Berbasis Foto: Perbandingan Kinerja YOLOv5 dan YOLOv8.	Melakukan pengujian komparatif sistem pengawasan ujian daring. Varian YOLOv8n menunjukkan konvergensi pelatihan yang sangat stabil dengan performa metrik kuantitatif (Precision, Recall, dan mAP@50) yang tangguh untuk mengenali berbagai jenis kecurangan

BAB 3

METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Tahapan penelitian merupakan langkah-langkah yang dilakukan secara sistematis untuk mencapai tujuan penelitian. Metodologi penelitian ini disusun agar proses penelitian berjalan terarah dan sesuai dengan tujuan yang telah ditetapkan. Pada penelitian ini terdapat beberapa tahapan yang dilakukan, yaitu pengamatan pendahuluan, identifikasi masalah, pengumpulan data, pengembangan sistem, implementasi sistem, pengujian sistem, serta penarikan kesimpulan dan saran.



Gambar 3. 1 Tahapan Penelitian

3.2 Pengamatan Pendahuluan

Pengamatan pendahuluan merupakan tahap awal yang dilakukan untuk memahami kondisi objek penelitian secara langsung. Pada tahap ini dilakukan pengamatan terhadap proses sortasi buah kelapa sawit di Peron Kamila yang berlokasi di Desa Dayo, Kecamatan Tandun, Kabupaten Rokan Hulu, Provinsi Riau.

Pengamatan dilakukan untuk mengetahui proses penentuan tingkat kematangan buah kelapa sawit yang masih dilakukan secara visual oleh pekerja. Hasil pengamatan menunjukkan bahwa proses identifikasi kematangan buah sangat bergantung pada pengalaman dan ketelitian pekerja sehingga berpotensi menimbulkan perbedaan hasil penilaian.

Selain itu, dilakukan studi awal mengenai penerapan teknologi *Artificial Intelligence*, khususnya *Computer Vision* dan *Object Detection*, dalam bidang pertanian. Hasil pengamatan ini menjadi dasar pemilihan metode *YOLO11n* sebagai metode yang digunakan dalam penelitian.

3.3 Identifikasi Masalah

Identifikasi masalah dilakukan berdasarkan hasil pengamatan pendahuluan yang telah dilakukan. Permasalahan utama yang ditemukan adalah proses identifikasi tingkat kematangan buah kelapa sawit masih dilakukan secara manual sehingga membutuhkan waktu yang relatif lama dan rentan terhadap kesalahan.

Perbedaan persepsi antar pekerja dalam menentukan kategori buah mentah, matang, dan lewat matang dapat menyebabkan ketidakkonsistenan hasil sortasi. Oleh karena itu, diperlukan suatu sistem yang mampu mendeteksi dan

mengklasifikasikan tingkat kematangan buah kelapa sawit secara otomatis dengan tingkat akurasi yang tinggi.

Sebagai solusi dari permasalahan tersebut, penelitian ini mengusulkan pembangunan sistem klasifikasi kematangan buah kelapa sawit menggunakan metode *Object Detection YOLO11n* berbasis web.

3.4 Pengumpulan Data

Peneliti melakukan pengumpulan data pada penelitian ini melalui tiga metode, yaitu studi kepustakaan, observasi, dan wawancara.

3.4.1 Studi Kepustakaan

Studi kepustakaan dilakukan dengan mengumpulkan berbagai referensi yang berkaitan dengan penelitian, seperti jurnal ilmiah, buku, artikel, dan penelitian terdahulu yang membahas *Computer Vision*, *Object Detection*, serta metode *YOLO11n*.

Studi kepustakaan bertujuan untuk memperoleh landasan teori yang kuat sebagai dasar dalam pengembangan sistem klasifikasi kematangan buah kelapa sawit.

3.4.2 Observasi (Pengamatan)

Observasi dilakukan secara langsung di Peron Kamila yang berlokasi di Desa Dayo, Kecamatan Tandun, Kabupaten Rokan Hulu, Provinsi Riau. Data citra buah kelapa sawit diperoleh dari hasil dokumentasi lapangan dan sumber tambahan dari internet yang relevan dengan objek penelitian.

3.4.3 Wawancara

Wawancara dilakukan secara langsung dengan pemilik peron kamila untuk memperoleh informasi mendalam mengenai kendala yang dihadapi dalam proses penyortiran manual, termasuk faktor kelelahan, standar kualitas yang diterapkan, serta kebutuhan terhadap sistem teknologi yang dapat membantu pekerjaan mereka.

3.5 Pengembangan Sistem

Pengembangan sistem pada penelitian ini bertujuan untuk membangun sistem deteksi dan klasifikasi tingkat kematangan buah kelapa sawit menggunakan metode *Object Detection YOLO11n*. Sistem dirancang untuk mendeteksi objek tandan buah kelapa sawit (TBS) dan mengklasifikasikannya ke dalam tiga kategori, yaitu mentah, matang, dan lewat matang. Proses pengembangan sistem dilakukan menggunakan bahasa pemrograman *Python* pada platform Google Colab untuk pelatihan model, sedangkan implementasi sistem berbasis web dikembangkan menggunakan *Visual Studio Code* dengan memanfaatkan dataset yang telah dikumpulkan dan dianotasi sebelumnya.

3.5.1 Input Data

Tahap *input* data merupakan tahap awal dalam pengembangan sistem. Data yang digunakan berupa citra buah kelapa sawit yang diperoleh dari hasil pengambilan gambar secara langsung di Peron Kamila, Desa Dayo, Kecamatan Tandun, Kabupaten Rokan Hulu, Provinsi Riau, serta beberapa citra tambahan yang diperoleh dari sumber internet yang relevan. Dataset terdiri dari tiga kelas, yaitu buah mentah, matang, dan lewat matang. Seluruh data citra disimpan dalam

format JPG atau PNG dan digunakan sebagai masukan utama dalam proses pelatihan model *YOLO11n*.

3.5.2 Anotasi Data

Setelah proses pengumpulan data selesai dilakukan, tahap selanjutnya adalah anotasi data. Proses anotasi dilakukan menggunakan platform *Roboflow* dengan cara memberikan *bounding box* pada setiap objek buah kelapa sawit yang terdapat pada citra. Setiap objek kemudian diberi label sesuai tingkat kematangannya, yaitu mentah, matang, atau lewat matang. Anotasi ini bertujuan untuk memberikan informasi lokasi objek dan kategori kelas yang nantinya digunakan sebagai data pelatihan bagi model *YOLO11n*.

3.5.3 Pembagian Dataset

Dataset yang telah dianotasi kemudian dibagi menjadi tiga bagian, yaitu data *training*, data *validation*, dan data *testing*. Pembagian dataset dilakukan untuk memastikan model dapat dilatih, divalidasi, dan diuji secara optimal. Pada penelitian ini digunakan pembagian data sebesar 80% untuk *training*, 10% untuk *validation*, dan 10% untuk *testing*. Data *training* digunakan untuk melatih model, data *validation* digunakan untuk memantau performa model selama proses pelatihan, sedangkan data *testing* digunakan untuk mengevaluasi kinerja model setelah proses pelatihan selesai dilakukan.

3.5.4 Pelatihan Model *YOLO11n*

Pada tahap ini, dataset yang telah dianotasi dan dibagi ke dalam beberapa subset digunakan untuk melatih model *YOLO11n*. Model *YOLO11n* dipilih karena memiliki kemampuan deteksi objek yang baik dengan tetap mempertahankan

efisiensi komputasi, sehingga sesuai untuk implementasi pada sistem deteksi tingkat kematangan buah kelapa sawit berbasis web. Selama proses pelatihan, model mempelajari karakteristik visual tandan buah kelapa sawit, seperti warna, bentuk, tekstur, dan pola yang membedakan tingkat kematangan buah. Proses pelatihan dilakukan secara iteratif menggunakan dataset latih hingga diperoleh model dengan performa terbaik berdasarkan hasil evaluasi, yang selanjutnya digunakan pada tahap implementasi sistem.

3.5.5 Deteksi dan Klasifikasi Objek

Setelah model selesai dilatih, sistem akan melakukan proses deteksi dan klasifikasi objek pada citra yang diberikan pengguna. Model *YOLO11n* akan mengidentifikasi lokasi objek buah kelapa sawit menggunakan *bounding box* sekaligus menentukan kelas objek yang terdeteksi. Proses ini dilakukan secara otomatis sehingga sistem dapat mengenali apakah buah kelapa sawit termasuk kategori mentah, matang, atau lewat matang.

3.5.6 Output Sistem

Tahap akhir dari pengembangan sistem adalah menghasilkan output berupa hasil deteksi objek pada citra. Sistem akan menampilkan *bounding box* pada objek yang berhasil dideteksi beserta label kelas yang menunjukkan tingkat kematangan buah kelapa sawit. Selain itu, sistem juga menampilkan nilai *confidence score* yang menunjukkan tingkat keyakinan model terhadap hasil prediksi yang diberikan. Output tersebut diharapkan dapat membantu pengguna dalam melakukan identifikasi tingkat kematangan buah kelapa sawit secara lebih cepat dan objektif.

3.6 Kesimpulan dan Saran

Tahapan terakhir adalah penarikan kesimpulan berdasarkan hasil yang diperoleh dari implementasi metode *YOLO11n* dalam mendeteksi dan mengklasifikasikan tingkat kematangan tandan buah kelapa sawit. Kesimpulan yang dihasilkan mencakup performa model dalam mendeteksi objek serta mengklasifikasikannya ke dalam tiga kategori, yaitu mentah, matang, dan lewat matang.

Selain itu pada bagian ini juga disampaikan saran sebagai acuan untuk pengembangan penelitian selanjutnya. Pengembangan dapat dilakukan dengan memperhatikan kualitas dataset yang digunakan, melakukan perbandingan performa *YOLO11n* dengan model deteksi objek lainnya, serta mengevaluasi pengaruh variasi dataset terhadap performa model sehingga diperoleh hasil deteksi yang lebih optimal.