

PENERAPAN CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI JENIS PENYAKIT PADA BUAH JAMBU BIJI

Ishab Mohtarrudin, Barry Ceasar Octariadi, Alda Cendekia Siregar

Teknik Informatika, Universitas Muhammadiyah Pontianak

Jl. A Yani No.111, Pontianak, Indonesia 78124

Ishabkun01@gmail.com

ABSTRAK

Jambu biji merupakan komoditas hortikultura penting yang produksinya terancam oleh penyakit lalat buah dan antraknosa. Identifikasi manual yang subjektif dan memerlukan keahlian khusus menjadi hambatan utama bagi petani. Penelitian ini bertujuan untuk mengimplementasikan dan menguji kinerja *Convolutional Neural Network* (CNN) dengan arsitektur DenseNet121 dalam mengklasifikasikan citra buah jambu biji ke dalam tiga kelas: sehat, lalat buah, dan antraknosa. Sebanyak 1.500 citra primer diproses menggunakan teknik Canny Edge Detection untuk ekstraksi ciri struktural dan augmentasi data. Hasil evaluasi pada 150 data uji menunjukkan akurasi model mencapai 91,33%, dengan performa sempurna (100%) pada klasifikasi jambu biji sehat. Penelitian ini membuktikan bahwa integrasi *DenseNet121* dan *Canny Edge Detection* dapat menjadi solusi otomatis yang efektif dan akurat bagi petani dalam mendeteksi penyakit buah.

Kata kunci : *Canny Edge Detection, Convolutional Neural Network (CNN), DenseNet121, Jambu Biji, Klasifikasi Penyakit.*

1. PENDAHULUAN

Jambu biji merupakan salah satu komoditas hortikultura bernilai ekonomi tinggi dan menjadi unggulan di berbagai daerah di Indonesia, termasuk Provinsi Kalimantan Barat. Berdasarkan data Dinas Tanaman Pangan dan Hortikultura Provinsi Kalimantan Barat, produksi jambu biji di wilayah ini mengalami fluktuasi signifikan, dengan kenaikan pesat dari 2.438 ton pada tahun 2019 menjadi 7.221 ton pada tahun 2023 [1]. Peningkatan produksi ini menuntut pengelolaan pascapanen yang lebih baik, terutama dalam menjaga kualitas buah dari serangan penyakit yang dapat menurunkan nilai jual di pasar.

Namun, tantangan besar yang dihadapi petani adalah ancaman penyakit seperti lalat buah dan antraknosa yang sering menyerang secara masif. Hingga saat ini, identifikasi penyakit pada data skala besar masih dilakukan secara manual, yang mana proses tersebut sangat sulit karena pola gejala antar penyakit seringkali terlihat mirip. Selain itu, pemantauan manual sangat melelahkan bagi pengamat dan bersifat subjektif, sehingga rentan terhadap kesalahan diagnosis yang berakibat pada ketidaktepatan penanganan [2].

Untuk mengatasi keterbatasan tersebut, penelitian ini bertujuan untuk mengimplementasikan sistem *Deep Learning* berbasis *Convolutional Neural Network* (CNN). Sistem ini diharapkan mampu mengekstraksi fitur visual penyakit pada citra jambu biji secara otomatis, objektif, dan konsisten guna menjamin akurasi diagnosis yang lebih tinggi dibandingkan metode konvensional [2]. Penggunaan teknologi ini bertujuan untuk memberikan standar baru dalam klasifikasi penyakit buah yang dapat membantu petani meminimalisir kerugian hasil panen.

Rencana penyelesaian yang diangkat dalam penelitian ini adalah penerapan CNN dengan arsitektur

spesifik yang dirancang secara fundamental untuk memproses dan menganalisis data visual [3]. CNN dipilih karena kemampuannya dalam melakukan ekstraksi fitur kompleks seperti bentuk spesifik bercak, tekstur, dan perubahan warna secara otomatis tanpa campur tangan manusia [2].

Secara teknis, model ini akan mempelajari pola gambar secara bertahap, dimulai dari pengenalan ciri sederhana seperti garis atau tepi hingga memahami representasi fitur kompleks seperti pola bercak penyakit secara utuh [4]. Untuk meningkatkan efisiensi dan keakuratan, penelitian ini akan memanfaatkan teknik *weight sharing* yang mengurangi jumlah parameter latih secara drastis, sehingga model menjadi lebih kokoh (*robust*) terhadap variasi posisi objek dalam citra dan mampu memberikan hasil klasifikasi yang stabil [5].

2. TINJAUAN PUSTAKA

Penelitian yang dilakukan oleh Torangto Yos P. Situngkir pada tahun 2022 berjudul "*Klasifikasi Penyakit Daun Kentang Menggunakan Convolutional Neural Network (CNN)*". Tahapan penelitian meliputi *preprocessing* citra berupa penghapusan *background* serta pengujian terhadap 18 jenis model dengan variasi *hyperparameter* (*epoch* dan *convolutional layer*) serta *optimizer* (Adam, SGD, dan RMSProp) untuk mendapatkan kombinasi optimal. Hasil penelitian menunjukkan bahwa model terbaik mampu mencapai akurasi pengujian sebesar 98,2% dengan kombinasi *optimizer* Adam, 5 *convolutional layer*, dan 100 *epoch* dalam klasifikasi penyakit *early blight* dan *late blight* pada daun kentang [4].

Penelitian oleh Itin Riati, Yuhandri, dan Gunadi W. Nurcahyo pada tahun 2024 berjudul "*Identifikasi Penyakit dan Hama pada Bibit Kelapa Sawit Menggunakan Convolutional Neural Network*

(CNN)". Penelitian ini menerapkan algoritma *Convolutional Neural Network (CNN)* yang merupakan bagian dari *Deep Learning* untuk mengidentifikasi penyakit dan hama pada bibit tanaman kelapa sawit. Model yang digunakan terdiri dari empat *hidden layer* dengan *optimizer Adam*. Hasil penelitian menunjukkan bahwa metode ini mampu mencapai akurasi sebesar 91,66% dalam mengidentifikasi penyakit pada bibit kelapa sawit [2].

Penelitian yang dilakukan oleh Andrian Herbert Parsaoran Sitohang, Teguh Iman Hermanto, dan Candra Dewi Lestari pada tahun 2024 berjudul "*Klasifikasi Penyakit Daun Stroberi Menggunakan Convolutional Neural Network (CNN) dengan Arsitektur InceptionV3*". Hasil penelitian menunjukkan bahwa model yang dikembangkan mampu mencapai akurasi pengujian sebesar 99% dalam mengklasifikasikan penyakit daun stroberi, termasuk *leaf scorch* [3].

Penelitian oleh Ferry Putrawansyah dan Tri Susanti pada tahun 2024 berjudul "*Klasifikasi Jenis Jambu Biji Menggunakan Support Vector Machine (SVM) Berbasis Image Processing*". Penelitian ini menerapkan metode *Support Vector Machine (SVM)* yang dikombinasikan dengan teknik *image processing* menggunakan perangkat lunak MATLAB. Hasil pengujian menunjukkan bahwa model SVM mampu mencapai akurasi sebesar 80% dalam klasifikasi jenis jambu biji [6].

Penelitian yang dilakukan oleh I Gede Wirayudhana pada tahun 2021 berjudul "*Klasifikasi Mutu Jambu Biji Getas Merah Berdasarkan Tekstur Menggunakan Gray Level Co-occurrence Matrix (GLCM) dan K-Nearest Neighbor (KNN)*". Metode GLCM digunakan untuk mengekstraksi fitur tekstur citra permukaan buah, yang kemudian diklasifikasikan ke dalam tiga kategori mutu, yaitu Super, A, dan B menggunakan KNN. Hasil penelitian menunjukkan bahwa akurasi terbaik yang diperoleh adalah sebesar 45,8% dengan nilai parameter $K=9$, yang menunjukkan keterbatasan dalam klasifikasi mutu jika hanya mengandalkan fitur tekstur permukaan [7].

2.1. Penyakit pada Buah Jambu Biji dan Kondisi Buah Sehat

Penyakit pada buah jambu biji, yang merupakan komoditas hortikultura bernilai ekonomi tinggi, seringkali menyebabkan kerusakan produksi. Suhu rendah dapat menghambat produksi buah, menyebabkan kerontokan bunga, dan memicu penyakit yang menyerang daun serta buah jambu biji. Deteksi dini penyakit ini sangat penting untuk mencegah penyebaran dan mengurangi risiko gagal panen, serta untuk membantu petani dalam mengambil tindakan pencegahan atau pengobatan yang tepat guna meningkatkan produktivitas panen dan mengurangi kerugian [8].

a. Penyakit Lalat Buah (*Bactrocera dorsalis*)

Penyakit lalat buah pada tanaman, termasuk jambu biji, disebabkan oleh serangan dari famili

Tephritidae, di mana spesies seperti *Bactrocera dorsalis* Complex dan *Bactrocera carambolae* merupakan lalat buah yang teridentifikasi menyerang buah jambu biji [9].

Secara visual, gejala awal serangan lalat buah pada buah jambu biji dapat terlihat sebagai bekas tusukan kecil pada kulit buah. Tusukan ini seringkali diikuti dengan munculnya bintik kecil berwarna coklat kehitaman di sekitar area tusukan. Bekas tusukan ini menjadi jalan masuk bagi larva lalat buah yang kemudian akan menetas dan berkembang biak di dalam buah. Aktivitas larva di dalam buah menyebabkan kerusakan internal pada daging buah, menjadikannya busuk dan tidak layak konsumsi, yang pada akhirnya menurunkan kualitas dan nilai jual buah secara signifikan [9].

b. Penyakit Antraknosa (*Colletotrichum gloeosporioides*)

Antraknosa merupakan salah satu penyakit penting yang menyerang tanaman cabai rawit (*Capsicum frutescens* L.), dan disebabkan oleh patogen *Colletotrichum* sp., termasuk *Colletotrichum gloeosporioides* [10].

Pada buah yang terinfeksi antraknosa, gejala yang ditunjukkan adalah adanya bercak bulat hingga oval berwarna coklat kehitaman yang membentuk lesi cekung pada permukaan buah. Bercak ini awalnya berukuran kecil, kemudian berkembang dan membentuk lesi cekung dengan perubahan miselium menjadi lebih gelap, serta dapat tersebar merata secara konsentris membentuk lingkaran. Pada bagian tengah bercak, terdapat bintik-bintik berwarna jingga yang sedikit berair. Bintik-bintik ini terkumpul pada satu titik dan akan menembus jauh ke dalam buah, mengakibatkan pembusukan. Buah yang terserang akan menjadi keriput dan mengering, dan penyakit ini dapat menyebabkan kerugian hingga 25-100% [10].

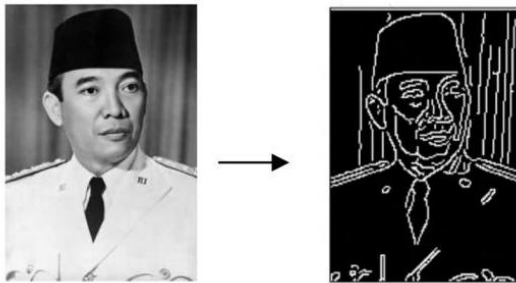
c. Buah Jambu Biji Sehat

Buah jambu biji yang sehat adalah buah yang tidak menunjukkan adanya tanda-tanda serangan hama maupun penyakit, serta memiliki karakteristik fisik yang optimal untuk mutu segarnya. Secara visual, buah jambu biji yang sehat dapat diidentifikasi melalui beberapa ciri: kulitnya berwarna hijau. Permukaan kulit buah yang sehat terasa mulus, dan buah yang matang secara umum memiliki tekstur yang lunak saat ditekan. Buah jambu biji merah umumnya berbentuk bulat. Kondisi umum buah terlihat segar dan padat. Penelitian menunjukkan bahwa kondisi fisik buah, susut bobot, dan tingkat kekerasan adalah indikator penting mutu segar buah jambu biji. Mempertahankan kondisi fisik, bobot buah, dan tingkat kekerasan ini sangat penting karena buah yang mengalami penyusutan bobot yang tinggi dapat menjadi keriput, dan pemeliharaan indikator ini dapat memperpanjang umur simpan buah [11].

2.2. Canny Edge Detector

Canny adalah salah satu metode deteksi tepi yang paling populer dan banyak digunakan. Popularitasnya didasarkan pada kemampuannya untuk mendeteksi kontur yang nyata dan konsisten. Metode ini diperkenalkan oleh John F. Canny pada tahun 1986. Pendekatan ini merupakan hasil dari optimasi berdasarkan tiga kriteria penting :

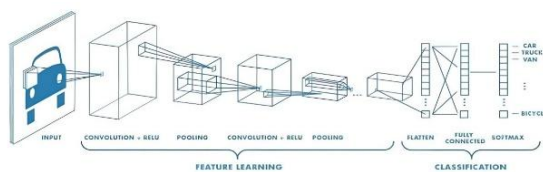
- Deteksi yang Baik (Good Detection)**
Semua tepi yang nyata (real) dalam gambar harus terdeteksi.
- Lokalisasi yang Baik (Good Localization)**
Tepi yang terdeteksi harus sedekat mungkin dengan posisi tepi yang sebenarnya.
- Respons Tunggal (Minimal Response)**
Setiap tepi hanya boleh ditandai satu kali oleh detektor (menghindari deteksi ganda/tepi tebal) [12].



Gambar 1. Canny edge detector

2.3. Convolutional Neural Network (CNN)

Convolutional Neural Network adalah jenis algoritma *deep learning* yang dapat menerima Input gambar dan menentukan obyek pada gambar yang dapat digunakan sebagai sumber mengenali gambar untuk membedakan antara setiap gambar yang ada [6]. CNN digunakan untuk mengklasifikasi data melalui metode *supervised Learning*, dimana ada data latih dan variabel yang ditargetkan. Oleh karena itu bertujuan untuk mengelompokkan data ke dalam kelompok yang sudah dikenal. CNN sering digunakan untuk mengidentifikasi objek, melakukan deteksi dan pembagian objek [13].

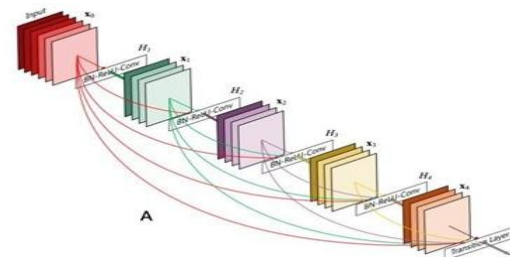


Gambar 2. Convolutional neural network

2.4. Arsitektur DenseNet

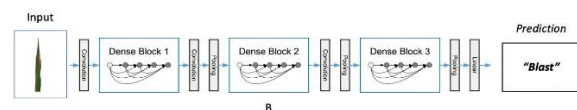
DenseNet (*Densely Connected Convolutional Network*) adalah salah satu arsitektur *Convolutional Neural Network* (CNN) yang inovatif dan terbukti sangat efektif, pertama kali diperkenalkan oleh Huang et al. pada tahun 2017. Tujuan DenseNet 121 adalah untuk meningkatkan akurasi dari Pelatihan model [15].

Konsep kunci yang membedakan DenseNet dari arsitektur CNN lainnya adalah "dense connectivity", di mana setiap lapisan dalam sebuah blok terhubung langsung ke semua lapisan berikutnya dalam blok yang sama [14].



Gambar 3. Arsitektur densenet

Gambar 3 adalah proses dari DenseNet. Lima lapisan menggunakan x sebanyak 4 sebagai laju pertumbuhan, dan semua masukan lapisan pada peta fitur diambil. DenseNet memerlukan parameter yang lebih sedikit daripada CNN tradisional. Oleh karena itu proses DenseNet tidak perlu mengetahui banyak tentang peta fitur. Jumlah parameter dalam arsitektur ResNets berada pada tingkat yang sangat penting dalam proses pembelajaran. Lapisan DenseNets, di sisi lain, sangat sempit (misalnya 12 filter) karena memerlukan kumpulan peta fitur berukuran kecil tambahan. Proses pembelajaran dalam Pelatihan menghadapi masalah aliran informasi dan gradien. Arsitektur DenseNet dapat menyelesaikan masalah ini dengan mengakses langsung Input, lapisan gradien fungsi kerugian gambar asli [15].



Gambar 4. Arsitektur densenet

Gambar 4 menunjukkan arsitektur DenseNet. Struktur DenseNet dapat dirumuskan sebagai berikut :

$$a[1] = g([a[0], a[1], a[2], \dots, a[l-1]]) \quad (1)$$

Gambar 4 adalah proses konvolusi dan membentuk Blok 1, diikuti oleh konvolusi lalu pooling. Proses dilanjutkan dengan proses Dense Blok 3, diikuti konvolusi dan pooling. Proses dilanjutkan dengan Dense Blok 3, diikuti pooling dan linier. Oleh karena itu sistem dapat mendeteksi dan mengklasifikasikan kelas target [15].

2.5. Confusion Matrix

Confusion matrix adalah alat yang sangat penting dalam mengukur kinerja suatu sistem klasifikasi. Ini memberikan wawasan yang mendalam tentang bagaimana model klasifikasi melakukan prediksi dan seberapa baik model tersebut dalam mengklasifikasikan data. Confusion matrix terdiri dari empat variabel kunci: True Positive (TP), True

Negative (TN), False Positive (FP), dan False Negative (FN). TP, yang merupakan singkatan dari True Positive, mewakili situasi ketika sistem klasifikasi berhasil memprediksi suatu kejadian sebagai positif, dan prediksi tersebut ternyata benar [16].

- True Positive (TP), yang menunjukkan jumlah kasus positif yang diprediksi dengan benar sebagai positif;
- True Negative (TN), yang merupakan jumlah kasus negatif yang diprediksi dengan benar sebagai negatif;
- False Positive (FP), yaitu jumlah kasus negatif yang salah diprediksi sebagai positif (sering disebut kesalahan Tipe I) ;
- False Negative (FN), yaitu jumlah kasus positif yang salah diprediksi sebagai negatif (kesalahan Tipe II).

Dalam penelitian ini, metrik akurasi akan digunakan untuk mengukur kinerja model DenseNet121 dalam mengklasifikasikan berbagai jenis penyakit pada buah jambu biji, memberikan gambaran yang akurat tentang efektivitas model yang dikembangkan. Dibawah adalah rumus Akurasi :

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \quad (2)$$

3. METODE PENELITIAN

Penelitian ini bertujuan untuk mengklasifikasikan penyakit pada buah jambu biji menggunakan metode Convolutional Neural Network (CNN) berbasis transfer learning dengan arsitektur DenseNet121. Penelitian dilakukan menggunakan Python pada platform Google Colaboratory dengan library TensorFlow dan Keras, serta menggunakan dataset citra yang telah melalui proses pre-processing dan augmentasi.

3.1. Pengumpulan Data

Sumber data dalam penelitian ini merupakan data primer yang dikumpulkan secara langsung oleh peneliti. Pengambilan data dilakukan di beberapa lokasi perkebunan buah jambu biji Kristal di Kabupaten Kubu Raya, Kalimantan Barat, yaitu Rasau Jaya 2, Jalan Sungai Bulan, dan Jalan Patok 36. Citra diambil menggunakan kamera smartphone dengan variasi pencahayaan dan sudut pengambilan.

Table 1. Jenis penyakit buah jambu biji

		
Antaknosa	Lalat buah	Buah sehat

Data yang dikumpulkan berupa 1500 citra buah jambu biji yang terbagi ke dalam tiga kelas, yaitu

jambu biji sehat, penyakit lalat buah, dan penyakit antraknosa, masing-masing sebanyak 500 citra.

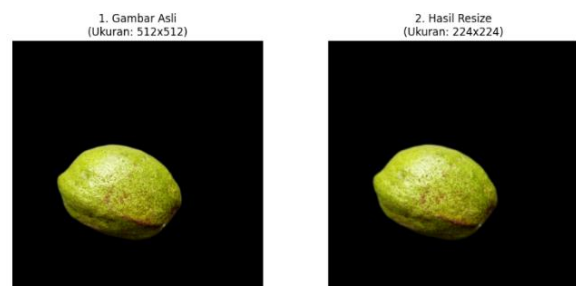
Table 2. Dataset

Jenis data	Jumlah Gambar	Jumlah Per-Gambar	Jumlah Kelas
Data latih	1200	400	3 kelas
Data validasi	150	50	3 kelas
Data Uji	150	50	3 kelas

Dataset kemudian dibagi menjadi 80% data latih, 10% data validasi, dan 10% data uji untuk mendukung proses pelatihan dan evaluasi model secara objektif.

3.2. Pre-processing Data

Citra yang dikumpulkan diubah ukurannya menjadi 224×224 piksel sesuai dengan kebutuhan input arsitektur DenseNet121. Selanjutnya, citra dinormalisasi ke rentang $[0,1]$ dan dilakukan augmentasi berupa rotasi, horizontal flipping, zooming, shifting, dan shear untuk meningkatkan variasi data.



Gambar 5. Pre-processing data

Selain itu, diterapkan metode Canny Edge Detection untuk mengekstraksi ciri tepi pada citra, sehingga model dapat lebih fokus pada pola bentuk dan tekstur objek. Tahap pre-processing ini bertujuan untuk menyeragamkan data, memperkaya variasi citra, serta mengurangi risiko overfitting.

3.3. Arsitektur Model

Penelitian ini menggunakan metode transfer learning dengan arsitektur DenseNet121 sebagai base model. Model yang digunakan merupakan model pre-trained pada dataset ImageNet dengan parameter `include_top=False`, sehingga lapisan klasifikasi asli dihilangkan dan hanya digunakan sebagai ekstraktor fitur.

Seluruh lapisan pada base model dibekukan (freeze), sehingga bobot tidak diperbarui selama proses pelatihan. Selanjutnya, ditambahkan custom layer yang terdiri dari GlobalAveragePooling2D, lapisan Dense dengan 128 neuron dan fungsi aktivasi ReLU, serta lapisan Dropout untuk mengurangi overfitting. Pada bagian akhir, digunakan lapisan Dense dengan fungsi aktivasi softmax untuk mengklasifikasikan citra ke dalam tiga kelas.

3.4. Ekstraksi Ciri

Pada penelitian ini, ekstraksi ciri dilakukan menggunakan pendekatan hibrida, yaitu kombinasi metode manual dan otomatis. Ekstraksi ciri manual dilakukan dengan menerapkan metode Canny Edge Detection pada tahap pre-processing untuk mendeteksi tepi dan struktur utama pada citra buah jambu biji.

Proses ini meliputi konversi citra ke grayscale, penerapan Gaussian blur untuk mengurangi noise, serta deteksi tepi menggunakan algoritma Canny. Hasil deteksi tepi kemudian dikonversi menjadi citra 3-channel dan dinormalisasi sebelum digunakan sebagai input model.

Selanjutnya, ekstraksi ciri otomatis dilakukan oleh arsitektur DenseNet121 melalui lapisan konvolusi yang mampu mempelajari pola fitur yang lebih kompleks dari citra. Kombinasi kedua pendekatan ini bertujuan untuk meningkatkan kemampuan model dalam mengenali pola penyakit berdasarkan bentuk dan tekstur citra [17].

3.5. Pelatihan Sistem

Pelatihan model dilakukan menggunakan Google Colaboratory dengan optimizer Adam dan learning rate 0.0001 serta fungsi loss categorical crossentropy. Proses pelatihan menggunakan data latih dan divalidasi dengan data validasi. Untuk mencegah overfitting, digunakan callback berupa ModelCheckpoint dan EarlyStopping.

3.6. Pengujian dan Evaluasi Model

Model dievaluasi menggunakan data uji dengan metrik akurasi, precision, recall, dan F1-score. Hasil pengujian disajikan dalam bentuk confusion matrix untuk melihat performa klasifikasi pada setiap kelas.

3.7. Implementasi

Tahap implementasi dilakukan dengan mengintegrasikan model CNN berbasis DenseNet121 ke dalam aplikasi berbasis web menggunakan Streamlit. Aplikasi ini memungkinkan pengguna mengunggah citra buah jambu biji dan memperoleh hasil klasifikasi secara langsung.

4. HASIL DAN PEMBAHASAN

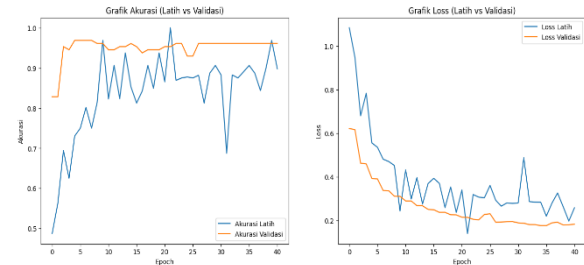
4.1. Hasil Pelatihan

Proses pelatihan menghasilkan model dengan performa yang baik. Berdasarkan hasil evaluasi, model mencapai akurasi sebesar 91,33% pada data uji. Hasil ini menunjukkan bahwa model mampu mengklasifikasikan citra buah jambu biji ke dalam tiga kelas dengan cukup akurat.

4.2. Hasil Proses Pelatihan (Epoch)

Pelatihan model DenseNet121 dilakukan dengan parameter learning rate 0,0001, batch size 32, dan optimizer Adam. Untuk mengoptimalkan waktu pelatihan dan mencegah overfitting, diterapkan mekanisme Early Stopping dengan patience 5 dan

ModelCheckpoint untuk menyimpan bobot terbaik berdasarkan validation loss terendah. Meskipun dikonfigurasi hingga 100 epoch, proses pelatihan mencapai titik konvergensi dan berhenti secara otomatis pada epoch ke-36. Performa model selama pelatihan disajikan pada Gambar 6.



Gambar 6. Grafik akurasi dan loss pelatihan

Berdasarkan Gambar 6, kurva akurasi menunjukkan tren peningkatan yang stabil hingga melampaui 90%. Akurasi validasi yang sedikit lebih tinggi dari akurasi latih mengindikasikan bahwa model mampu menangani variasi data hasil augmentasi dengan baik. Sementara itu, grafik loss yang menurun tajam dan konvergen tanpa jarak (gap) signifikan antara kurva latih dan validasi membuktikan bahwa model tidak mengalami overfitting.

```
Epoch 35: val_loss improved from 0.18189 to 0.17673, saving model to /content/drive/MyDrive/IshabMuhammad211220021_Peng
37/37 ————— 24s 662ms/step - accuracy: 0.8874 - loss: 0.3062 - val_accuracy: 0.9609 - val_loss: 0.1767
Epoch 36/100 ————— 3s 108ms/step - accuracy: 0.9062 - loss: 0.2208
Epoch 36: val_loss improved from 0.17673 to 0.17664, saving model to /content/drive/MyDrive/IshabMuhammad211220021_Peng
37/37 ————— 3s 72ms/step - accuracy: 0.9062 - loss: 0.2208 - val_accuracy: 0.9609 - val_loss: 0.1766
Epoch 37/100 ————— 0s 574ms/step - accuracy: 0.8739 - loss: 0.3048
Epoch 37: val_loss did not improve from 0.17664
37/37 ————— 23s 624ms/step - accuracy: 0.8742 - loss: 0.3041 - val_accuracy: 0.9609 - val_loss: 0.1895
Epoch 38/100 ————— 4s 117ms/step - accuracy: 0.8848 - loss: 0.3263
Epoch 38: val_loss did not improve from 0.17664
37/37 ————— 2s 41ms/step - accuracy: 0.8838 - loss: 0.3263 - val_accuracy: 0.9609 - val_loss: 0.1922
Epoch 39/100 ————— 0s 566ms/step - accuracy: 0.8970 - loss: 0.2640
Epoch 39: val_loss did not improve from 0.17664
37/37 ————— 23s 609ms/step - accuracy: 0.8973 - loss: 0.2639 - val_accuracy: 0.9609 - val_loss: 0.1798
Epoch 40/100 ————— 3s 80ms/step - accuracy: 0.9008 - loss: 0.1978
Epoch 40: val_loss did not improve from 0.17664
37/37 ————— 1s 8ms/step - accuracy: 0.9008 - loss: 0.1978 - val_accuracy: 0.9609 - val_loss: 0.1801
Epoch 41/100 ————— 0s 584ms/step - accuracy: 0.9060 - loss: 0.2477
Epoch 41: val_loss did not improve from 0.17664
37/37 ————— 23s 627ms/step - accuracy: 0.9058 - loss: 0.2480 - val_accuracy: 0.9609 - val_loss: 0.1886
Epoch 41: early stopping
Restoring model weights from the end of the best epoch: 36.
```

Gambar 7. Model terbaik epoch-36

Detail performa model pada titik terbaik (epoch ke-36) yang disimpan melalui *ModelCheckpoint* dirinci pada Tabel 3.

Table 3. Model terbaik epoch-36

Parameter Evaluasi	Nilai	Keterangan
Validation Accuracy	96.09%	Akurasi tertinggi pada data validasi
Validation Loss	0.1766	Nilai error terendah (Optimal)
Training Accuracy	90.62%	Akurasi pada data latih
Training Loss	0.22	Nilai error pada data latih

Pencapaian akurasi validasi sebesar 96,09% dengan nilai loss yang sangat rendah (0,1766) mengonfirmasi bahwa integrasi arsitektur DenseNet121 dan metode *Canny Edge Detection* sangat efektif.

4.3. Hasil Evaluasi Kinerja Model pada Data Uji

Hasil evaluasi menunjukkan akurasi sebesar 91,33% dengan nilai test loss sebesar 0,1841.

Test Loss: 0.1841
Test Accuracy: 0.9133 (91.33%)

Gambar 8. Evaluasi model pada data uji

Capaian akurasi ini secara signifikan melampaui ambang batas kinerja (performance threshold) minimum yang ditetapkan sebesar 80%.

4.4. Laporan Klasifikasi (Presisi, Recall, F1-Score)

Hasil evaluasi ini diperoleh dari perbandingan prediksi model terhadap data uji sebanyak 150 citra (50 citra per kelas). Hasilnya ditunjukkan pada Gambar 9 dan dirangkum dalam Tabel 4.

Laporan Klasifikasi (Data Uji):				
	precision	recall	f1-score	support
Anthraco	0.86	0.88	0.87	50
Fruit_fly	0.88	0.86	0.87	50
Healthy_guava	1.00	1.00	1.00	50
accuracy			0.91	150
macro avg	0.91	0.91	0.91	150
weighted avg	0.91	0.91	0.91	150

Gambar 9. Laporan Klasifikasi

Table 4. Laporan Klasifikasi

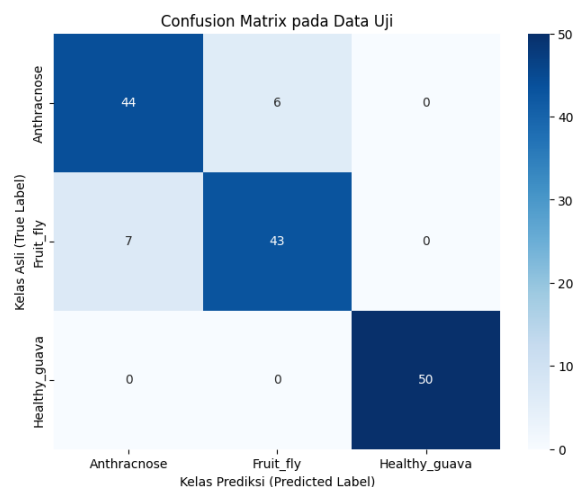
Kelas	Presisi	Recall	F1-Score	Support
Anthraco	0.86	0.88	0.87	50
Fruit Fly	0.88	0.86	0.87	50
Healthy Guava	1.00	1.00	1.00	50
Akurasi Total			0.91	150

Berdasarkan pengujian pada data uji, model menghasilkan akurasi rata-rata sebesar 91%. Berikut adalah analisis performa pada masing-masing kelas:

- Healthy (Sehat): Mencapai performa sempurna (F1-Score 1.00) dengan nilai Presisi dan Recall 1.00. Hasil ini membuktikan efektivitas metode Canny Edge Detection dalam memisahkan fitur buah sehat dari tekstur buah berpenyakit tanpa kesalahan.
- Anthraco (Antraknosa): Memiliki performa yang sangat baik dengan F1-Score 0.87. Nilai Recall (0.88) yang sedikit lebih tinggi dari Presisi menunjukkan model sangat sensitif dalam mengenali gejala antraknosa, meskipun terdapat sedikit risiko salah identifikasi pada gejala lalat buah.
- Fruit Fly (Lalat Buah): Mencapai F1-Score 0.87. Adanya kesalahan klasifikasi sebesar 14% (Recall 0.86) mengindikasikan adanya kemiripan fitur morfologi tepi pada bercak lalat buah dengan antraknosa, sehingga model terkadang sulit membedakan keduanya pada kondisi tertentu.

4.5. Hasil Confusion Matrix

Confusion Matrix digunakan untuk memvisualisasikan secara rinci di mana letak kesalahan prediksi model pada 150 data uji. Hasilnya ditunjukkan pada Gambar 10.



Gambar 10. Hasil confusion matrix

Berdasarkan matriks evaluasi, model berhasil memprediksi secara tepat 137 dari 150 data uji. Analisis mendalam terhadap kesalahan klasifikasi dirinci sebagai berikut:

- Kelas Anthracnose: Dari 50 sampel, 44 diprediksi benar. Terdapat 6 citra yang terklasifikasi salah sebagai Fruit Fly, namun tidak ada kesalahan prediksi ke kelas Healthy.
- Kelas Fruit Fly: Dari 50 sampel, 43 diprediksi benar. Sebanyak 7 citra salah diprediksi sebagai Anthracnose. Kesalahan ini merupakan kebingungan utama model dalam membedakan kedua jenis penyakit tersebut.
- Kelas Healthy: Mencapai akurasi sempurna (100%) tanpa ada kesalahan klasifikasi, membuktikan model sangat handal dalam membedakan buah sehat dari buah berpenyakit.

Analisis Teknis:

Kebingungan model dalam membedakan Fruit Fly dan Anthracnose disebabkan oleh penggunaan Canny Edge Detection yang mengeliminasi informasi warna. Tanpa fitur warna (hitam pada Antraknosa vs coklat pada Lalat Buah), model hanya mengandalkan fitur morfologi tepi. Karena kedua penyakit memiliki pola bercak lingkaran yang identik pada citra tepi, model mengidentifikasi keduanya sebagai fitur yang serupa, sehingga memicu terjadinya misklasifikasi pada kedua kelas tersebut.

Akurasi pada pengujian hasil Confusion matrix ini dihitung menggunakan rumus akurasi :

Diketahui :

$$TP + TN \text{ (Prediksi Benar)} = 137$$

$$TP + FP + FN + TN \text{ (Total Data)} = 150$$

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \quad (3)$$

$$\text{Accuracy} = \frac{137}{150}$$

Accuracy = 0.91,33

Hasil pengujian Confusion matrix menunjukkan bahwa model memiliki tingkat akurasi sebesar 91.33%.

4.6. Hasil Pengujian Antarmuka

Pengujian ini bertujuan untuk menunjukkan kinerja model terbaik (hasil dari Epoch 36) saat diterapkan pada antarmuka aplikasi. Total citra uji yang digunakan adalah 150 citra (50 per kelas). Data prediksi benar diambil dari hasil Confusion Matrix (CM) model yang telah dievaluasi pada data uji. Rinciannya dapat dilihat pada Tabel 5 berikut :

Table 5. pengujian model pada antarmuka

Kelas Penyakit	Jumlah Citra Diuji	Jumlah Citra yang Benar	Jumlah Citra yang Salah
Anthraco nose	50	44	6
Fruit_fly	50	45	5
Healthy_guava	50	50	0
Total	150	139	11

Berdasarkan Tabel 5, total citra yang berhasil dikenali dengan benar adalah 139 dari total 150 citra. Akurasi pada pengujian antarmuka ini dihitung menggunakan rumus akurasi :

Diketahui :

TP + TN (Prediksi Benar) = 139

TP + FP + FN + TN (Total Data) = 150

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \quad (4)$$

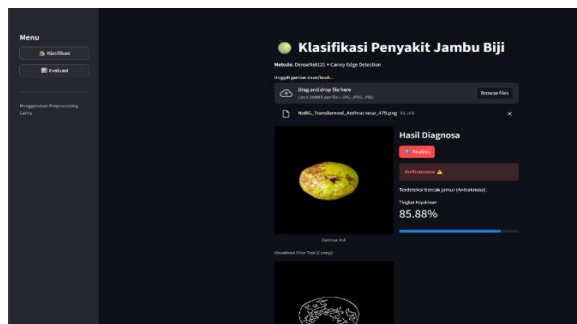
$$\text{Accuracy} = \frac{139}{150}$$

$$\text{Accuracy} = 0.92,67$$

Hasil pengujian antarmuka menunjukkan bahwa model memiliki tingkat akurasi sebesar 92.67%.

4.7. Hasil Perancangan Antarmuka Pengguna (GUI)

Aplikasi berhasil dibangun menggunakan Streamlit. Tampilan antarmuka website aplikasi ditunjukkan pada Gambar 11.



Gambar 11. Hasil perancangan antarmuka pengguna

5. KESIMPULAN DAN SARAN

Penelitian ini membuktikan bahwa metode Convolutional Neural Network (CNN) dengan arsitektur DenseNet121 yang dikombinasikan dengan

Canny Edge Detection mampu mengklasifikasikan penyakit jambu biji dengan akurasi pengujian yang tinggi sebesar 91,33% dan akurasi antarmuka sebesar 92,67%. Capaian ini secara signifikan melampaui ambang batas kinerja 80%, dengan keberhasilan sempurna (100%) pada kelas buah sehat, yang menunjukkan bahwa fitur struktural hasil deteksi tepi sangat efektif meskipun tanpa informasi warna. Namun, model masih mengalami sedikit misklasifikasi antara penyakit Antraknosa dan Lalat Buah akibat kemiripan pola morfologi pada citra tepi. Oleh karena itu, untuk penelitian selanjutnya disarankan penerapan metode hybrid yang menggabungkan fitur warna (RGB) dan fitur tepi (Canny), serta pengembangan aplikasi ke platform mobile menggunakan TensorFlow Lite agar dapat memberikan manfaat diagnosa secara real-time bagi petani di lapangan.

DAFTAR PUSTAKA

- [1] Dinas Tanaman Pangan dan Hortikultura Provinsi Kalimantan Barat, "Data Produksi Tanaman Buah Sayur Tahunan (BST) 2019-2023," distan.kalbarprov.go.id. Accessed: Sep. 10, 2025. [Online]. Available: <https://distan.kalbarprov.go.id/data-dan-informasi/hortikultura/>
- [2] I. Riati, Yuhandri, and G. W. Nurcahyo, "Penerapan Convolutional Neural Network Untuk Mengidentifikasi Penyakit Tanaman Kelapa Sawit," *Jurnal KomtekInfo*, pp. 237–246, Sep. 2024, doi: 10.35134/komtekinfo.v11i4.554.
- [3] A. H. P. Sitohang, T. I. Hermanto, and C. D. Lestari, "KLASIFIKASI JENIS PENYAKIT PADA DAUN TUMBUHAN STROBERI MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK ARSITEKTUR INCEPTIONV3," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3S1, Oct. 2024, doi: 10.23960/jitet.v12i3S1.5274.
- [4] T. Y. P. Situngkir, "Klasifikasi Penyakit pada Daun Kentang Menggunakan Pengolahan Citra dengan Metode Convolutional Neural Network (CNN)," Skripsi, Universitas Pembangunan Nasional "Veteran" Yogyakarta, Yogyakarta, 2022.
- [5] D. Iswanto and D. Handayani UN, "Klasifikasi Penyakit Tanaman Jagung Menggunakan Metode Convolutional Neural Network (CNN)," *Jurnal Ilmiah Universitas Batanghari Jambi*, vol. 22, no. 2, p. 900, Jul. 2022, doi: 10.33087/jiubj.v22i2.2065.
- [6] F. Putrawansyah, "Penerapan Metode Support Vector Machine Terhadap Klasifikasi Jenis Jambu Biji," *JIKO (Jurnal Informatika dan Komputer)*, vol. 8, no. 1, p. 193, Feb. 2024, doi: 10.26798/jiko.v8i1.988.
- [7] I. Gede Wirayudhana, "KLASIFIKASI MUTU BUAH JAMBU BIJI GETAS MERAH

- BERDASARKAN TEKSTUR MENGGUNAKAN GREY LEVEL CO-OCCURENCE MATRIX (GLCM) DENGAN KLASIFIKASI KNN,” vol. 2, no. 6, 2021.
- [8] S. Keputusan *et al.*, “Terakreditasi SINTA Peringkat 3 Klasifikasi Penyakit Pada Daun dan Buah Jambu Menggunakan Convolutional Neural Network,” 2026.
- [9] W. Widiastuty, R. Amalia, W. Fadhillah, and S. Utami, “INVENTARISASI DAN IDENTIFIKASI HAMA LALAT BUAH PADA BUAH JAMBU BIJI (*Psidium guajava*), JAMBU AIR (*Syzygium aqueum*) DAN JERUK (*Citrus sp.*),” *Jurnal SOMASI (Sosial Humaniora Komunikasi)*, vol. 3, no. 2, pp. 10–27, Dec. 2022, doi: 10.53695/js.v3i2.812.
- [10] A. N. Asyifa and A. L. Abadi, “UJI EFEKTIVITAS BAHAN AKTIF FLUXAPYROXAD 50% + PYRACLOSTROBIN 50% DENGAN PENAMBAHAN AGENS HAYATI *Bacillus subtilis* TERHADAP PENYAKIT ANTRAKNOSA (*Colletotrichum gloeosporioides*) PADA TANAMAN CABAI RAWIT (*Capsicum frutescens L.*),” *Jurnal Hama dan Penyakit Tumbuhan*, vol. 13, no. 1, pp. 32–46, Mar. 2025, doi: 10.21776/ub.jurnalhpt.2025.013.1.3.
- [11] A. Lastryanto, B. I. Bintoro, L. C. Hawa, and S. A. Wibowo, “Pengawetan Buah Jambu Biji (*Psidium Guajava L.*) Segar dengan Teknologi Hypobaric Storage,” *Jurnal Keteknik Pertanian Tropis dan Biosistem*, vol. 10, no. 1, pp. 55–65, Apr. 2022, doi: 10.21776/ub.jkptb.2022.010.01.07.
- [12] A. C. Siregar, B. S. W. Poetro, B. C. Octariadi, Robet, and Sucipto, *Buku Ajar Pengolahan Citra Digital*. Bantul, Daerah Istimewa Yogyakarta: PT. Green Pustaka Indonesia, 2025.
- [13] M. H. Ahmad, F. M. Hana, T. G. Pratama, and H. Aulida, “Klasifikasi Empat Jenis Daun Herbal Menggunakan Metode Convolutional Neural Network,” *Jurnal Ilmu Komputer dan Matematika*, vol. 4, no. 2, pp. 69–76, 2023.
- [14] K. Setiyanto and M. Bolang, “ANALISIS PERBANDINGAN HASIL KLASIFIKASI JENIS PENYAKIT TANAMAN TOMAT MENGGUNAKAN ARSITEKTUR MOBILENET, DENSENET121, DAN XCEPTION,” *JTS*, vol. 3, no. 3.
- [15] A. D. Saputra, D. Hindarto, and H. Santoso, “Disease Classification on Rice Leaves using DenseNet121, DenseNet169, DenseNet201,” *Sinkron*, vol. 8, no. 1, pp. 48–55, Jan. 2023, doi: 10.33395/sinkron.v8i1.11906.
- [16] S. Suhendar, A. Purnama, and E. Fauzi, “Deteksi Penyakit Pada Daun Tanaman Ubi Jalar Menggunakan Metode Convolutional Neural Network,” *Jurnal Ilmiah Informatika Global*, vol. 14, no. 3, pp. 62–67, Dec. 2023, doi: 10.36982/jiig.v14i3.3478.
- [17] Q. N. Azizah, “Klasifikasi Penyakit Daun Jagung Menggunakan Metode Convolutional Neural Network AlexNet,” *sudo Jurnal Teknik Informatika*, vol. 2, no. 1, pp. 28–33, Feb. 2023, doi: 10.56211/sudo.v2i1.227