

Rancang Bangun Sistem Deteksi Buah Jeruk Menggunakan Yolov8

**Gilang Ramadhani¹, Regi Cendika Pratama², Wildan Ramadhani Yahya³,
Resty Wulanningrum⁴**

^{1,2}Teknik Informatika, Fakultas Teknik, Universitas Nusantara PGRI Kediri

E-mail: ^{*}gilangramadhani19.22@gmail.com, ²regipratamaofc@gmail.com,

³tugaskuliah803@gmail.com, ⁴restyw@unpkdr.ac.id

Abstrak – Produksi jeruk di Indonesia yang mencapai 2,4 juta ton pada tahun 2021 menunjukkan pentingnya pengelolaan hasil panen yang efisien. Namun, proses penghitungan jumlah jeruk secara manual memakan waktu dan tenaga yang besar, sehingga kurang efektif untuk memenuhi permintaan pasar yang terus meningkat. Penelitian ini menawarkan solusi berupa sistem otomatis berbasis algoritma YOLOv8 untuk mendeteksi dan menghitung jumlah jeruk secara real-time. Algoritma ini dipilih karena kecepatan dan akurasinya yang tinggi, mencapai tingkat akurasi hingga 95% dalam berbagai kondisi pencahayaan dan latar belakang. Hasil pengujian menunjukkan bahwa sistem ini mampu meningkatkan efisiensi proses identifikasi dan penghitungan jeruk dibandingkan metode manual. Implementasi teknologi ini diharapkan dapat membantu petani dalam perencanaan panen, pengelolaan produksi, dan memperbaiki distribusi jeruk di pasar.

Kata Kunci — Jeruk, Deteksi, YOLOv8, Roboflow

1. PENDAHULUAN

Salah satu sub bidang teknologi pengolahan citra dan pembelajaran mesin yang berkembang pesat adalah deteksi objek[1]. Deteksi objek memiliki aplikasi tertentu yang menarik dalam dunia pertanian, khususnya dalam bidang pengenalan dan klasifikasi buah-buahan[2]. Salah satu jenis buah yang paling banyak dikonsumsi di Indonesia adalah jeruk, yang termasuk dalam genus Citrus spp. Berdasarkan BPS, pada tahun 2021, produksi jeruk di Indonesia hampir mencapai angka 2,4 juta ton[3]. Oleh karena itu, jeruk adalah salah satu buah tropis yang paling banyak dibudidayakan dan dikonsumsi. Melihat popularitas buah ini, deteksi dan pengklasifikasian jeruk yang otomatis semakin relevan.

Klasifikasi jeruk yang otomatis membantu dalam menghitung jumlah hasil panen mereka setiap musim [4][5][6]. Pengelolaan produksi ini sangat penting untuk memprediksi jumlah buah yang akan dihasilkan, sehingga dapat mendukung perencanaan dan peningkatan kualitas panen. Saat ini, proses ini masih dilakukan secara manual, yang memakan waktu dan tenaga cukup banyak. Dengan penerapan teknologi, proses penghitungan jumlah buah dapat dipercepat dan diotomatisasi, sehingga membantu petani dalam mengelola produksi dengan lebih efisien serta memenuhi permintaan pasar.

Permintaan pasar yang terus meningkat mendorong penerapan teknologi ini, yang didukung oleh algoritma YOLOv8, yang merupakan kegiatan inovatif yang menggunakan *deep learning* algoritma untuk mendeteksi dan mengklasifikasikan buah jeruk berdasarkan parameter-parameter tertentu, seperti ukuran, warna, dan bentuknya[7]. YOLOv8 atau *You Only Look Once* merupakan salah satu algoritma deteksi objek paling populer karena berbagai keuntungannya, yang terdiri dari kecepatan dan akurasi yang tinggi[8][9][10][11]. Menggunakan model bergenerasi terakhir dari YOLO, yaitu YOLOv8, kita dapat mencapai tingkat deteksi yang lebih tinggi dari model-model yang lebih lama. YOLOv8 mampu mendeteksi dengan akurasi mencapai 95% dibawah latar belakang tersendiri, kondisi iklim dan pencahayaan [12].

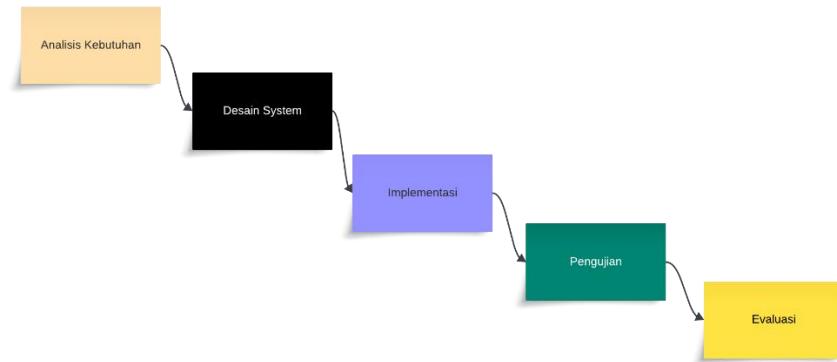
Pada penelitian terdahulu yang dilakukan oleh Syadina A. Prasetya dkk [13], metode YOLOv8 digunakan untuk mengklasifikasikan kematangan nanas dengan melatih model menggunakan dataset gambar nanas yang telah dianotasi dengan empat tingkat kematangan sebagai berikut: Mentah. Setengah Matang. Matang. Matang (Proses Data Mining melalui beberapa tahapan: Pengumpulan, Prapemrosesan [Data], Pemodelan, dan Pengujian setelah model dilatih, pengujian dilakukan dengan menggunakan berbagai media: foto, video, dan umpan kamera langsung (*webcam*). Hasil percobaan menunjukkan bahwa algoritma YOLOv8 mampu mendeteksi dan mengklasifikasikan tingkat kematangan nanas secara otomatis dengan hasil yang memuaskan dari segi akurasi, dan dapat berjalan secara *real time*.

Tujuan penelitian ini adalah untuk merancang dan mengembangkan sistem yang mampu mendeteksi dan menghitung jumlah buah jeruk secara otomatis dengan memanfaatkan algoritma YOLOv8. Dengan adanya sistem

otomatis ini, proses identifikasi jeruk dapat dilakukan dengan lebih cepat, akurat, dan konsisten dibandingkan dengan metode manual yang membutuhkan waktu serta tenaga lebih besar. Dengan mendukung efisiensi dalam pengelolaan hasil panen, sistem ini mampu memberikan manfaat langsung kepada petani dalam mengoptimalkan kualitas produksi serta memperbaiki rantai pasokan buah jeruk di pasar.

2. METODE PENELITIAN

Pada penelitian ini, metode yang digunakan adalah Metode *Waterfall*. Metode *Waterfall* merupakan salah satu model *SDLC* (*System Development Life Cycle*) yang sering digunakan dalam pengembangan sistem informasi atau perangkat lunak. Model ini menggunakan pendekatan sistematis dan berurutan[14]. Tahapan dari Metode *Waterfall* dapat dilihat pada gambar di bawah ini:



Gambar 1 Metode *Waterfall*

2.1 Analisis Kebutuhan

Tahap analisis kebutuhan merupakan langkah awal dalam metode *SDLC*, metode *waterfall* yang bertujuan untuk mengidentifikasi kebutuhan sistem. Dalam pengembangan sistem ini, data diambil dari *Roboflow*, sebuah platform yang menyediakan dataset dan alat untuk membangun model machine learning berbasis *computer vision*[15]. Data yang diambil meliputi gambar jeruk beserta anotasinya, yang dianalisis untuk memahami variasi penting seperti pencahayaan, sudut pengambilan gambar, dan kualitas visual. Analisis ini memastikan bahwa dataset yang digunakan telah siap untuk mendukung proses desain sistem, pelatihan, dan pengujian model *YOLOv8* pada tahap berikutnya.

2.2 Desain Sistem

Tahap desain mencakup perencanaan sistem deteksi buah jeruk menggunakan *YOLOv8*. Proses ini melibatkan pemilihan arsitektur model yang sesuai, pengaturan parameter pelatihan, serta perencanaan implementasi *YOLOv8* untuk mendeteksi buah jeruk dalam gambar dari dataset *Roboflow*. Selain itu, tahap ini mencakup pemilihan teknik augmentasi data yang bertujuan untuk meningkatkan kualitas dan keragaman dataset, sehingga mendukung kinerja model dalam berbagai kondisi pengujian. Output dari sistem ini dirancang dalam bentuk antarmuka berbasis *website*, yang akan dibangun menggunakan *framework Streamlit*. Website ini memungkinkan pengguna untuk mengunggah gambar secara langsung dan mendapatkan hasil deteksi buah jeruk dengan cepat, memudahkan pengujian dan penerapan sistem deteksi objek secara interaktif.

2.3 Implementasi

Implementasi dilakukan dengan melatih model *YOLOv8* menggunakan dataset jeruk yang diperoleh dari *Roboflow*. Pada tahap ini, model dikembangkan dengan memanfaatkan gambar yang telah dianotasi untuk mendeteksi buah jeruk dalam berbagai kondisi dan tampilan. Proses pelatihan dilakukan dalam beberapa epoch, di mana model mempelajari pola dari dataset melalui siklus iterasi penuh.

2.4 Pengujian

Tujuan dari tahap uji ini adalah untuk menguji kemampuan model *YOLOv8* dalam mendeteksi buah jeruk melalui antarmuka *website* berbasis *Streamlit*. Proses pengujian dilakukan dengan mengunggah gambar jeruk sebagai input ke *website*, yang kemudian diproses oleh model untuk mendeteksi dan menampilkan hasil berupa *bounding box* dan label "orange". Pengujian ini bertujuan untuk memastikan bahwa sistem deteksi dapat berjalan dengan baik dan memberikan hasil sesuai dengan desain yang direncanakan.

2.5 Evaluasi

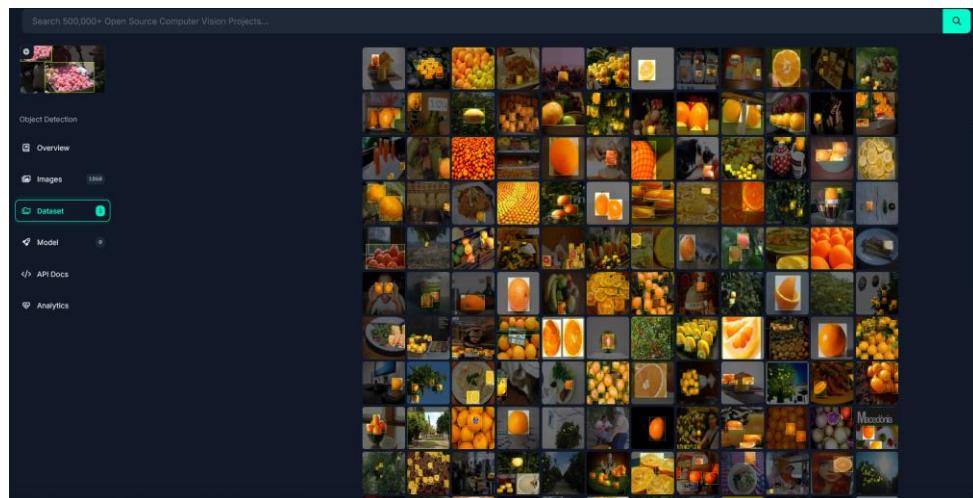
Tujuan dari tahap uji ini adalah untuk mengevaluasi kemampuan model *YOLOv8* dalam mendeteksi jeruk dengan akurasi tinggi. Pengujian melibatkan metrik seperti akurasi, presisi, recall, dan mean average precision (mAP). Data uji yang digunakan berisi gambar jeruk yang dianotasi dengan *bounding box*, yang akan menguji sejauh mana model mendeteksi jeruk secara tepat, menghindari kesalahan deteksi, dan menangkap semua objek jeruk yang ada. Hasil evaluasi ini membantu mengidentifikasi area perbaikan dan memberikan umpan balik untuk meningkatkan performa model dalam kondisi nyata, seperti variasi pencahayaan dan objek latar belakang.

3. HASIL DAN PEMBAHASAN

Bab ini memaparkan hasil dari setiap tahap pengembangan sistem deteksi jeruk menggunakan *YOLOv8*. Dimulai dari analisis kebutuhan hingga evaluasi, hasil dan pembahasan disusun untuk memberikan gambaran menyeluruh tentang proses pengembangan sistem. Fokus utama pembahasan mencakup performa model, tantangan yang dihadapi selama proses, serta solusi dan perbaikan yang dilakukan untuk mencapai hasil yang optimal.

3.1 Analisis Kebutuhan

Tahap analisis kebutuhan dilakukan untuk mengidentifikasi elemen-elemen utama yang diperlukan dalam pengembangan sistem deteksi jeruk berbasis *YOLOv8*. Dataset yang digunakan diperoleh dari platform *Roboflow*, yang menyediakan kumpulan gambar dengan anotasi *bounding box* untuk objek jeruk. Dataset ini disiapkan untuk mendukung pelatihan, validasi, dan pengujian model. Berikut adalah gambar dataset yang akan digunakan:

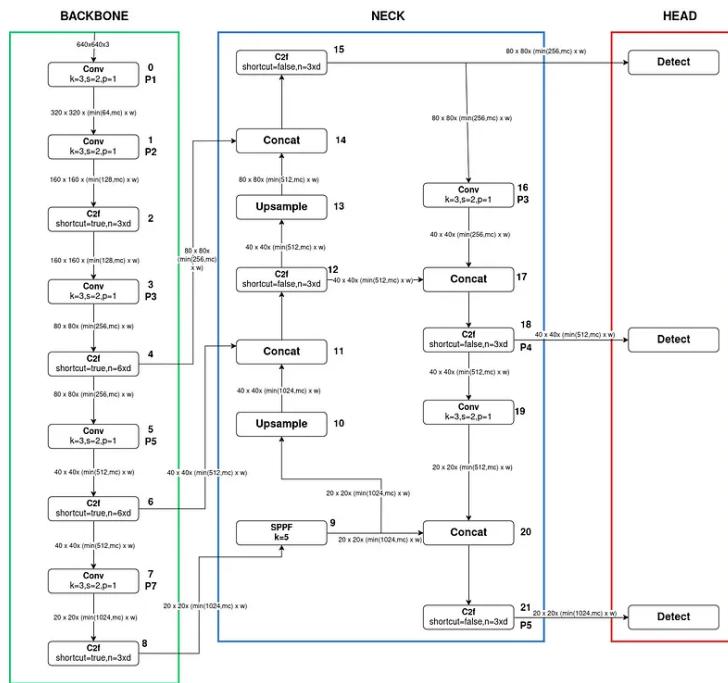


Gambar 2 Data Gambar dari Roboflow

Analisis menunjukkan bahwa dataset memiliki variasi yang memadai untuk melatih model, meskipun ditemukan beberapa gambar dengan kondisi pencahayaan yang kurang optimal. Kondisi ini dipertimbangkan sebagai tantangan yang akan diatasi pada tahap desain.

3.2 Desain Sistem

Pada tahap ini, sistem deteksi buah jeruk dirancang untuk mendukung proses identifikasi objek secara efisien menggunakan *YOLOv8*. Desain sistem meliputi desain arsitektur model *YOLOv8*, Sistem deteksi buah jeruk menggunakan *YOLOv8* dirancang tanpa database untuk memastikan efisiensi dan kesederhanaan. Model *YOLOv8* memanfaatkan *Backbone CSPDarknet* untuk ekstraksi fitur, *Neck PAN* untuk penggabungan fitur multi-skala, dan *Head* untuk menghasilkan prediksi *bounding box*, *confidence score*, serta label objek. Parameter seperti resolusi *input* gambar, *learning rate*, dan *batch size* diatur optimal menggunakan dataset jeruk dari *Roboflow*. Pengguna dapat mengunggah gambar melalui *website* berbasis *Streamlit*, yang kemudian diproses langsung oleh model *YOLOv8*. Hasil deteksi berupa *bounding box* dan label "orange" ditampilkan kembali kepada pengguna tanpa penyimpanan data permanen, arsitektur *YOLOv8* dapat dilihat pada Gambar berikut :



Gambar 3 Gambar Arsitektur YOLOv8

(Sumber gambar : <https://abintimilsina.medium.com/yolov8-architecture-explained-a5e90a560ce5>)

3.3 Implementasi

Pada tahap implementasi, model *YOLOv8* dilatih menggunakan dataset jeruk yang diperoleh dari *Roboflow*. Proses pelatihan dapat dilihat di gambar di bawah ini

```

100 epochs completed in 2.663 hours.
Optimizer stripped from runs/detect/epoch100/weights/last.pt, 5.6MB
Optimizer stripped from runs/detect/epoch100/weights/best.pt, 5.6MB

Validating runs/detect/epoch100/weights/best.pt...
Ultralytics 8.3.8 🚀 Python-3.12.3 torch-2.4.1 CPU (Apple M3 Pro)
Model summary (fused): 186 layers, 2,684,563 parameters, 0 gradients, 6.8 GFLOPS
  Class   Images  Instances   Box(P   R   mAP50   mAP50-95): 100% |██████████| 6/6 [00:09<00:00,  1.63s/it]
    all     180      980     0.601     0.61     0.587     0.358
Speed: 0.3ms preprocess, 51.2ms inference, 0.0ms loss, 0.2ms postprocess per image
Results saved to runs/detect/epoch100

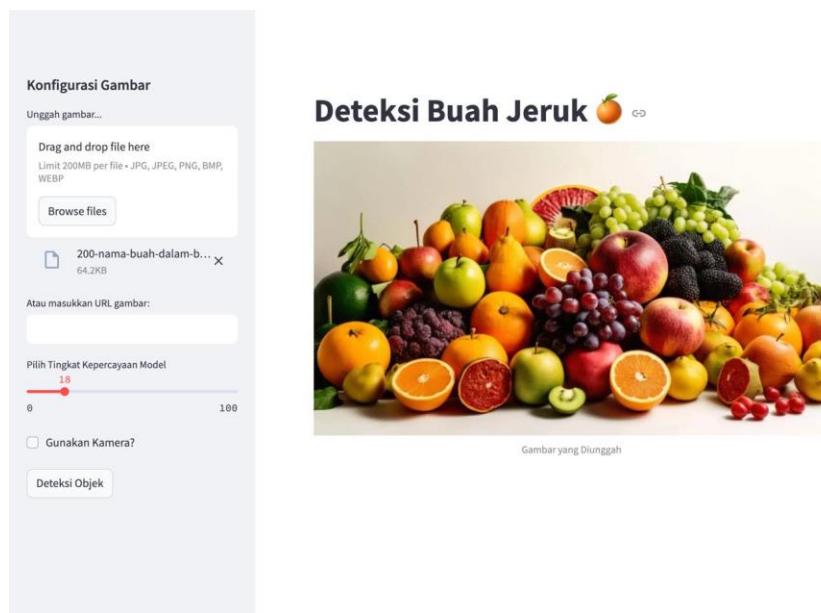
```

Gambar 4 Proses Epoch

Dengan performa tersebut, model *YOLOv8* menunjukkan kemampuan yang baik dalam mendeteksi buah jeruk, meskipun masih ada peluang untuk peningkatan lebih lanjut, terutama pada nilai mAP50-95 yang menunjukkan potensi untuk deteksi dengan tingkat presisi yang lebih tinggi. Meskipun demikian, model ini sudah memberikan hasil yang cukup memadai dalam mendeteksi objek jeruk secara akurat. Implementasi ini juga siap untuk diuji pada dataset baru atau diterapkan dalam aplikasi berbasis deteksi objek, dengan kemungkinan pengoptimalan lebih lanjut untuk meningkatkan akurasi, efisiensi, dan keandalan deteksi di berbagai kondisi.

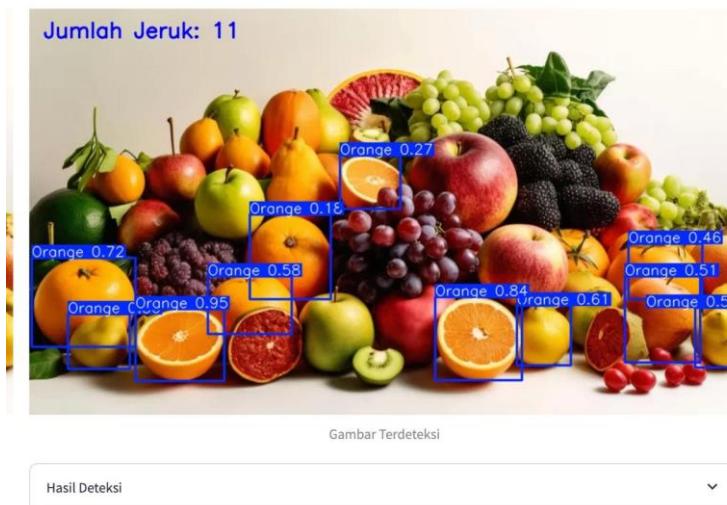
3.4 Pengujian

Bagian ini menjelaskan hasil pengujian sistem deteksi buah jeruk yang dikembangkan menggunakan model *YOLOv8* dan diimplementasikan melalui antarmuka *website* berbasis *Streamlit*. Pengujian dilakukan dengan menggunakan input berupa gambar buah jeruk yang diunggah oleh pengguna ke dalam sistem. Proses pengujian ini bertujuan untuk memastikan bahwa model dapat mendeteksi buah jeruk secara akurat, menampilkan hasil berupa *bounding box* dan label pada gambar yang diuji.



Gambar 5 Upload Gambar jeruk

Gambar di atas menunjukkan berbagai jenis buah, termasuk jeruk, yang diujikan untuk sistem deteksi. Model *YOLOv8* digunakan untuk mengenali dan membedakan buah jeruk dari buah lainnya yang ada pada gambar tersebut. Pengujian ini penting untuk menilai kinerja model dalam menghadapi kompleksitas gambar dengan berbagai objek yang berbeda, serta mengukur akurasi dan efektivitas deteksi yang dilakukan.

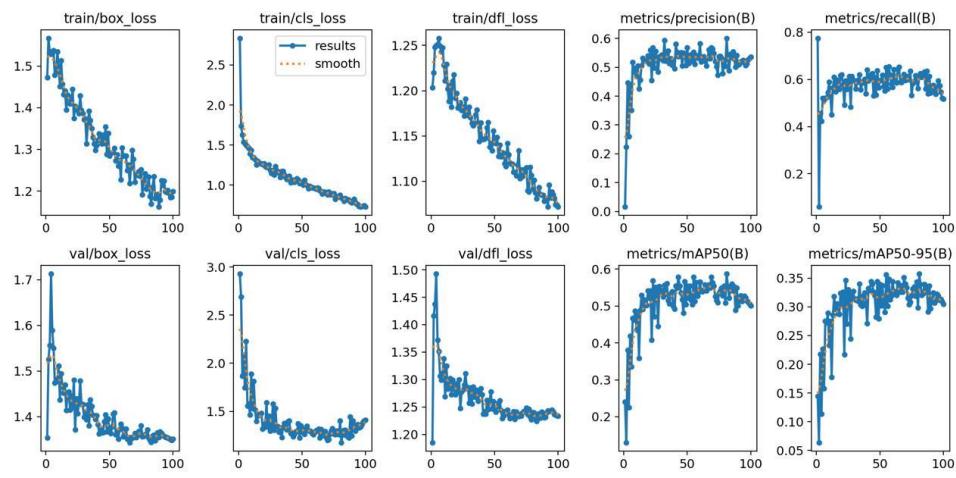


Gambar 6 Gambar Jeruk yang terdeteksi dan jumlah jeruk

Gambar di atas menunjukkan hasil deteksi jeruk pada gambar yang berisi berbagai jenis buah. Model berhasil mengidentifikasi 11 jeruk, dengan setiap jeruk diberikan kotak pembatas dan skor kepercayaan (*confidence score*). Hasil deteksi ini menunjukkan seberapa baik model dapat mengenali jeruk meskipun ada berbagai objek lain di dalam gambar. Skor kepercayaan yang ditampilkan menunjukkan tingkat keyakinan model dalam mendekteksi jeruk, yang bervariasi antara 0,1 hingga 0,95. Hasil ini mengindikasikan bahwa model memiliki kemampuan yang cukup baik dalam mendekteksi jeruk meskipun kondisi gambar yang kompleks.

3.5 Evaluasi

Setelah proses pelatihan model selesai, dilakukan evaluasi untuk mengukur kinerja model *YOLOv8* pada dataset uji. Evaluasi ini mencakup pengujian beberapa metrik penting, yaitu train loss, validation loss, precision, recall, dan mean average precision (mAP). Grafik berikut menunjukkan hasil metrik pelatihan dan pengujian model *YOLOv8*



Gambar 7 Hasil Matrik Evaluasi

Dari Gambar di atas, dapat dianalisis sebagai berikut:

1. Train Loss: Terdapat penurunan yang signifikan pada metrik *train/box_loss*, *train/cls_loss*, dan *train/dfl_loss* sepanjang 100 epoch, menunjukkan bahwa model berhasil mengurangi kesalahan selama pelatihan.
2. Validation Loss: Grafik *val/box_loss*, *val/cls_loss*, dan *val/dfl_loss* menunjukkan tren yang serupa, di mana model semakin mampu menggeneralisasi pada data uji setelah pelatihan yang cukup.
3. Precision: Nilai precision model menunjukkan peningkatan yang konsisten, menandakan bahwa sebagian besar prediksi model untuk jeruk adalah benar.
4. Recall: Recall model juga menunjukkan hasil yang stabil, menggambarkan kemampuan model untuk mendeteksi sebanyak mungkin objek jeruk yang ada pada data uji.
5. mAP (mean average precision): Hasil untuk mAP50 mencapai sekitar 0.5, yang mengindikasikan bahwa model memiliki akurasi yang baik dalam mendeteksi jeruk pada *threshold* IoU 0.5. Sedangkan mAP50-95 berada pada 0.25, memberikan gambaran akurasi model dengan berbagai variasi *threshold*.

Dengan demikian, hasil dari grafik ini menunjukkan bahwa model *YOLOv8* berhasil melaksanakan tugas deteksi objek dengan tingkat akurasi yang tinggi, meskipun terdapat fluktuasi pada beberapa metrik di awal pelatihan. Evaluasi lebih lanjut bisa dilakukan dengan memperbaiki parameter dan meningkatkan kualitas dataset jika diperlukan.

4. SIMPULAN

Penelitian ini berhasil merancang dan mengembangkan sistem deteksi jeruk otomatis menggunakan algoritma *YOLOv8* yang diimplementasikan melalui antarmuka berbasis *Streamlit*. Hasil evaluasi menunjukkan bahwa model mampu mendeteksi jeruk dengan akurasi yang baik, dengan nilai mAP50 sekitar 0,5, yang menunjukkan performa yang memadai dalam deteksi objek pada *threshold* IoU 0,5. Sistem ini memungkinkan deteksi jeruk dilakukan secara otomatis dan efisien, mendukung pengelolaan hasil panen yang lebih cepat dan konsisten. Namun, terdapat beberapa tantangan, seperti fluktuasi nilai mAP50-95 dan sensitivitas terhadap kondisi pencahayaan tertentu, yang membuka peluang untuk pengembangan lebih lanjut.

5. SARAN

Untuk penelitian selanjutnya, disarankan untuk memperkaya dataset dengan variasi gambar yang lebih beragam, termasuk gambar dengan latar belakang yang lebih kompleks, kondisi pencahayaan yang berbeda, serta sudut pandang yang lebih bervariasi. Hal ini akan membantu model *YOLOv8* untuk mengatasi tantangan dunia nyata yang sering kali melibatkan pengambilan gambar dengan kondisi lingkungan yang dinamis. Selain itu, pengoptimalan parameter model seperti *learning rate*, jumlah epoch, dan teknik regularisasi lainnya sangat penting untuk meningkatkan akurasi deteksi. Penyesuaian ini dapat membantu mengurangi *overfitting* dan meningkatkan generalisasi model pada data yang tidak terlihat sebelumnya.

Pengembangan lebih lanjut dari sistem deteksi ini dapat mencakup penerapan model pada perangkat berbasis *Internet of Things (IoT)*, seperti drone atau robot pemanen, untuk otomatisasi proses penghitungan dan deteksi buah jeruk dalam skala besar. Dengan penerapan teknologi ini, petani dapat mengoptimalkan proses pemanenan dan manajemen hasil panen secara lebih efisien, mengurangi ketergantungan pada tenaga manusia, serta menghemat waktu dan biaya operasional. Selain itu, penerapan model pada sistem berbasis *IoT* juga membuka peluang untuk integrasi dengan teknologi lain, seperti sistem pemantauan cuaca dan prediksi hasil panen, yang dapat membantu meningkatkan keberlanjutan dan produktivitas dalam industri pertanian.

DAFTAR PUSTAKA

- [1] S. Y. Z. Mutianniza, “Aplikasi Kamera Cerdas Untuk Deteksi Kendaraan Menggunakan Library Tensorflow,” *Jurnal Sintaks Logika*, vol. 3, no. 3, pp. 61–68, 2023.
- [2] L. Lusiana, A. Wibowo, and T. K. Dewi, “Implementasi Algoritma Deep Learning You Only Look Once (YOLOv5) Untuk Deteksi Buah Segar Dan Busuk,” *Paspalum: Jurnal Ilmiah Pertanian*, vol. 11, no. 1, pp. 123–130, 2023.
- [3] Badan Pusat Statistik, “Produksi Tanaman Buah-buahan, 2021-2023,” <https://www.bps.go.id/id/statistics-table/2/NjIjMg==/produksi-tanaman-buah-buahan.html>.
- [4] E. Amalia, A. N. Izzah, T. S. Billa, W. Hamidah, A. B. Kaswar, and M. R. Edy, “Deteksi Jumlah Jeruk Menggunakan Metode Tranformasi Hough,” *Journal of Embedded Systems, Security and Intelligent Systems*, pp. 24–31, 2023.
- [5] E. Amalia, A. N. Izzah, T. S. Billa, W. Hamidah, A. B. Kaswar, and M. R. Edy, “Deteksi Jumlah Jeruk Menggunakan Metode Tranformasi Hough,” *Journal of Embedded Systems, Security and Intelligent Systems*, pp. 24–31, 2023.
- [6] Y. F. N. Ashfani, Y. Litanianda, and R. A. Putri, “Klasifikasi Jenis Buah Jeruk Menggunakan Metode Convolutional Neural Network: Deep Learning Studi,” *Uranus: Jurnal Ilmiah Teknik Elektro, Sains dan Informatika*, vol. 2, no. 2, pp. 70–79, 2024.
- [7] A. Setiyadi, E. Utami, and D. Ariatmanto, “Analisa Kemampuan Algoritma YOLOv8 Dalam Deteksi Objek Manusia Dengan Metode Modifikasi Arsitektur,” *J-SAKTI (Jurnal Sains Komputer dan Informatika)*, vol. 7, no. 2, pp. 891–901, 2023.
- [8] A. A. Permana, M. R. Muttaqin, and M. A. Sunandar, “SISTEM DETEKSI API SECARA REAL TIME MENGGUNAKAN ALGORITMA YOU ONLY LOOK ONCE (YOLO) VERSI 8,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 5, pp. 10395–10400, 2024.
- [9] R. T. Hutabarat and R. Kurniawan, “Deteksi Sampah di Permukaan Sungai menggunakan Convolutional Neural Network dengan Algoritma YOLOv8,” in *Seminar Nasional Official Statistics*, 2024, pp. 537–548.
- [10] D. S. Ariansyah, “PENDETEKSI KATA DALAM BAHASA ISYARAT MENGGUNAKAN ALGORITMA YOLO VERSI 8,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3, 2024.

- [11] D. N. Alfarizi, R. A. Pangestu, D. Aditya, M. A. Setiawan, and P. Rosyani, “Penggunaan Metode YOLO Pada Deteksi Objek: Sebuah Tinjauan Literatur Sistematis,” *AI dan SPK: Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan*, vol. 1, no. 1, pp. 54–63, 2023.
- [12] J. Thompson, R. Benac, K. Olana, T. Hassan, A. Sward, and T. Khan Mohd, “Nftrig: Using blockchain technologies for math education,” in *International Conference on Intelligent Sustainable Systems*, Springer, 2023, pp. 609–624.
- [13] S. A. Prasetya, M. Mihuandayani, Y. Abast, M. Mangole, and J. Rahman, “Implementasi Convolutional Neural Network untuk Klasifikasi Tingkat Kematangan Buah Nanas Menggunakan YOLOv8,” *Building of Informatics, Technology and Science (BITS)*, vol. 6, no. 1, pp. 567–575, 2024.
- [14] A. A. Wahid, “Analisis metode waterfall untuk pengembangan sistem informasi,” *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, vol. 1, no. 1, pp. 1–5, 2020.
- [15] N. J. Hayati, D. Singasatia, and M. R. Muttaqin, “Object Tracking Menggunakan Algoritma You Only Look Once (Yolo) V8 Untuk Menghitung Kendaraan,” *Komputa: Jurnal Ilmiah Komputer dan Informatika*, vol. 12, no. 2, pp. 91–99, 2023.