

Implementasi Deep Learning Untuk Pengenalan Penyakit Antraks Pada Buah Cabai

Diterima: 10 Juni 2024
Revisi: 10 Juli 2024
Terbit: 1 Agustus 2024

^{1*}Bella Nurbuana Tri Cahya Ningrum, ²Umi Mahdiyah, ³Daniel Swanjaya
¹⁻³Universitas Nusantara PGRI Kediri
E-mail : ¹Bellanurbuana@gmail.com,
²umimahdiyah@unpkediri.ac.id, ³daniel@unpkediri.ac.id

Abstrak— Penelitian ini mengimplementasikan *Deep Learning*, khususnya *Convolutional Neural Network* (CNN), untuk mendeteksi penyakit antraks pada buah cabai. CNN digunakan untuk mengklasifikasikan citra cabai terinfeksi dan tidak terinfeksi. Data gambar dikumpulkan dan diproses menggunakan TensorFlow's *ImageDataGenerator* untuk normalisasi, resize, dan pembagian batch. Grafik akurasi pada data pelatihan dan validasi dipantau selama proses pelatihan, sementara akurasi pada data pengujian dievaluasi setelahnya. Penelitian ini diharapkan dapat memberikan kontribusi dalam pengendalian penyakit antraks pada tanaman cabai, meningkatkan hasil panen, dan mengurangi kegagalan panen yang disebabkan oleh penyakit. Implementasi pengenalan penyakit cabai menggunakan Convolutional Neural Network (CNN) menunjukkan bahwa CNN berhasil mengklasifikasikan penyakit Antraks pada cabai. Model mencapai akurasi validasi terbaik sebesar 90% pada epoch ke-23, berdasarkan grafik Training and Validation Loss serta Training and Validation Accuracy.

Kata Kunci— *Convolutional Neural Network* (CNN); Antraks;Cabai

Abstract— The study implements *Deep Learning*, particularly *Convolutional Neural Network* (CNN), to detect anthracnose disease in chili peppers. CNN is employed to classify images of infected and uninfected chili peppers. Image data is collected and processed using TensorFlow's *ImageDataGenerator* for normalization, resizing, and batch splitting. Accuracy graphs during training and validation are monitored, while accuracy on test data is evaluated afterward. This research is expected to contribute to anthracnose disease control in chili plants, improving harvest yields, and reducing crop failures due to diseases. The implementation of chili disease recognition using Deep Learning with Convolutional Neural Network (CNN) shows that the CNN architecture can classify Anthracnose disease in chili by labeling the input data. Based on the Training and Validation Loss and Training and Validation Accuracy graphs, the model achieved the best validation accuracy of 90% at epoch 23.

Keywords— *Convolutional Neural Network* (CNN); Anthracnose;Chili

This is an open access article under the CC BY-SA License.



Penulis Korespondensi:

Bella Nurbuana Tri Cahya Ningrum
Teknik Informatika
Universitas Nusantara PGRI Kediri
Bellanurbuana@gmail.com
ID Orcid: [<https://orcid.org/register>]
Handphone: 085853401627

I. PENDAHULUAN

Cabai rawit (*Capsicum frutescens L.*) adalah komoditas hortikultura di Indonesia [1]. Tanaman cabai memerlukan perawatan untuk mendapatkan hasil optimal [2]. Panen cabai sering mengalami kegagalan karena penyakit [3]. Antraks, disebabkan oleh jamur *Colletotrichum spp.*, merupakan tantangan serius bagi pertumbuhan tanaman cabai di wilayah tropis dan subtropis [4]. Oleh karena itu, pengendalian yang efektif diperlukan untuk mencegah hal ini [5]. Diperlukan metode untuk mendeteksi dan mengklasifikasikan penyakit antraks pada cabai. Pendekatan menggunakan *Deep Learning* dengan (CNN), klasifikasi citra memproses data melalui beberapa lapisan tersembunyi [6] [7] [8]. *Convolutional* layer adalah komponen CNN. Setiap filter menghasilkan satu feature map dengan menggabungkan informasi subregional gambar, menambahkan bias, dan menerapkan fungsi aktivasi [9].

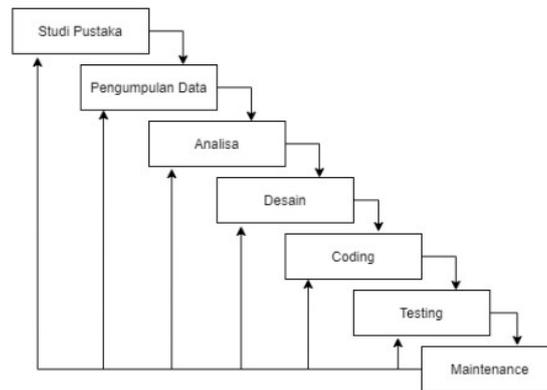
Adapun beberapa penelitian yang dilakukan oleh Rosalina dan Ardi Wijaya berjudul "Pendeteksian Penyakit pada Daun Cabai dengan Metode Deep Learning" [10]. Selanjutnya penelitian yang dilakukan oleh Ulyy Nuhanatika yang berjudul "Penentuan Tingkat Kematangan Cabe Rawit (*Capsicum frutescens L.*) Berdasarkan *Gray Level Co-Occurrence Matrix*" [11].

Berdasarkan permasalahan diatas maka peneliti membuat penelitian tentang Implementasi *Deep Learning*. Metode yang digunakan adalah *CNN*. Digunakan Untuk mengukur akurasi model, grafik akurasi pada data pelatihan dan validasi ditampilkan selama proses pelatihan, dan akurasi pada data pengujian dicetak setelah evaluasi model. Data buah cabai yang terinfeksi dan tidak terinfeksi dikumpulkan dari berbagai sumber, diorganisir dalam *DataFrame* yang berisi jalur file gambar dan labelnya. Pengolahan citra *ImageDataGenerator* dari TensorFlow, mencakup normalisasi nilai piksel, pengubahan ukuran, dan pemisahan data menjadi batch pelatihan dan validasi. Hasil penelitian diharapkan mampu mengenali penyakit Antraks pada buah Cabai.

II. METODE

2.1 Metode penelitian

Pada gambar 1 merupakan metode *waterfall* yang digunakan dalam penelitian ini, berikut penjelasannya.



Gambar 1 Metode *waterfall*

Metode yang digunakan dalam penelitian ini adalah *waterfall* seperti pada gambar 1. Proses yang digunakan memiliki beberapa tahapan dari persiapan data sampai data gambar berhasil klasifikasikan dengan *Deep Learning*.

2.2 Studi pustaka

Pengumpulan data dan informasi dilakukan dari hasil penelitian sebelumnya. Penelitian ini bertujuan untuk mengembangkan model klasifikasi *Deep Learning* untuk mendeteksi penyakit buah cabai. Tujuan proyek diidentifikasi, termasuk pembagian data, preprocessing, pembangunan model, dan evaluasi hasil.

2.3 Pengumpulan data

Penelitian dimulai dengan memilih data sampel untuk training, validasi, dan uji, lalu merancang jaringan *CNN* untuk klasifikasi [12]. Dalam penelitian ini, 236 data dibagi menjadi tiga: data training (80%), data validasi (10%) untuk memantau kinerja model dan menyesuaikan parameter selama pelatihan, dan data test (10%) untuk menilai hasil akhir model.

2.4 Analisa

Analisa dilakukan untuk memastikan dataset siap digunakan dalam pelatihan model *CNN preprocessing*. Citra dibagi data menjadi tiga kategori, data latih, data validasi, serta data uji [13]. agar dapat meningkatkan kualitas citra, mempercepat, dan mempermudah kinerja sistem dalam mendeteksi penyakit pada tanaman cabai [14]. Preprocessing mencakup penyesuaian gambar ke 224x336 piksel, normalisasi nilai piksel ke rentang [0, 1], dan pembentukan batch (64 untuk data latih dan validasi, batch optimal untuk data uji). Augmentasi data meningkatkan generalisasi model, dan visualisasi sampel gambar memverifikasi label sebelum pelatihan.



Gambar 2 dataset cabai antraks

Pada gambar 2 menunjukkan gambar cabai berpenyakit antraks sebelum di lakukan proses *preprocessing*.



Gambar 3 cabai Antraks

Gambar 3 menunjukkan dataset cabai setelah dilakukan proses *preprocessing* pada dataset.

2.5 Desain

Program dibuat menggunakan *TensorFlow* dan *Keras* dimulai dengan *import library*. Data gambar dibagi menjadi train, test, dan validasi, kemudian diaugmentasi dan dinormalisasi menggunakan *ImageDataGenerator*. Model *CNN* dibangun dengan *pre-trained Xception* sebagai base model, ditambah lapisan tambahan yang tidak dilatih ulang. Model dikompilasi dengan *Adam optimizer* dan *categorical_crossentropy*, lalu dilatih selama 100 epoch. Evaluasi dilakukan dengan data testing dan hasilnya ditampilkan.

Berikut ini struktur model *CNN* yang terdapat pada tabel 1 dari hasil penelitian pengenalan penyakit Antraks cabai.

Tabel 1 Struktur model *CNN*

Tabel Struktur Model <i>CNN</i>		
<i>Layer (type)</i>	Output Shape	Param#
Xception (base model)	(None, 7, 10, 2048)	20,861,480
Conv2D	(None, 7, 10, 32)	589,856
Max_pooling2D	(None, 3, 5, 32)	0

Dropout	(None, 3, 5, 32)	0
Flatten	(None, 480)	0
Dense	(None, 2)	962

Total params: 21,451,779
 Trainable params: 591,299
 Non-trainable params: 20,860,480

Ini adalah jumlah keseluruhan parameter. Ini termasuk parameter dari model *Xception* yang sudah dilatih sebelumnya Trainable Params adalah parameter yang akan diperbarui selama pelatihan model. Sedangkan Trainable Params adalah parameter yang tidak akan diperbarui selama pelatihan model. Ini berasal dari lapisan base model *Xception* yang sudah dilatih sebelumnya.

2.6 Coding

Proses coding menggunakan model *Deep Learning CNN* bagian dari *Artificial Intelligence* yang menggunakan jaringan saraf tiruan dan algoritma yang terinspirasi oleh cara kerja otak manusia [15]. Diproses dengan *TensorFlow* dan *Keras*, lalu dilatih dengan data *training* dan divalidasi selama 100 epoch.

2.7 Testing dan Evaluasi

Model dievaluasi menggunakan data testing dan hasilnya divisualisasikan dengan plot loss, akurasi, confusion matrix, dan classification report. Plot loss menunjukkan perubahan nilai loss selama pelatihan, confusion matrix menampilkan jumlah prediksi benar dan salah untuk setiap kelas, dan classification report merangkum precision, recall, serta F1-score untuk setiap kelas. Model yang telah dilatih disimpan, kesalahan klasifikasi dianalisis untuk memahami pola kesalahan dan meningkatkan kinerja, serta akurasi model pada data pengujian dihitung untuk mengukur kinerjanya dalam mengklasifikasikan gambar baru.

III. HASIL DAN PEMBAHASAN

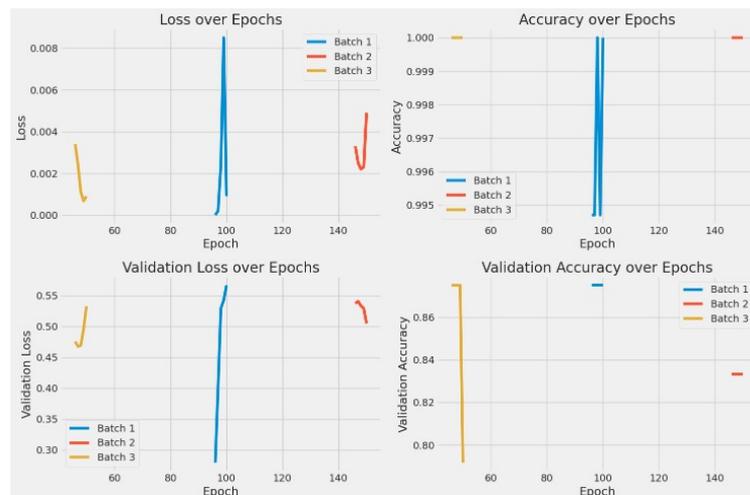
3.1 Epoch

Dilakukan tiga kali percobaan dengan Epoch 50, 100 dan 150 dengan batch size 128 pada penelitian ini hingga didapatkan pada tabel berikut.

Tabel 2 Uji

Epoch	Time per Step	Loss	Accuracy	Validation Loss	Validation Accuracy	Precision	Recall
50	11s 5s	8.9797	10000	0.5326	0.7917	1.0	0.75
100	9s 3s	9.0741	1.0000	0.5667	0.8750	1.0	0.75
150	10s 7s	0.0049	1.0000	0.5049	0.8333	1.0	0.75

Pelatihan model selama 50 epoch menghasilkan loss dan akurasi sempurna pada data pelatihan, dengan akurasi validasi sekitar 0.8750 dan loss yang sedikit lebih rendah, berkisar antara 0.4674 hingga 0.5326, menunjukkan performa yang baik dan stabil. Pelatihan selama 100 epoch juga mencapai loss dan akurasi sempurna pada data pelatihan, dengan akurasi validasi tetap di angka 0.8750, mirip dengan hasil pada 50 epoch, meskipun loss sedikit berfluktuasi antara 0.2791 hingga 0.5667. Hasil ini baik, namun tidak secara signifikan lebih baik dibandingkan dengan 50 epoch. Sementara itu, pelatihan model selama 150 epoch mencapai loss dan akurasi sempurna pada data pelatihan dengan akurasi 1.0000 dan loss yang sangat rendah. Namun, akurasi validasi stabil di angka 0.8333 dengan fluktuasi loss antara 0.5049 hingga 0.5473. Precision yang sempurna (1.0) dan recall yang sedikit lebih rendah (0.75) menunjukkan bahwa model sangat akurat dalam prediksi positif (tidak ada False Positives), tetapi masih ada beberapa kasus positif yang terlewat (False Negatives ada 1 kasus). Validation accuracy bervariasi dengan epoch, menunjukkan perubahan performa model pada data validasi. Validation loss juga bervariasi, yang menunjukkan seberapa baik model meminimalkan kesalahan pada data validasi..



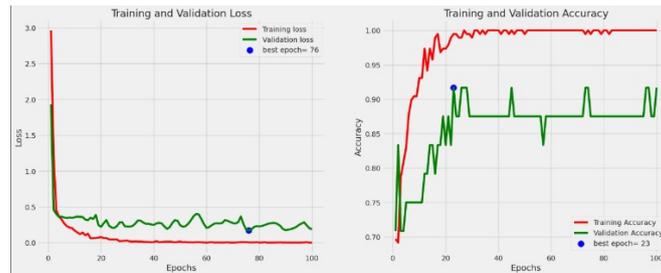
Gambar 4 Grafik Kinerja Model

Dari semua data tersebut maka didapatkan Grafik Kinerja Model Gambar tersebut menunjukkan empat grafik yang digunakan untuk mengevaluasi kinerja model dalam pembelajaran mendalam. Grafik pertama, *Loss over Epochs*, menggambarkan nilai kerugian (*loss*) untuk data pelatihan, dengan tiga batch data yang dibandingkan. Grafik kedua, *Accuracy over Epochs*, menunjukkan akurasi pelatihan untuk tiga batch data yang berbeda. Grafik ketiga, *Validation Loss over Epochs*, memplot nilai kerugian pada data validasi untuk memantau overfitting, juga membandingkan tiga batch data. Grafik keempat, *Validation Accuracy over Epochs*, menunjukkan akurasi validasi untuk ketiga batch. Secara keseluruhan, grafik ini membantu memahami bagaimana model belajar dan

kinerjanya terhadap data pelatihan dan validasi, yang penting untuk mengevaluasi dan menyetel hyperparameters model.

Maka dapat disimpulkan pelatihan selama 50 epoch dengan batch size 128 memberikan hasil terbaik dalam hal akurasi validasi (0.8750) dan stabilitas loss, dibandingkan dengan pelatihan selama 100 atau 150 epoch yang cenderung menunjukkan overfitting dan penurunan akurasi validasi.

3.2 Grafik



Gambar 5 Grafik *Training and Validation*

Grafik pada gambar 5 menunjukkan kinerja model selama pelatihan. *Training Loss* yang menurun dan *Training Accuracy* yang meningkat pada garis merah, menunjukkan bahwa model semakin baik dalam memprediksi data pelatihan. *Validation Loss* yang relatif stabil dengan sedikit fluktuasi pada garis hijau dan *Validation Accuracy* yang meningkat namun menunjukkan kemampuan model dalam memprediksi data yang tidak terlihat selama pelatihan. *Best epoch* yang ditandai dengan titik biru pada kedua grafik membantu mengidentifikasi titik dimana model mencapai kinerja terbaik pada data validasi. Ini menunjukkan bahwa model belajar dengan baik pada data pelatihan, namun perlu dioptimalkan lebih lanjut untuk meningkatkan stabilitas dan kinerja pada data validasi, untuk mencegah *overfitting* dan memastikan generalisasi yang baik.

IV. KESIMPULAN

Hasil implementasi pengenalan penyakit cabai dengan metode Deep Learning menggunakan Convolutional Neural Network (CNN) menunjukkan bahwa arsitektur CNN dapat mengklasifikasikan penyakit Antraks pada cabai dengan memberikan label pada data yang telah diinput. Berdasarkan grafik *Training and Validation Loss* serta *Training and Validation Accuracy*, model mencapai akurasi terbaik pada data validasi sebesar 90% di epoch ke-23. Meskipun terdapat fluktuasi pada *Validation Accuracy*, penurunan *Validation Loss* menunjukkan model dapat mengenali pola dengan baik meski perlu dioptimalkan lebih lanjut untuk meningkatkan kestabilan dan kinerja pada data validasi.

DAFTAR PUSTAKA

- [1] Z. A. Noli dan H. V. Labukti, "Pengaruh Ekstrak Paku Resam (*Gleichenia linearis*) sebagai Biostimulan terhadap Pertumbuhan dan Hasil Cabai Keriting (*Capsicum annum L.*)

- Kultivar Kopay,” *Agro Bali Agric. J.*, vol. 5, no. 3, hal. 492–497, 2022, doi: 10.37637/ab.v5i3.999.
- [2] N. M. Yasen, S. Rifka, R. Vitria, dan Y. Yulindon, “Pemanfaatan Yolo Untuk Deteksi Hama Dan Penyakit Pada Daun Cabai Menggunakan Metode Deep Learning,” *Elektron J. Ilm.*, vol. 15, hal. 63–71, 2023, doi: 10.30630/eji.0.0.397.
- [3] D. Damaiyanti, R. Yulianty, A. Marzuki, S. Kasim, dan H. Rante, “ANALISIS RESIDU PESTISIDA KLORPIRIFOS PADA CABAI (*Capsicum sp.*) DARI DESA BUNGIN KECAMATAN BUNGIN KABUPATEN ENREKANG,” *Maj. Farm. dan Farmakol.*, vol. 23, no. 3, hal. 106–108, 2020, doi: 10.20956/mff.v23i3.9401.
- [4] N. Prihatiningsih, H. A. Djatmiko, dan E. Erminawati, “Komponen epidemi penyakit antraknosa pada tanaman cabai di kecamatan baturaden kabupaten Banyumas,” *J. Agro*, vol. 7, no. 2, hal. 203–212, 2020, doi: 10.15575/8000.
- [5] R. Z. Rahman, G. -, dan T. N. Padilah, “Sistem Pakar Hama Dan Penyakit Cabai Berbasis Teorema Bayes (Studi Kasus : Dinas Pertanian Karawang),” *JUTEKIN (Jurnal Tek. Inform.)*, vol. 9, no. 1, 2021, doi: 10.51530/jutekin.v9i1.468.
- [6] I. Jamaledyn, R. El Ayachi, dan M. Biniz, “Automated Arabic News Classification using the Convolutional Neural Network,” *Int. J. Electr. Eng. Informatics*, vol. 15, no. 2, hal. 277–290, 2023, doi: 10.15676/ijeei.2023.15.2.7.
- [7] M. A. Rahman, M. R. Islam, M. A. H. Rafath, dan S. Mhejabin, “CNN Based Covid-19 Detection from Image Processing,” *J. ICT Res. Appl.*, vol. 17, no. 1, hal. 99–113, 2023, doi: 10.5614/itbj.ict.res.appl.2023.17.1.7.
- [8] I. K. G. Darma Putra, I. P. D. Jayantha Putra, R. Fauzi, dan D. Witarasyah, “Classification of Tomato Plants Diseases Using Convolutional Neural Network,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 10, no. 5, hal. 1821–1827, 2020, doi: 10.18517/ijaseit.10.5.11665.
- [9] D. Avianto dan I. E. Handayani, “Klasifikasi Penyakit Antraknosa Pada Cabai Merah Teropong ”Inko Hot” Dengan Metode Convolutional Neural Network,” *SINTECH (Science Inf. Technol. J.)*, vol. 6, no. 2, hal. 76–88, 2023, doi: 10.31598/sintechjournal.v6i2.1377.
- [10] R. Rosalina dan A. Wijaya, “Pendeteksian Penyakit pada Daun Cabai dengan Menggunakan Metode Deep Learning,” *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 3, hal. 452–461, 2020, doi: 10.28932/jutisi.v6i3.2857.
- [11] Z. E. Fitri, U. Nuhanatika, A. Madjid, dan A. M. N. Imron, “Penentuan Tingkat Kematangan Cabe Rawit (*Capsicum frutescens L.*) Berdasarkan Gray Level Co-Occurrence Matrix,” *J. Teknol. Inf. dan Terap.*, vol. 7, no. 1, hal. 1–5, 2020, doi: 10.25047/jtit.v7i1.121.
- [12] D. S. Anggraeni, A. Widayana, P. D. Rahayu, dan C. Rozikin, “Metode Algoritma Convolutional Neural Network pada Klasifikasi Penyakit Tanaman Cabai,” *STRING (Satuan Tulisan Ris. dan Inov. Teknol.)*, vol. 7, no. 1, hal. 73, 2022, doi: 10.30998/string.v7i1.13304.
- [13] K. N. Sami, Z. M. A. Amin, dan R. Hassan, “Waste Management Using Machine Learning and Deep Learning Algorithms,” *Int. J. Perceptive Cogn. Comput.*, vol. 6, no. 2, hal. 97–106, 2020, doi: 10.31436/ijpcc.v6i2.165.
- [14] N. D. Miranda, L. Novamizanti, dan S. Rizal, “Convolutional Neural Network Pada Klasifikasi Sidik Jari Menggunakan Resnet-50,” *J. Tek. Inform.*, vol. 1, no. 2, hal. 61–68, 2020, doi: 10.20884/1.jutif.2020.1.2.18.
- [15] F. Ertam, “Data classification with deep learning using tensorflow,” *2nd Int. Conf. Comput. Sci. Eng. UBMK 2017*, hal. 755–758, 2017, doi: 10.1109/UBMK.2017.8093521.