

APLIKASI KLASIFIKASI PENYAKIT KENTANG MENGGUNAKAN ALGORITMA CONVOLUTION NEURAL NETWORK (CNN)

I Putu Astya Prayudha¹, Gde Brahupadhya Subiksa², Putu Satya Saputra³, I Nyoman Rai Widartha Kesuma⁴, Vianne Clarinta Putri Gurning⁵

Program Studi Teknologi Rekayasa Perangkat Lunak^{1,2,4,5}, Jurusan Teknologi Informasi^{1,2,4,5},

Program Studi D3 Akuntansi³, Jurusan Akuntansi³,

Politeknik Negeri Bali

Kampus Politeknik Negeri Bali, Bukit Jimbaran, Kuta Selatan, Badung, Bali

astyaprayudha@pnb.ac.id¹, brahupadhya@pnb.ac.id², satya@pnb.ac.id³, raiwidartha@pnb.ac.id⁴,
vianneclarinta.7@gmail.com⁵

Abstrak

Kentang merupakan komoditas hortikultura penting yang rentan terhadap berbagai penyakit seperti *Black Scurf*, *Blackleg*, *Common Scab*, *Dry Rot*, dan *Pink Rot*. Deteksi dini penyakit menjadi krusial untuk mengurangi kerugian dan meningkatkan produktivitas. Penelitian ini bertujuan mengembangkan model klasifikasi penyakit kentang berbasis citra menggunakan metode *Convolutional Neural Network* (CNN). Dataset yang digunakan terdiri dari 1760 citra yang terbagi ke dalam delapan kelas, dengan skema pembagian data 70% data latih, 20% validasi, dan 10% data uji. Proses pengolahan data meliputi preprocessing dan augmentasi untuk meningkatkan generalisasi model. Arsitektur CNN yang digunakan terdiri dari empat lapisan konvolusi dengan kombinasi *batch normalization*, *max pooling*, *dropout*, dan *fully connected layer*. Model dilatih menggunakan optimizer Adam dengan learning rate 0.0003 dan batch size 32. Hasil pengujian menunjukkan bahwa model mencapai akurasi sebesar 86%, dengan nilai rata-rata *precision* 0.87, *recall* 0.86, dan *F1-score* 0.86. Hasil evaluasi menunjukkan bahwa model mampu mengenali sebagian besar kelas dengan baik, meskipun masih terdapat kesalahan pada kelas dengan kemiripan visual tinggi. Penelitian ini menunjukkan bahwa metode CNN efektif dalam klasifikasi penyakit kentang berbasis citra digital dan berpotensi untuk dikembangkan lebih lanjut pada skala dataset yang lebih besar.

Kata Kunci: Klasifikasi, Penyakit Kentang, *Convolutional Neural Network*, Citra Digital

Abstract

Potatoes are an important horticultural commodity that is susceptible to various diseases such as black scurf, blackleg, common scab, dry rot, and pink rot. Early disease detection is crucial for reducing losses and increasing productivity. This study aims to develop an image-based potato disease classification model using the *Convolutional Neural Network* method. The dataset consists of 1,760 images divided into eight classes, with a data split of 70% training data, 20% validation data, and 10% test data. The data processing involves preprocessing and augmentation to improve the model's generalisation. The convolutional neural network architecture used consists of four convolutional layers combined with batch normalisation, max pooling, dropout, and fully connected layers. The model was trained using the Adam optimiser with a learning rate of 0.0003 and a batch size of 32. Testing results show that the model achieved an accuracy of 86%, with an average precision of 0.87, a recall of 0.86, and an *F1-score* of 0.86. The evaluation results indicate that the model is capable of recognising most classes well, although errors still occur in those with high visual similarity. This study demonstrates that the CNN method is effective for classifying potato diseases based on digital images and has the potential for further development on larger datasets.

Keyword: Classification, Potato Disease, *Convolutional Neural Network*, Digital Image.

PENDAHULUAN

Berbagai penyakit pada kentang masih menjadi faktor utama penurunan produktivitas kentang di Indonesia. Kentang sendiri merupakan komoditas hortikultura strategis, publikasi resmi BPS dan Kementerian Pertanian menunjukkan produksi kentang nasional berada pada kisaran >1 juta ton per tahun dalam kurun 2021–2023, menegaskan pentingnya pengelolaan penyakit untuk menjaga pasokan dan stabilitas agribisnis kentang domestik (Wahyuningsih, 2022).

Deteksi dini berbasis citra menawarkan alternatif yang lebih cepat dan konsisten dibanding observasi manual yang rentan subjektivitas. Dalam lima tahun terakhir, penelitian di Indonesia banyak mengeksplorasi *Convolutional Neural Network* (CNN) untuk pengenalan penyakit tanaman karena kemampuannya mengekstraksi fitur visual dan mengenali pola kompleks secara *end-to-end*. Studi-studi mutakhir menunjukkan CNN mampu mencapai akurasi tinggi pada klasifikasi penyakit tanaman (termasuk komoditas pangan), sehingga layak diadopsi pada skenario lapangan (Rachman et al., 2024). Di sisi implementasi, platform aplikasi web menjadi kunci agar model dapat digunakan praktisi pertanian secara luas tanpa perlu instalasi rumit. Framework Flask populer karena ringan, mudah diintegrasikan dengan model TensorFlow (Fathurrahman & Akbar, 2024), serta mendukung penyajian antarmuka prediksi berbasis web (Wijaya Kusuma et al., 2023). Riset pengembangan sistem deteksi penyakit tanaman berbasis CNN yang diintegrasikan ke aplikasi web (termasuk yang memanfaatkan Flask) di Indonesia menunjukkan kelayakan teknis dan kemudahan adopsi pengguna akhir (Mursyid et al., 2025).

Namun demikian, tantangan praktis tetap ada seperti keberagaman kondisi pencahayaan, sudut pengambilan gambar, latar belakang yang bising, hingga keterbatasan ukuran dataset lokal dapat menurunkan generalisasi model (Anhar & Putra, 2023). Pendekatan yang umum digunakan untuk mengatasinya meliputi augmentasi data, pemilihan arsitektur CNN yang tepat. (Rachman et al., 2024). Maka dari itu peneliti mencoba untuk meneliti dan mengembangkan klasifikasi penyakit kentang dengan menggunakan CNN berbasis Web Flask.

PENELITIAN RELEVAN

Penelitian terkait penerapan metode CNN telah banyak dilakukan sebelumnya. (Azmi et al., 2023) mengklasifikasikan motif batik tanah liat Sumatera Barat menggunakan CNN dengan 400 dataset (4 kelas). Hasilnya, akurasi mencapai 98,75% pada data latih dan 62,5% pada data uji, menunjukkan CNN efektif namun masih perlu peningkatan pada data uji.

(Dacipta & Putra, 2022) mengembangkan sistem klasifikasi sampah berbasis web menggunakan CNN (VGGNet) dengan 8.371 citra (9 kelas). Model dilatih selama 28 epoch dan menghasilkan akurasi 69,77% (training) dan 64,45% (testing).

(Pratama et al., 2024) meneliti klasifikasi penyakit daun pisang menggunakan CNN dengan transfer learning. Dengan 400 dataset, model mencapai akurasi 92%, namun terbatas oleh jumlah data yang relatif kecil.

(Putra & Prihartono, 2025) mengembangkan klasifikasi penyakit daun cabai menggunakan CNN dengan 222 dataset (2 kelas). Dengan teknik augmentasi dan implementasi berbasis web, model memperoleh akurasi 91%, tetapi masih terbatas pada jumlah data dan kelas.

(Hawari et al., 2022) menggunakan CNN untuk klasifikasi penyakit daun padi dengan 759 dataset. Model menghasilkan akurasi sekitar 85–95%, meskipun jumlah epoch yang digunakan masih rendah.

(Rozaqi et al., 2021) meneliti deteksi penyakit daun kentang menggunakan CNN dengan 1.152 dataset (3 kelas). Hasilnya menunjukkan akurasi tinggi, yaitu 95% (training) dan 94% (validasi), namun jumlah kelas masih terbatas.

Secara umum, metode CNN terbukti efektif dalam klasifikasi citra, tetapi masih memiliki keterbatasan pada jumlah dataset, variasi kelas, dan performa pada data uji.

Tabel 1. Perbandingan Penelitian Sebelumnya

No	Peneliti	Objek	Metode	Dataset	Jumlah Kelas	Akurasi	Kelebihan	Kekurangan
1	Azmi et al. (2023)	Batik tanah liat	CNN	400	4 kelas	62,5%	Baik dalam pengenalan pola	Overfitting
2	Dacipta & Putra (2022)	Sampah	CNN	8.371	9 kelas	64,45%	Dataset besar, multi-kelas	Akurasi rendah
3	Pratama et al. (2024)	Daun pisang	CNN	400	2 kelas	92%	Performa tinggi	Dataset kecil

No	Peneliti	Objek	Metode	Dataset	Jumlah Kelas	Akurasi	Kelebihan	Kekurangan
4	Putra & Prihartono (2025)	Daun cabai	CNN	222	2 kelas	91%	Augmentasi, implementasi web	Kelas terbatas
5	Hawari et al. (2022)	Daun padi	CNN	759	4 kelas	86%	Multi-kelas	Epoch rendah
6	Rozaqi et al. (2021)	Kentang	CNN	1.152	3 kelas	94%	Akurasi tinggi	Kelas terbatas
7	Penelitian ini	Kentang	CNN	1.760	8 kelas	86%	Multi-kelas kompleks, dataset cukup besar, evaluasi lengkap (precision, recall, F1-score), arsitektur CNN optimal (dropout & augmentasi), terdapat UI berupa web yang mudah diakses.	Masih ada kemiripan antar kelas tapi dengan jumlah kecil

Berdasarkan penelitian terdahulu, metode *Convolutional Neural Network* (CNN) terbukti efektif dalam klasifikasi citra, namun masih memiliki beberapa keterbatasan seperti jumlah kelas yang sedikit, dataset yang terbatas, serta evaluasi yang belum komprehensif. Selain itu, pada beberapa penelitian, meskipun dataset cukup besar, akurasi yang dihasilkan masih belum optimal akibat kompleksitas dan kemiripan antar kelas.

Penelitian ini memiliki keunggulan dengan menggunakan delapan kelas penyakit kentang dan dataset yang cukup besar yaitu 1760 citra. Kombinasi jumlah kelas yang banyak dan dataset yang besar mampu menghasilkan akurasi yang baik sebesar 86%, sehingga menunjukkan bahwa CNN dapat dioptimalkan untuk menangani klasifikasi multi-kelas yang kompleks. Selain itu, penelitian ini juga menggunakan augmentasi data dan *dropout* untuk meningkatkan performa model.

Keunggulan lainnya adalah model diimplementasikan dalam aplikasi berbasis web dengan *user interface* yang mudah diakses, sehingga pengguna dapat melakukan klasifikasi secara langsung dengan lebih praktis. Dengan demikian, penelitian ini tidak hanya memberikan kontribusi akademis, tetapi juga memiliki manfaat praktis.

METODE PENELITIAN

Tahapan Penelitian

Penelitian ini melalui beberapa tahapan yang dijelaskan dalam diagram alir berikut.

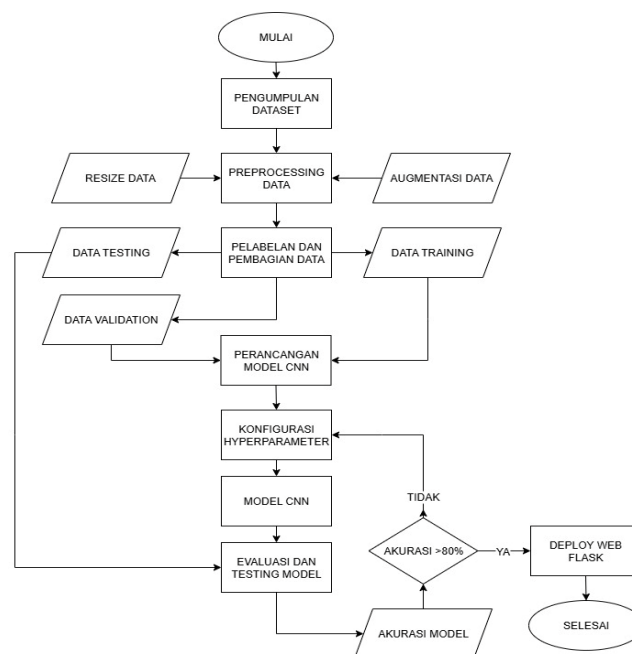


Gambar 1. Diagram Alir Tahapan Penelitian

1. Identifikasi Masalah
Menentukan permasalahan yang akan diteliti, misalnya rendahnya efektivitas deteksi penyakit kentang secara manual.
2. Studi Literatur
Melakukan kajian pustaka dari jurnal, buku, dan penelitian sebelumnya yang relevan untuk memperkuat landasan teori.
3. Perumusan Tujuan dan Hipotesis
Menentukan tujuan penelitian serta dugaan sementara (hipotesis) mengenai solusi yang akan ditawarkan.
4. Pengumpulan Data
Mengumpulkan data yang diperlukan, baik data primer (eksperimen, observasi) maupun sekunder (dataset publik, laporan).
5. Pengolahan dan Analisis Data
Melakukan preprocessing, pengklasifikasian, serta analisis sesuai dengan metode yang dipilih yaitu CNN.
6. Perancangan dan Implementasi Model
Merancang model CNN yang digunakan untuk menyelesaikan permasalahan penelitian.
7. Pengujian dan Evaluasi
Menguji hasil implementasi dengan metode tertentu (misalnya evaluasi akurasi, confusion matrix, atau uji coba sistem).
8. Pembahasan Hasil
Menjelaskan hasil pengujian dan membandingkannya dengan penelitian terdahulu untuk menilai keunggulan serta kelemahan metode.
9. Kesimpulan dan Saran
Menyimpulkan hasil penelitian sesuai tujuan awal serta memberikan saran untuk pengembangan penelitian selanjutnya.

Algoritma CNN

Berikut merupakan tahapan dari diagram alir proses penelitian yang meliputi pengumpulan dataset, proses CNN sampai pada deploy ke web flask.



Gambar 2. Diagram Alir CNN

1. Pengumpulan Dataset

Dataset citra kentang dikumpulkan dari berbagai sumber yaitu dari Keagle dan pengamatan langsung, yang mencakup kelas : Black Scurf, Blackleg, Common Scab, Dry Rot, Healthy Potatoes, Miscellaneous, Not Potatoes, dan Pink Rot dengan jumlah total 1760 gambar kentang.

2. Preprocessing Data

Preprocessing dilakukan melalui beberapa langkah berikut:

Tabel 2. Teknik preprocessing Data

No	Teknik	Deskripsi
1.	Normalisasi Piksel (Rescale)	Nilai piksel citra yang semula berada pada rentang 0–255 diubah menjadi 0–1 dengan pembagian $rescale=1./255$
2.	rotation_range=15	memutar gambar hingga 15°
3.	width_shift_range=0.1 & height_shift_range=0.1	menggeser gambar secara horizontal maupun vertikal sebesar 10%
4.	shear_range=0.1	melakukan distorsi geser pada gambar
5.	zoom_range=0.1	memperbesar gambar secara acak hingga 10%
6.	horizontal_flip=True	membalik gambar secara horizontal
7.	fill_mode='nearest'	mengisi piksel kosong akibat transformasi dengan nilai piksel terdekat

Preprocessing dilakukan melalui normalisasi piksel dengan mengubah rentang nilai 0–255 menjadi 0–1 untuk meningkatkan stabilitas dan mempercepat proses pelatihan. Selain itu, diterapkan data augmentation seperti rotasi, pergeseran, shear, zoom, dan pembalikan horizontal untuk menambah variasi data dan mengurangi overfitting. Penggunaan fill_mode='nearest' memastikan kualitas citra tetap terjaga setelah transformasi. Dengan demikian, model CNN dapat belajar lebih optimal dan memiliki kemampuan generalisasi yang lebih baik.

3. Pelabelan dan Pembagian Data

Dataset diberi label sesuai kelas penyakit lalu dibagi dengan persentase 70% data training, 20% data validation, dan 10% data testing.

4. Perancangan Model CNN

Arsitektur CNN dalam penelitian ini terdiri atas 4 lapisan konvolusi bertingkat dengan jumlah filter yang meningkat (32, 64, 128, 256). Setiap lapisan konvolusi dilengkapi dengan Batch Normalization dan MaxPooling untuk memperbaiki stabilitas serta mereduksi dimensi. Setelah proses ekstraksi fitur, lapisan Flatten digunakan untuk mengubah fitur 2D menjadi vektor 1D yang selanjutnya diproses oleh fully connected layer dengan 512 neuron dan dropout 0.5. Lapisan output menggunakan 8 neuron dengan aktivasi Softmax untuk mengklasifikasikan citra ke dalam delapan kelas penyakit kentang. Konfigurasi

5. Hyperparameter

Proses tuning hyperparameter dilakukan dengan menetapkan optimizer Adam dengan learning rate sebesar 0.0003. Batch size ditetapkan 32 agar seimbang antara akurasi dan efisiensi komputasi, sedangkan jumlah epoch maksimum adalah 50 dengan mekanisme EarlyStopping untuk mencegah overfitting. Selain itu, diterapkan ReduceLROnPlateau untuk menyesuaikan learning rate secara dinamis, Dropout dengan nilai 0.5 sebagai regularisasi, serta Batch Normalization guna mempercepat konvergensi dan menjaga kestabilan model.

6. Model CNN

Proses training dilakukan dengan dataset training sehingga menghasilkan model berupa format .h5.

7. Evaluasi dan Testing Model

Kinerja model diuji dengan dataset validation untuk melihat performa dan menghindari overfitting serta model diuji kembali menggunakan dataset testing untuk memastikan hasil akhir.

8. Akurasi Model

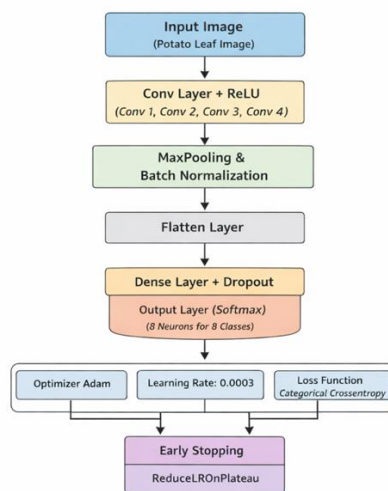
Jika akurasi sudah mencapai target (>80%), maka model dianggap layak digunakan. Jika tidak, perlu dilakukan penyesuaian ulang pada arsitektur atau hyperparameter.

9. Deploy Web Flask

Model terbaik disimpan dalam format .h5 lalu diintegrasikan ke aplikasi berbasis Flask untuk klasifikasi citra kentang melalui web.

HASIL DAN PEMBAHASAN

Proses pengolahan yang terjadi di arsitektur CNN pada penelitian ini dapat dijelaskan pada gambar berikut.



Gambar 3. Proses pada Arsitektur CNN

Arsitektur CNN pada penelitian ini dimulai dari input citra dataset sebanyak 1760 gambar yang terbagi ke dalam 8 kelas. Data kemudian melalui tahap preprocessing berupa normalisasi (rescale 1/255) dan augmentasi (rotasi, pergeseran, zoom, dan flipping) untuk meningkatkan variasi data. Selanjutnya, citra diproses melalui 4 lapisan konvolusi (Conv2D) dengan jumlah filter bertingkat yaitu 32, 64, 128, dan 256, yang masing-masing diikuti oleh Batch Normalization dan MaxPooling untuk menstabilkan pelatihan serta mengurangi dimensi fitur.

Hasil ekstraksi fitur kemudian diubah menjadi vektor satu dimensi menggunakan Flatten layer, lalu diproses oleh Fully Connected layer (Dense) sebanyak 512 neuron dengan fungsi aktivasi ReLU. Untuk mengurangi overfitting, diterapkan Dropout sebesar 0,5. Pada lapisan akhir, digunakan 8 neuron output dengan fungsi aktivasi Softmax untuk menghasilkan klasifikasi sesuai jumlah kelas.

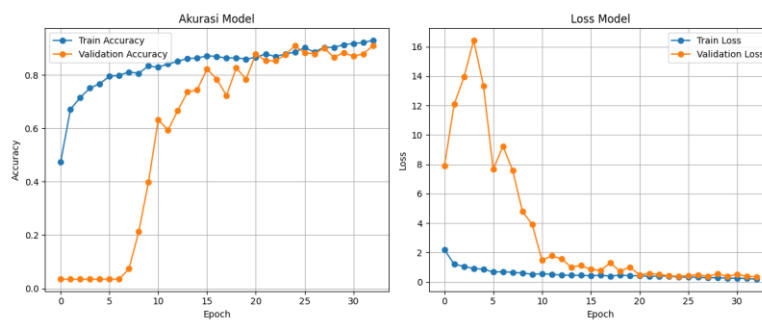
Model dilatih menggunakan optimizer Adam dengan learning rate 0,0003, batch size 32, dan maksimum 50 epoch, serta didukung oleh teknik Early Stopping dan ReduceLROnPlateau untuk meningkatkan performa dan mencegah overfitting.

Berdasarkan perancangan model CNN maka model tersebut dilatih sebanyak 33 epoch dan menghasilkan nilai akurasi yang baik dengan nilai akurasi training sebesar 92% dan nilai akurasi validasi sebesar 90%.

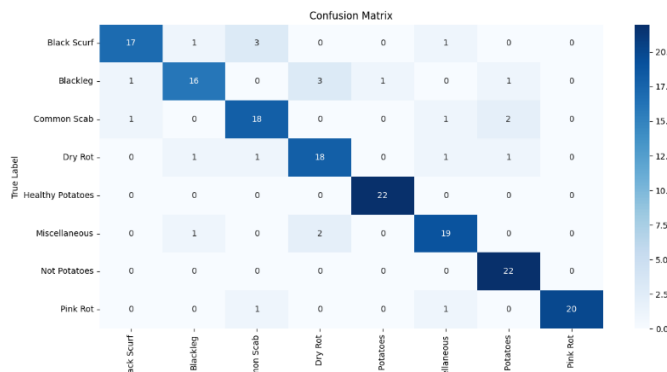
```

Epoch 1/33 - loss: 2.1880 - accuracy: 0.4741 - val_loss: 7.8847 - val_accuracy: 0.0346
Epoch 2/33 - loss: 1.1991 - accuracy: 0.6706 - val_loss: 12.0795 - val_accuracy: 0.0346
Epoch 3/33 - loss: 1.0598 - accuracy: 0.7152 - val_loss: 13.9242 - val_accuracy: 0.0346
Epoch 4/33 - loss: 0.9096 - accuracy: 0.7507 - val_loss: 16.4325 - val_accuracy: 0.0346
Epoch 5/33 - loss: 0.8684 - accuracy: 0.7673 - val_loss: 13.3354 - val_accuracy: 0.0346
Epoch 6/33 - loss: 0.6724 - accuracy: 0.7944 - val_loss: 7.6508 - val_accuracy: 0.0346
Epoch 7/33 - loss: 0.6926 - accuracy: 0.7989 - val_loss: 9.2288 - val_accuracy: 0.0346
Epoch 8/33 - loss: 0.6481 - accuracy: 0.8107 - val_loss: 7.5638 - val_accuracy: 0.0736
Epoch 9/33 - loss: 0.6051 - accuracy: 0.8062 - val_loss: 4.7807 - val_accuracy: 0.2121
Epoch 10/33 - loss: 0.5199 - accuracy: 0.8335 - val_loss: 3.9948 - val_accuracy: 0.3983
Epoch 11/33 - loss: 0.5579 - accuracy: 0.8289 - val_loss: 1.4704 - val_accuracy: 0.6320
Epoch 12/33 - loss: 0.5184 - accuracy: 0.8488 - val_loss: 1.7779 - val_accuracy: 0.5931
Epoch 13/33 - loss: 0.4628 - accuracy: 0.8508 - val_loss: 1.5657 - val_accuracy: 0.6667
Epoch 14/33 - loss: 0.4510 - accuracy: 0.8617 - val_loss: 1.0100 - val_accuracy: 0.7359
Epoch 15/33 - loss: 0.4494 - accuracy: 0.8626 - val_loss: 1.1219 - val_accuracy: 0.7446
Epoch 16/33 - loss: 0.4291 - accuracy: 0.8717 - val_loss: 0.8496 - val_accuracy: 0.8225
Epoch 17/33 - loss: 0.4514 - accuracy: 0.8690 - val_loss: 0.7832 - val_accuracy: 0.7835
Epoch 18/33 - loss: 0.4083 - accuracy: 0.8626 - val_loss: 1.2962 - val_accuracy: 0.7229
Epoch 19/33 - loss: 0.4593 - accuracy: 0.8644 - val_loss: 0.7231 - val_accuracy: 0.8268
Epoch 20/33 - loss: 0.4297 - accuracy: 0.8690 - val_loss: 1.0100 - val_accuracy: 0.7835
Epoch 21/33 - loss: 0.4262 - accuracy: 0.8662 - val_loss: 0.4838 - val_accuracy: 0.8788
Epoch 22/33 - loss: 0.3796 - accuracy: 0.8781 - val_loss: 0.5794 - val_accuracy: 0.8528
Epoch 23/33 - loss: 0.3876 - accuracy: 0.8690 - val_loss: 0.5170 - val_accuracy: 0.8528
Epoch 24/33 - loss: 0.4010 - accuracy: 0.8781 - val_loss: 0.4161 - val_accuracy: 0.8745
Epoch 25/33 - loss: 0.3384 - accuracy: 0.8854 - val_loss: 0.3879 - val_accuracy: 0.9091
Epoch 26/33 - loss: 0.3357 - accuracy: 0.8917 - val_loss: 0.4243 - val_accuracy: 0.8831
Epoch 27/33 - loss: 0.3171 - accuracy: 0.8863 - val_loss: 0.4847 - val_accuracy: 0.8788
Epoch 28/33 - loss: 0.2844 - accuracy: 0.9026 - val_loss: 0.3771 - val_accuracy: 0.9004
Epoch 29/33 - loss: 0.2945 - accuracy: 0.9035 - val_loss: 0.5407 - val_accuracy: 0.8606
Epoch 30/33 - loss: 0.2580 - accuracy: 0.9126 - val_loss: 0.3862 - val_accuracy: 0.8831
Epoch 31/33 - loss: 0.2539 - accuracy: 0.9181 - val_loss: 0.5105 - val_accuracy: 0.8701
Epoch 32/33 - loss: 0.2277 - accuracy: 0.9208 - val_loss: 0.3889 - val_accuracy: 0.8788
Epoch 33/33 - loss: 0.1873 - accuracy: 0.9299 - val_loss: 0.3520 - val_accuracy: 0.9091
    
```

Gambar 4. Hasil Akurasi Model

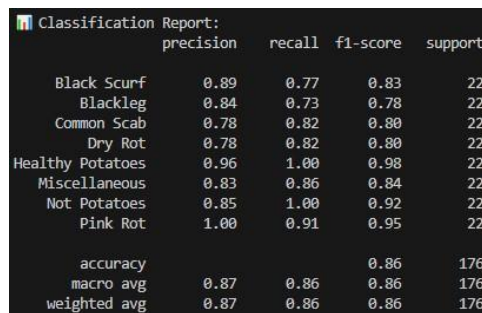


Gambar 5. Visualisasi Akurasi Model



Gambar 6. Pengujian Confusion Matrix

Berdasarkan hasil confusion matrix, model CNN mampu mengklasifikasikan penyakit kentang dengan cukup baik. Kelas Healthy Potatoes dan Not Potatoes berhasil diprediksi dengan akurasi sempurna, sedangkan kelas lain seperti Pink Rot dan Miscellaneous juga menunjukkan hasil yang memuaskan. Meski demikian, masih terdapat kesalahan prediksi pada kelas yang memiliki kemiripan visual, seperti Black Scurf yang terkadang diprediksi sebagai Common Scab serta Blackleg yang sering tertukar dengan Dry Rot. Secara keseluruhan, sebagian besar prediksi berada pada diagonal utama confusion matrix, sehingga dapat disimpulkan bahwa model sudah cukup efektif, meskipun masih diperlukan peningkatan melalui penambahan data dan penggunaan arsitektur yang lebih kompleks.



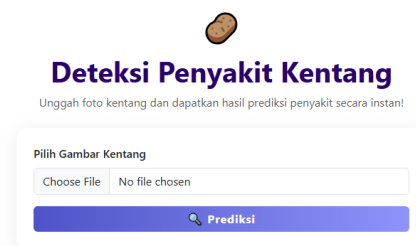
Classification Report:				
	precision	recall	f1-score	support
Black Scurf	0.89	0.77	0.83	22
Blackleg	0.84	0.73	0.78	22
Common Scab	0.78	0.82	0.80	22
Dry Rot	0.78	0.82	0.80	22
Healthy Potatoes	0.96	1.00	0.98	22
Miscellaneous	0.83	0.86	0.84	22
Not Potatoes	0.85	1.00	0.92	22
Pink Rot	1.00	0.91	0.95	22
accuracy			0.86	176
macro avg	0.87	0.86	0.86	176
weighted avg	0.87	0.86	0.86	176

Gambar 7. Hasil Klasifikasi

Berdasarkan classification report, model CNN mencapai akurasi 86% dengan performa yang cukup baik, ditunjukkan oleh nilai precision, recall, dan f1-score rata-rata pada kisaran 0,86–0,87. Kelas *Healthy Potatoes* dan *Not Potatoes* memiliki recall tertinggi (1,00), sementara *Pink Rot* menunjukkan precision sempurna (1,00). Namun, kelas *Black Scurf* dan *Blackleg* masih memiliki kelemahan karena recall yang lebih rendah. Performa ini didukung oleh penggunaan dataset seimbang, data augmentation, serta arsitektur CNN dengan Batch Normalization dan Dropout yang mampu mengurangi overfitting. Meskipun masih terdapat kesalahan pada kelas dengan kemiripan visual tinggi, hasil ini tergolong kompetitif dibanding penelitian sebelumnya dan menunjukkan bahwa CNN efektif untuk klasifikasi penyakit kentang multi-kelas.

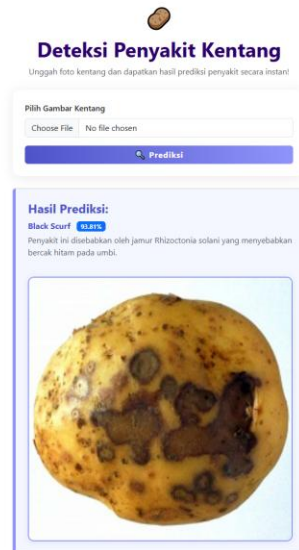
Tampilan Sistem

Sistem yang dibuat sudah dibangun berbasis web flask agar memudahkan pengguna dalam mengakses dan menggunakan fitur dari klasifikasi penyakit kentang. Berikut merupakan tampilan beranda dari sistem yang dibuat.



Gambar 8. Tampilan Beranda

Gambar diatas merupakan tampilan beranda saat sistem diakses melalui web. Berdasarkan tampilan diatas, pengguna hanya perlu memasukkan gambar kentang yang ingin diketahui penyakitnya. Setelah pengguna memilih gambar, pengguna harus menekan *button* prediksi untuk memunculkan hasil prediksinya.



Gambar 9. Hasil Prediksi Penyakit Kentang

Sistem akan menampilkan hasil prediksi berupa presentase keakuratan prediksi beserta penjelasan dari penyakit tersebut. Sistem juga akan menampilkan gambar yang sebelumnya telah dimasukkan agar dapat membandingkan dengan penjelasan dari sistem.

SIMPULAN

Penelitian ini berhasil mengembangkan model klasifikasi penyakit kentang berbasis citra digital menggunakan metode *Convolutional Neural Network* (CNN) yang mampu mengklasifikasikan 8 kelas penyakit dengan kinerja yang cukup baik. Dataset yang digunakan berjumlah 1.760 citra yang dibagi menjadi 70% data training (1.232 citra), 20% data validation (352 citra), dan 10% data testing (176 citra). Hasil pelatihan menunjukkan bahwa model mencapai akurasi training sebesar 92% dan akurasi validasi sebesar 90%, yang mengindikasikan proses pembelajaran berjalan stabil tanpa overfitting yang signifikan.

Pada tahap pengujian menggunakan data testing, model memperoleh akurasi sebesar 86%, dengan nilai rata-rata precision 0.87, recall 0.86, dan F1-score 0.86. Evaluasi menggunakan confusion matrix menunjukkan bahwa sebagian besar prediksi berada pada diagonal utama, yang menandakan model mampu mengklasifikasikan data dengan baik. Namun, masih terdapat kesalahan klasifikasi pada beberapa kelas dengan kemiripan visual tinggi, seperti antar jenis penyakit yang memiliki pola warna dan tekstur yang serupa.

Kontribusi ilmiah dari penelitian ini terletak pada pengembangan model CNN untuk klasifikasi multi-kelas penyakit kentang dengan jumlah kelas yang lebih banyak, serta penggunaan dataset yang relatif besar sehingga mampu meningkatkan kemampuan generalisasi model. Selain itu, penelitian ini juga mengintegrasikan model ke dalam aplikasi berbasis web menggunakan Flask, sehingga memberikan kontribusi praktis berupa sistem yang mudah diakses dan digunakan oleh pengguna untuk melakukan klasifikasi secara real-time.

Untuk pengembangan selanjutnya (*future work*), penelitian ini masih memiliki beberapa keterbatasan yang dapat ditingkatkan, antara lain dengan menambah jumlah dan variasi dataset agar model lebih robust, menerapkan arsitektur CNN yang lebih kompleks atau berbasis transfer learning seperti MobileNet atau ResNet untuk meningkatkan akurasi, serta mengembangkan sistem ke platform mobile atau integrasi dengan teknologi IoT guna mendukung implementasi pada pertanian cerdas (*smart farming*). Dengan pengembangan tersebut, diharapkan sistem dapat memberikan performa yang lebih optimal dan dapat diterapkan secara luas di lingkungan nyata.

DAFTAR PUSTAKA

- Anhar, A., & Putra, R. A. (2023). Perancangan dan implementasi self-checkout system pada toko ritel menggunakan convolutional neural network (CNN). *Elkomika: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 11(2), 466. <https://doi.org/10.26760/elkomika.v11i2.466>
- Azmi, K., Defit, S., & Sumijan, S. (2023). Implementasi convolutional neural network (CNN) untuk klasifikasi batik tanah liat Sumatera Barat. *Jurnal Unitek*, 16(1), 28–40. <https://doi.org/10.52072/unitek.v16i1.504>
- Dacipta, P. N., & Putra, R. E. (2022). Sistem klasifikasi limbah menggunakan metode convolutional neural network (CNN) pada webservice berbasis framework Flask. *Journal of Informatics and Computer Science (JINACS)*, 3(04), 394–402. <https://doi.org/10.26740/jinacs.v3n04.p394-402>
- Fathurrahman, A. A., & Akbar, F. (2024). Perancangan sistem identifikasi jenis sampah menggunakan TensorFlow object detection dan transfer learning. *Jurnal Nasional Teknologi dan Sistem Informasi*, 10(1), 64–71. <https://doi.org/10.25077/teknosi.v10i1.2024.64-71>
- Hawari, F. H., Fadillah, F., Alviandi, M. R., & Arifin, T. (2022). Klasifikasi penyakit padi menggunakan algoritma CNN (convolutional neural network). *Jurnal Responsif*, 4(2), 184–189.
- Mursyid, F., Albarzah, M. A., Irnawati, I., Juliana, W. C., Adiyatma, M. R., & Hamundu, F. M. (2025). Aplikasi deteksi penyakit tanaman cabai menggunakan teknologi machine learning dan Flask. *Jurnal Pustaka Data*, 5(1), 176–182. <https://doi.org/10.55382/jurnalpustakadata.v5i1.1026>
- Pratama, M. D., Gustriansyah, R., & Purnamasari, E. (2024). Klasifikasi penyakit daun pisang menggunakan convolutional neural network (CNN). *Jurnal Teknologi Terpadu*, 10(1), 1–6.
- Putra, A., & Prihartono, W. (2025). Klasifikasi penyakit daun cabai dengan metode CNN untuk deteksi awal. *Jurnal Profesi Insinyur (JPI)*, 6(1).
- Rachman, Yusuf Fadlila, Susanti, P., Putra, A. B. R. P. A., & Rahmawati, N. I. (2024). Sistem informasi deteksi penyakit pada tanaman padi menggunakan metode convolutional neural network (CNN). *Decode: Jurnal Pendidikan Teknologi Informasi*, 4(3), 1193–1204. <https://doi.org/10.51454/decode.v4i3.846>
- Rozaqi, A. J., Sunyoto, A., & Arief, R. (2021). Detection of potato leaves disease using image processing with convolutional neural network methods. *Citec Journal*, 8(1).
- Wahyuningsih, S., & M. (2022). Analisis kinerja perdagangan kentang. *Pusat Data dan Sistem Informasi Pertanian Kementerian Pertanian*, 12, 57.
- Wijaya Kusuma, W., Isnanto, R. R., Fauzi, A., & Korespondensi, P. (2023). DenseNet121 menggunakan kerangka kerja TensorFlow untuk deteksi jenis hewan. *Jurnal Teknik Komputer*, 1(4), 141–147. <https://doi.org/10.14710/jtk.v1i4.37009>