



Implementasi CNN untuk Identifikasi Penyakit Daun Cabai

Alamsyah Aqmal Salya Nur¹, Lesmana Iwan², Priantama Rio³

Email: ¹aqmalimam@gmail.com, ²iwanlesmana@gmail.com, ³riopriantama@gmail.com

¹Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Kuningan

²Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Kuningan

³Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Kuningan

Diterima: 11 Juni 2025 | Direvisi: 23 Agustus 2025 | Disetujui: 24 Desember 2025

©2020 Program Studi Teknik Informatika Fakultas Ilmu Komputer,
Universitas Muhammadiyah Riau, Indonesia

Abstrak

Deteksi penyakit pada tanaman cabai merupakan langkah penting dalam mencegah kerusakan yang dapat mengurangi produktivitas dan menyebabkan kerugian ekonomi bagi petani. Penelitian ini merancang aplikasi *Chili Leaf Disease App* berbasis Android yang dapat mendeteksi penyakit daun cabai secara otomatis. Aplikasi ini menggunakan algoritma *Convolutional Neural Network (CNN)* dengan arsitektur *MobileNetV2* untuk melakukan klasifikasi penyakit melalui gambar daun yang diambil langsung dari kamera atau diunggah dari galeri. *Dataset* yang digunakan terdiri dari 4000 gambar daun cabai dengan empat kelas penyakit. Hasil pengujian menunjukkan bahwa model mampu mencapai akurasi sebesar 97,5%. Pengembangan sistem dilakukan menggunakan metode *Rapid Application Development (RAD)*, yang dipilih karena siklus pengembangannya lebih singkat, fleksibel, dan meningkatkan keterlibatan pengguna. Pendekatan ini memungkinkan pengembangan aplikasi yang efisien, cepat, dan responsif terhadap kebutuhan pengguna. Aplikasi ini diharapkan dapat membantu petani mendeteksi penyakit secara dini dan mengambil tindakan preventif lebih cepat guna menjaga kesehatan tanaman.

Kata kunci: Deteksi Penyakit Cabai, CNN, RAD, Android

Implementation of CNN for Chili Leaf Disease Identification

Abstract

Disease detection in chili plants is a crucial step in preventing damage that can reduce productivity and cause economic losses for farmers. This study presents the design of an Android-based application called Chili Leaf Disease App that can automatically detect chili leaf diseases. The application uses a Convolutional Neural Network (CNN) algorithm with the MobileNetV2 architecture to classify leaf diseases through images captured directly from the camera or uploaded from the gallery. The dataset used consists of 4,000 chili leaf images across four disease classes. Testing results show that the model achieves an accuracy of 97.5%. The system was developed using the Rapid Application Development (RAD) method, chosen for its shorter development cycle, flexibility, and ability to increase user involvement. This approach enables efficient, fast, and user-responsive application development. The application is expected to help farmers detect diseases early and take preventive action more quickly to maintain plant health.

Keywords: *Chili Disease Detection, CNN, RAD, Android*

1. PENDAHULUAN

Tanaman cabai juga memerlukan sinar matahari yang cukup, karena selain digunakan untuk fotosintesis juga berperan dalam mengurangi hama. Deteksi penyakit tanaman adalah salah satu aspek paling penting dalam merawat tanaman [1]. Penyakit yang

tidak terdeteksi dan dibiarkan dapat menyebabkan kerusakan tanaman secara serius, yang pada akhirnya berdampak pada kerugian ekonomi bagi para petani. Beberapa penelitian telah dilakukan untuk memahami berbagai penyakit tanaman, sehingga perawatan menjadi lebih mudah dengan mengetahui tanda-tanda dari setiap penyakit, cara mengobati tanaman, dan cara mencegah penyakit [2].

Beberapa penelitian yang dilakukan oleh peneliti sebelumnya, yaitu penelitian oleh Ully Nuhanatika berjudul “Penentuan Tingkat Kematangan Cabe Rawit (*Capsicum frutescens* L.) Berdasarkan *Gray Level Co-Occurrence Matrix*” bertujuan untuk mengklasifikasikan tingkat kematangan cabai rawit dengan menggunakan ekstraksi fitur warna dan tekstur. Ekstraksi fitur warna diambil dari rata-rata nilai saturasi, sementara ekstraksi fitur tekstur menggunakan nilai fitur *Gray Level Co-Occurrence Matrix* (GLCM). Hasilnya adalah sistem pelatihan yang mampu mengklasifikasikan tingkat kematangan cabai rawit dengan akurasi sebesar 81,4% dan akurasi pengujian cabai rawit sebesar 74,2% [3]. Selanjutnya, penelitian oleh Joelyan Vicky, Frisca Ayu, dan Bagas Julianto berjudul “Implementasi Pendeteksi Penyakit pada Daun Alpukat Menggunakan Metode CNN” di mana proses implementasi dengan model dan data uji pada daun alpukat yang terinfeksi penyakit menunjukkan hasil yang baik, dengan menggunakan metode CNN dapat mendeteksi penyakit pada daun alpukat dengan akurasi 80% [4].

Sistem deteksi ini sangat direkomendasikan karena hasil akurasinya cukup baik. Banyak penelitian yang menggunakan metode *Deep Learning* sebagai dasar untuk membuat sistem. Penggunaan metode deteksi secara langsung dalam pembelajaran saat ini memiliki batasan, seperti waktu yang lama dan akurasi yang kurang memadai, terutama saat gejala penyakit sulit dibedakan. Untuk mengatasi batasan tersebut, pemanfaatan teknologi seperti kecerdasan buatan (AI) dan pembelajaran mesin (*Machine Learning*) dapat membantu dalam mendeteksi penyakit dengan lebih cepat dan akurat. Algoritma *Convolutional Neural Network* (CNN) telah terbukti unggul dalam bidang pengenalan pola dan pengolahan citra digital, sehingga cocok digunakan untuk mendeteksi penyakit tanaman secara otomatis berdasarkan gambar daun yang terinfeksi. CNN dapat diimplementasikan dalam aplikasi berbasis Android untuk memberikan solusi praktis dalam mendeteksi penyakit daun tanaman secara otomatis [5]. Penggunaan aplikasi ini juga relevan untuk mendukung pembelajaran di lingkungan pendidikan seperti SMK.

Pengembangan aplikasi ini diharapkan dapat meningkatkan efisiensi siswa dan guru melalui deteksi awal, serta menjadi alat belajar yang berguna bagi siswa di SMK Negeri 1 Kuningan. Dengan demikian, penelitian ini akan berfokus pada pembuatan aplikasi deteksi penyakit pada daun tanaman cabai menggunakan algoritma CNN, yang diharapkan dapat membantu siswa dan guru dalam menemukan masalah penyakit khususnya pada tanaman cabai dengan lebih cepat dan tepat.

2. METODE PENELITIAN

2.1. Metode Pengumpulan Data

Metode yang digunakan untuk mengumpulkan data adalah pembahasan terkait cara memperoleh data yang dibutuhkan untuk penelitian yang akan dilaksanakan. Beberapa metode yang digunakan dalam proses penelitian ini adalah sebagai berikut:

- 1) Observasi: Metode untuk mengumpulkan data melalui pengamatan di SMK Negeri 1 Kuningan untuk memahami bagaimana siswa mengenali penyakit pada tanaman cabai, serta mengamati kondisi pertanian cabai untuk mendapatkan informasi yang akurat.
- 2) Wawancara : Pengumpulan data dilakukan melalui tanya jawab kepada Bapak Andry Riyandi, S.Pd selaku Guru Jurusan Agribisnis Tanaman Pangan dan Hortikultura (ATPH) di SMK Negeri 1 Kuningan, guna mendapatkan wawasan lebih dalam mengenai tantangan yang dihadapi oleh siswa dalam proses mengenali penyakit tanaman cabai.
- 3) Studi Pustaka: Metode untuk mengumpulkan data dengan menggunakan literatur dilakukan seiring memperoleh informasi dari berbagai literatur seperti jurnal, artikel ilmiah, dan buku yang berhubungan dengan aplikasi deteksi penyakit, metode *Convolutional Neural Network* (CNN), dan permasalahan yang diteliti. Sumber-sumber ini digunakan untuk melengkapi informasi yang diperlukan dalam mendukung pelaksanaan penelitian ini.

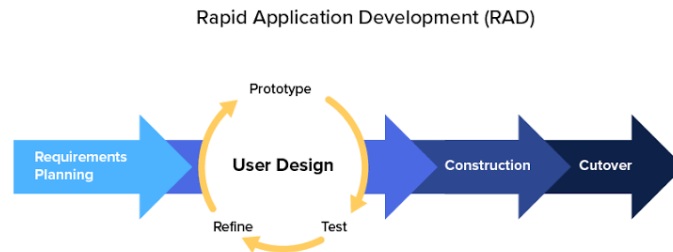
2.2. Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian Aplikasi Deteksi Penyakit Tanaman Cabai ini adalah *Rapid Application Development* (RAD), salah satu model dari *System Development Life Cycle* (SDLC). RAD merupakan model pengembangan perangkat lunak secara linear *sequential* dengan siklus pengembangan yang singkat [6]. Metode ini dipilih karena memiliki kelebihan seperti: siklus pengembangan lebih pendek, fleksibel dibanding *Waterfall*, meningkatkan keterlibatan pengguna, serta menekan kemungkinan kesalahan.

Tahapan RAD yang digunakan meliputi:

- 1) Rencana Kebutuhan (*Requirements Planning*): Karena ini merupakan langkah awal keberhasilan pembuatan sistem dan dapat mencegah kesalahan komunikasi antara pengguna dan penulis, pengguna dan penulis bertemu untuk mempelajari dan memecahkan masalah yang muncul.
- 2) Desain Pengguna (*User Design*): Tahap pembuatan rancangan yang akan diusulkan untuk memenuhi dengan kebutuhan dan permasalahan yang dialami yakni Rancang Bangun Aplikasi Deteksi Penyakit Tanaman Cabai Menggunakan Algoritma

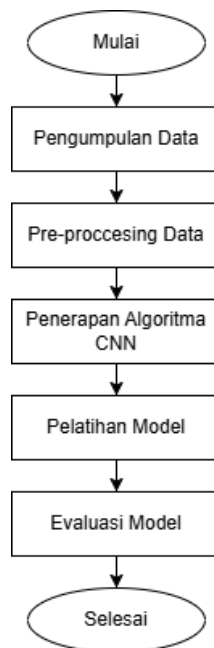
- Convolutional Neural Network (CNN) di SMK Negeri 1 Kuningan, agar sesuai dengan rencana dan diharapkan dapat mengatasi masalah. Penelitian ini menggambarkan desain sistem menggunakan *Tools Unified Modeling Language* (UML).
- 3) Pengembangan (*Construction*): Tahap ini adalah tahap memulai membuat sistem yang sudah direncanakan. Memulai menyusun suatu kode program atau biasa disebut *coding*, untuk merubah desain sistem yang telah dibuat menjadi sebuah aplikasi yang telah direncanakan agar dapat digunakan.
 - 4) Pengujian (*Cutover*): Pengujian menyeluruh terhadap sistem dan komponennya sebelum digunakan.



Gambar 1 Metode *Rapid Application Development* [6]

2.3. Metode Penyelesain Masalah

Analisis penyelesaian masalah yang menjelaskan bagaimana peneliti melakukan semua tahapan proses, mulai dari pengumpulan data, *preprocessing* data, perancangan algoritma CNN, pelatihan model, evaluasi model, hingga menghasilkan sebuah *output*.



Gambar 2 Alur Penyelesain Masalah

Berdasarkan Gambar 2, alur aplikasi yang akan dibuat meliputi:

1. Pengumpulan Data

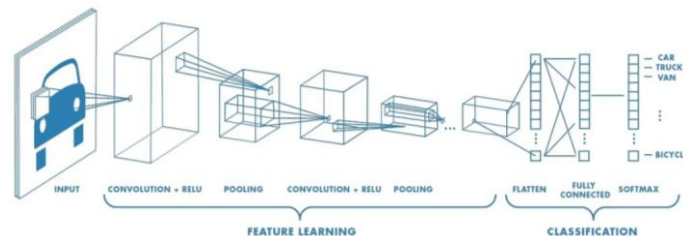
Penelitian ini memulai dengan pengumpulan *dataset*. Data penelitian ini berasal dari kumpulan gambar yang diambil secara manual menggunakan smartphone serta dari sumber Kaggle (<https://www.kaggle.com/datasets/nasar43/chilli>) yang bersumber asli dari pemilik [7], untuk mengumpulkan 4000 *dataset*, masing-masing dengan 1000 data untuk setiap penyakit daun pada tanaman cabai dan 1000 daun sehat. *Dataset* ini terdiri dari gambar Keriting Daun, Virus Kuning, Bercak Daun, dan Daun Sehat.

2. *Preprocessing* Data

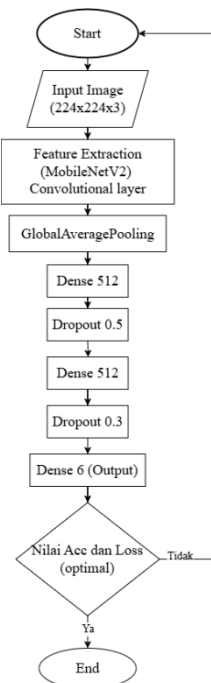
Preprocessing data, tahap penting dalam analisis prediktif, bertujuan untuk meningkatkan kualitas dan kesiapan data sebelum digunakan dalam pelatihan model. [8]. Data yang telah terkumpul akan diolah melalui augmentasi, termasuk *rotasi*, *flipping*, dan *rescale*, agar model dapat lebih variatif dalam mengenali pola penyakit. Selanjutnya, gambar akan diubah menjadi ukuran yang sesuai agar mudah dipelajari oleh model.

3. Penerapan Algoritma CNN

Convolutional Neural Network terdiri dari beberapa tahap pengolahan. Lapisan konvolusi akan mengekstrak fitur dari data input, dan lapisan pooling adalah tugas utamanya untuk mengurangi dimensi data input [9]. CNN sangat cocok untuk pengolahan gambar digital karena input utamanya adalah foto. Untuk mengidentifikasi dan membedakan objek yang terlihat dalam gambar, CNN adalah salah satu teknik *deep learning* yang paling efektif [10]. Arsitektur CNN terdiri dari beberapa lapisan khusus yang mencerminkan keberhasilan gagasan ini. Arsitektur *convolutional neural network* diperlihatkan pada Gambar 3.



Gambar 3 Arsitektur Dasar CNN [11]



Gambar 4 Flowchart Algoritma CNN [12]

Berikut merupakan penjelasan dari setiap tahapan pada *flowchart* algoritma *Convolutional Neural Network*:

Arsitektur model dimulai dengan proses awal menggunakan algoritma *Convolutional Neural Network* (CNN) untuk melakukan analisis terhadap citra input. Gambar masukan memiliki ukuran 224×224 piksel dengan tiga channel warna (RGB). Selanjutnya, fitur penting dari citra diekstraksi menggunakan arsitektur *MobileNetV2* yang berfungsi sebagai *backbone*. Setelah proses ekstraksi fitur, dilakukan reduksi dimensi menggunakan *Global Average Pooling*, yang mengubah feature map menjadi vektor representasi. Vektor ini kemudian diproses oleh dense layer pertama yang terdiri atas 512 *neuron* dengan fungsi aktivasi ReLU. Untuk mengurangi risiko *overfitting*, ditambahkan lapisan dropout sebesar 0,5. Setelah itu, fitur diperdalam melalui *dense layer* kedua dengan jumlah *neuron* yang sama, yaitu 512, diikuti oleh *dropout* kedua sebesar 0,3. Pada tahap akhir, terdapat *dense layer output* dengan 6 *neuron*, yang digunakan untuk memetakan ke empat kelas penyakit daun cabai yang telah ditentukan. Selama pelatihan, model terus dievaluasi berdasarkan nilai akurasi dan *loss*. Jika nilai tersebut belum mencapai kondisi optimal, proses pelatihan dilanjutkan hingga hasil yang diinginkan tercapai. Proses pelatihan dinyatakan selesai jika model telah menunjukkan performa optimal berdasarkan metrik evaluasi tersebut.

4. Pelatihan Model

Setelah data *training* selesai di implementasi, maka data *testing* berupa gambar akan di panggil untuk menguji coba hasil *training*. Hasil akhir dari testing adalah berupa prediksi gambar sesuai dengan jenis obyek pada gambar. Hasil keluaran testing akan dibandingkan kembali dengan hasil sebenarnya untuk mengukur seberapa besar tingkat akurasi fitur prediksi.

5. Evaluasi Model

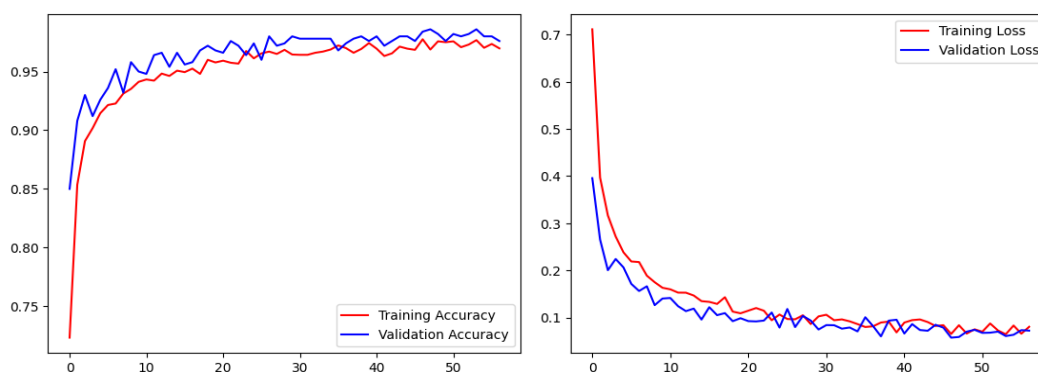
Model akan dievaluasi berdasarkan akurasi dan *loss* yang dicapai setelah proses pelatihan. Grafik *loss* dan *accuracy* digunakan untuk memantau kinerja model dan menghindari *underfitting* atau *overfitting*. Jika hasil evaluasi menunjukkan akurasi tinggi dan nilai *loss* rendah, maka model siap digunakan.

3. HASIL DAN PEMBAHASAN

Pada tahap ini dilakukan implementasi model CNN untuk deteksi penyakit daun cabai, integrasi model ke aplikasi Android, serta perancangan antarmuka agar sistem dapat digunakan secara praktis oleh pengguna.

3.1. Hasil Pelatihan

Setelah pelatihan model selesai, dilakukan evaluasi menggunakan visualisasi grafik untuk menampilkan nilai *loss*, *accuracy*, *val_loss*, dan *val_accuracy*. Nilai *loss* menunjukkan tingkat kesalahan prediksi model, sedangkan *accuracy* menunjukkan tingkat keberhasilan model dalam klasifikasi citra. Visualisasi grafik Gambar 5 digunakan untuk memberikan gambaran performa model secara lebih jelas dan memudahkan analisis perbandingan antara nilai akurasi dan kesalahan.



Gambar 5. Hasil Model Training dan Validation Accuracy-Loss

3.2. Hasil Evaluasi

4. KESIMPULAN

Berdasarkan hasil evaluasi, sistem deteksi penyakit daun cabai yang dikembangkan menggunakan algoritma *Convolutional Neural Network* dengan arsitektur *MobileNetV2* menunjukkan performa yang tinggi dan stabil. Model mampu mendeteksi empat kelas utama penyakit daun cabai seperti Bercak Daun, Daun Sehat, Keriting Daun, dan Virus Kuning dengan nilai *F1-Score* antara 0.975 hingga 0.98. Evaluasi metrik menunjukkan bahwa kelas Bercak Daun dan Keriting Daun memiliki *precision*, *recall*, dan *F1-score* sebesar 98%, sedangkan kelas Daun Sehat dan Virus Kuning masing-masing mencapai *F1-score* sebesar 97.5%, meskipun terdapat sedikit ketidakseimbangan antara *precision* dan *recall*. Hasil ini mencerminkan kemampuan model dalam mendeteksi dan mengklasifikasikan citra daun cabai secara akurat. Meskipun demikian, faktor eksternal seperti pencahayaan, jarak kamera, dan kondisi visual daun tetap menjadi aspek penting yang dapat mempengaruhi akurasi sistem saat digunakan di lingkungan nyata.

DAFTAR PUSTAKA

- [1] L. Kaur and S. G. Sharma, "Identification of plant diseases and distinct approaches for their management," *Bull Natl Res Cent*, vol. 45, no. 1, Dec. 2021, doi: 10.1186/s42269-021-00627-6.
- [2] A. T. R. Dzaky and F. A. M. Wikky, "Deteksi Penyakit Tanaman Cabai Menggunakan Metode Convolutional Neural Network," Apr. 2021.
- [3] U. Nurhanatika, Z. E. Fitri, A. M. N. Imron, and A. Madjid, "Penentuan Tingkat Kematangan Cabe Rawit (*Capsicum frutescens* L.) Berdasarkan Gray Level Co-Occurrence Matrix," *Jurnal Teknologi Informasi dan Terapan (J-TIT)*, vol. 7, no. 1, pp. 2580–2291, 2020.
- [4] J. Vicky, F. Ayu, and B. Julianto, "Implementasi Pendeteksi Penyakit pada Daun Alpukat Menggunakan Metode CNN," vol. 2, 2023.
- [5] K. Pendke, N. Pawar, A. Pawar, S. Gawande, P. Chauhan, and D. Balvi, "Identification of Plant Leaf Disease and Provision of Treatment using CNN," *International Journal of Innovations in Engineering and Science*, vol. 7, no. 9, pp. 44–47, Aug. 2022, doi: 10.46335/ijies.2022.7.9.9.
- [6] N. Hidayat and K. Hati, "Penerapan Metode Rapid Application Development (RAD) dalam Rancang Bangun Sistem Informasi Rapor Online (SIRALINE)," *Jurnal Sistem Informasi*, vol. 10, no. 1, pp. 8–17, Feb. 2021, doi: 10.51998/jsi.v10i1.352.
- [7] P. R. Aishwarya M P, "Chilli dataset." Accessed: May 13, 2025. [Online]. Available: <https://data.mendeley.com/datasets/tf9dtfz9m6/1>
- [8] A. Alsyar, R. Amin Putra, W. A. Ramadhani, F. R. Hidayatullah, and E. Ismanto, "Pemodelan Prediktif Diabetes Menggunakan Pendekatan Multimodel Machine Learning dan Deep Learning Predictive Modeling of Diabetes Using Multimodel Machine learning and Deep learning Approaches," vol. 6, no. 2, pp. 158–165, 2025, doi: 10.37859/coscitech.v6i2.9812.
- [9] S. SATRIA, Sumijan, and Billy Hendrik, "Implementasi Convolutional Neural Network Untuk Klasifikasi Citra KTP-El," *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 5, no. 1, pp. 169–176, May 2024, doi: 10.37859/coscitech.v5i1.6708.
- [10] S. M. Zainorzuli, S. A. Che Abdullah, H. Zainol Abidin, and F. Ahmat Ruslan, "Comparison Study on Convolution Neural Network (CNN) Techniques for Image Classification," *Journal of Electrical & Electronic Systems Research*, vol. 20, no. APR2022, pp. 11–17, Apr. 2022, doi: 10.24191/jeesr.v20i1.002.
- [11] Anhar and R. A. Putra, "Perancangan dan Implementasi Self-Checkout System pada Toko Ritel menggunakan Convolutional Neural Network (CNN)," *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 11, no. 2, p. 466, Apr. 2023, doi: 10.26760/elkomika.v11i2.466.
- [12] M. Sholawati, K. Auliasari, and F. X. Ariwibisono, "PENGEMBANGAN APLIKASI PENGENALAN BAHASA ISYARAT ABJAD SIBI MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)," 2022.