

ANALISIS PERBANDINGAN OPTIMIZER SGD DAN ADAM PADA MODEL CNN UNTUK KLASIFIKASI JAMUR EDIBLE DAN POISONOUS

Yogi Prasetyo, Mohammad Faris Al Fatih, M. Jauharul Iqbal An Zida,

Mohammad Isnain, Agung Mustika Rizki

Informatika, UPN "Veteran" Jawa Timur

Jl.Raya Rungkut Madya, Gunung Anyar, Surabaya

22081010297@student.upnjatim.ac.id

ABSTRAK

Identifikasi jamur secara tradisional menghadapi berbagai tantangan, terutama dalam hal kompleksitas karakteristik morfologi dan kebutuhan akan keahlian khusus yang mendalam. Hal ini dapat meningkatkan risiko kesalahan identifikasi yang berpotensi membahayakan kesehatan manusia, mengingat beberapa jenis jamur bersifat beracun. Untuk mengatasi tantangan tersebut, penelitian ini mengadopsi pendekatan deep learning dengan menggunakan Convolutional Neural Network (CNN) untuk mengklasifikasikan jamur ke dalam kategori edible dan poisonous. Model CNN dikembangkan dan dioptimalkan menggunakan dua algoritma optimasi populer, yaitu Stochastic Gradient Descent (SGD) dan Adaptive Moment Estimation (Adam). Dataset yang digunakan berisi 2.000 gambar jamur dari dua kelas dengan proporsi data pelatihan, validasi, dan pengujian masing-masing sebesar 80%, 10%, dan 10%. Hasil eksperimen menunjukkan bahwa model dengan optimizer SGD mencapai akurasi tertinggi sebesar 95,50% dan F1-Score 0,9550, sedangkan model dengan optimizer Adam menghasilkan akurasi 94,50% dan F1-Score 0,9450. Analisis lebih lanjut menunjukkan bahwa model SGD memiliki performa yang lebih stabil dibandingkan Adam, terutama dalam recall untuk kelas jamur edible sebesar 97%. Dengan demikian, penelitian ini menunjukkan bahwa optimizer SGD lebih unggul untuk implementasi klasifikasi jamur berbasis CNN pada dataset ini. Hasil ini diharapkan dapat memberikan kontribusi bagi pengembangan metode identifikasi jamur yang lebih efisien dan akurat.

Kata kunci : Deep Learning, CNN, SGD, Adam, Klasifikasi Jamur.

1. PENDAHULUAN

Identifikasi spesies jamur memiliki peran yang sangat penting dalam memahami keanekaragaman hayati serta potensi pemanfaatannya di berbagai bidang. Jamur berperan sebagai dekomposer utama dalam ekosistem, membantu mendaur ulang materi organik, menjaga keseimbangan lingkungan, serta mendukung siklus nutrisi di alam [1]-[2]. Selain itu, beberapa jenis jamur memiliki nilai ekonomi yang tinggi karena potensinya sebagai sumber pangan, obat-obatan, serta senyawa bioaktif lainnya yang bermanfaat bagi kesehatan manusia dan industri [1].

Proses identifikasi spesies jamur secara tradisional umumnya dilakukan melalui pengamatan karakteristik makroskopik, seperti bentuk, warna, ukuran, dan tekstur tubuh buah jamur [1]. Namun, pendekatan konvensional ini menghadapi beberapa tantangan yang signifikan. Salah satu permasalahan utama adalah kompleksitas karakter morfologi jamur, di mana banyak spesies memiliki kemiripan yang tinggi, sehingga menyulitkan proses klasifikasi secara akurat [3]. Selain itu, metode ini membutuhkan keahlian khusus dan pengalaman yang mendalam, yang tidak mudah dikuasai oleh peneliti pemula [3]. Oleh karena itu, diperlukan pendekatan yang lebih efisien, akurat, dan mudah diimplementasikan untuk mengatasi keterbatasan metode tradisional.

Dalam beberapa tahun terakhir, kemajuan teknologi di bidang kecerdasan buatan (Artificial Intelligence) telah memberikan solusi inovatif, khususnya dalam klasifikasi gambar menggunakan metode deep learning. Salah satu metode yang paling

banyak digunakan dan terbukti efektif adalah *Convolutional Neural Network* (CNN). CNN mampu mengekstraksi fitur gambar secara otomatis dan mendalam, sehingga memberikan hasil klasifikasi yang lebih akurat dibandingkan metode konvensional. Beberapa penelitian sebelumnya menunjukkan bahwa CNN berhasil mengklasifikasikan spesies jamur dengan performa yang baik. Sebagai contoh, penelitian yang menggunakan 1.200 gambar jamur dari tiga *genus Boletus, Ganoderma, dan Russula* berhasil mencapai akurasi pelatihan sebesar 89% dan akurasi validasi sebesar 82% dengan konfigurasi tertentu seperti batch size 64 dan jumlah epoch tertentu [4].

Meskipun metode CNN menjanjikan hasil yang akurat, performanya sangat dipengaruhi oleh algoritma optimasi yang digunakan dalam proses pelatihan model. Algoritma optimasi bertugas mempercepat konvergensi model menuju nilai minimum loss function, sehingga mempengaruhi akurasi dan stabilitas model. Berbagai algoritma optimasi seperti *Stochastic Gradient Descent* (SGD), *Adaptive Moment Estimation* (ADAM), dan *RMSprop* telah banyak digunakan untuk meningkatkan kinerja CNN. Misalnya, implementasi algoritma *RMSprop* menghasilkan akurasi sebesar 87,99% dengan nilai *presisi* 0,88, *recall* 0,86, dan *F1 score* 0,87, sedangkan penggunaan SGD hanya menghasilkan akurasi sebesar 66,22% dengan performa evaluasi yang lebih rendah [5]. Sementara itu, penelitian lain yang mengkombinasikan optimasi ADAM dalam model hibrida CNN-KNN dengan learning rate 0,001 dan parameter nilai tetangga terdekat 9 berhasil mencapai

akurasi tertinggi sebesar 94,03% dengan nilai *presisi*, *recall*, dan *F1 score* masing-masing sebesar 94% [6].

Berdasarkan tantangan dalam metode identifikasi jamur konvensional dan potensi yang ditawarkan oleh teknologi CNN, penelitian ini bertujuan untuk membandingkan kinerja algoritma optimasi SGD dan ADAM dalam meningkatkan performa CNN untuk klasifikasi jamur. Melalui penelitian ini, diharapkan dapat diperoleh metode identifikasi jamur yang lebih akurat, efisien, dan mudah diimplementasikan, sehingga memberikan kontribusi signifikan dalam bidang taksonomi jamur serta penerapannya di berbagai sektor.

2. TINJAUAN PUSTAKA

2.1. Jamur

Jamur merupakan salah satu keanekaragaman hayati yang banyak tumbuh di Indonesia. Sebagai bagian dari kingdom Fungi, jamur merupakan organisme eukariotik yang memiliki membran inti sel namun tidak mampu melakukan fotosintesis. Jamur juga tidak memiliki akar dan daun sejati, sehingga bergantung pada lingkungan untuk mendapatkan nutrisi. Keunikannya terletak pada variasi ukuran, bentuk, dan warna yang beragam [4].

Dalam ekosistem, jamur memainkan peran penting, baik sebagai pengurai dalam rantai makanan maupun sebagai bahan pangan dan sumber obat-obatan. Namun, penelitian tentang keanekaragaman dan taksonomi jamur makroskopis di Indonesia masih banyak bergantung pada metode identifikasi morfologi tradisional [3].

Salah satu tantangan utama dalam penelitian jamur adalah kurangnya tenaga ahli di bidang ini. Hal ini menyebabkan pengidentifikasian jamur, terutama jamur beracun, menjadi sulit. Namun, jamur beracun dapat dikenali berdasarkan karakteristik seperti ukuran, warna, bentuk tudung, dan batangnya [7].

Pengenalan jamur yang dapat dikonsumsi dan beracun juga dilakukan melalui pengamatan ciri-ciri morfologi. Untuk meningkatkan akurasi dan efisiensi proses identifikasi ini, diperlukan sistem berbasis teknologi yang mampu mengenali jamur berdasarkan ciri morfologi tersebut [8]. Salah satu metode yang menjanjikan dalam hal ini adalah *Convolutional Neural Network* (CNN), yang memiliki kelebihan dalam mengekstraksi detail kecil dari gambar secara otomatis [9].

2.2. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu algoritma dalam bidang Deep Learning yang dikembangkan dari arsitektur *Multilayer Perceptron* (MLP). CNN dirancang khusus untuk mengolah data berbentuk grid, seperti gambar atau suara, sehingga sangat cocok digunakan dalam tugas-tugas klasifikasi gambar, segmentasi, atau pengenalan pola [4].

Salah satu keunggulan utama dari CNN adalah kemampuannya dalam mengklasifikasikan gambar dengan akurasi tinggi. Hal ini dimungkinkan karena CNN dapat mengurangi jumlah parameter yang tidak

relevan dan mampu menangani variasi deformasi pada gambar input, seperti translasi, rotasi, serta perubahan skala. Fitur ini membuat CNN unggul dibandingkan metode tradisional dalam mengolah data gambar [10].

Secara teknis, CNN menerima input berupa gambar mentah dan mentransformasikannya melalui serangkaian lapisan (*layers*) untuk menghasilkan nilai skoring kelas, yang kemudian digunakan untuk melakukan klasifikasi. Proses ini mencakup transformasi piksel gambar asli ke dalam representasi yang lebih abstrak melalui lapisan-lapisan tertentu, seperti *convolutional layer*, *pooling layer*, dan *fully connected layer*. Setiap lapisan ini berkontribusi pada ekstraksi fitur dari gambar secara otomatis [11].

Lapisan-lapisan pada CNN memiliki parameter, seperti bobot (*weights*) dan bias [12], yang dapat dioptimalkan selama pelatihan. Selain itu, CNN memiliki hyperparameters yang mengatur arsitektur jaringan, seperti ukuran filter, jumlah filter, dan ukuran stride. Keunikan CNN terletak pada proses ekstraksi fitur otomatis ini, yang memungkinkan algoritma untuk mengenali pola-pola kompleks dalam gambar tanpa memerlukan desain fitur manual.

2.3. Optimasi SGD dan Adam

Stochastic Gradient Descent (SGD) adalah metode iteratif yang sering digunakan dalam pembelajaran mesin untuk mengoptimalkan penurunan gradien. Algoritma ini membantu model melatih lebih cepat dan mencegah terjebak dalam *local minima* [5].

Adaptive Moment Estimation (Adam) adalah algoritma optimasi yang menggantikan langkah-langkah *Stochastic Gradient Descent* konvensional untuk memperbarui bobot jaringan berdasarkan data latih secara iteratif. Cara kerja Adam menggabungkan aspek terbaik dari dua variasi *Stochastic Gradient Descent*, yaitu *adaptive gradient algorithm* dan *root mean square propagation* [6]. Adam menyesuaikan laju pembelajaran berdasarkan rata-rata gradien pertama (momentum) dan kedua (percepatan) pada setiap parameter.

2.4. Confusion Matrix dan Metrik Evaluasi

Matriks kebingungan (*confusion matrix*) adalah alat yang sangat berguna dalam evaluasi model klasifikasi, yang memberikan gambaran jelas tentang kinerja model dengan membandingkan hasil prediksi dengan label yang sebenarnya [13]. Matriks ini terdiri dari empat komponen utama: True Positives (TP), True Negatives (TN), False Positives (FP), dan False Negatives (FN). Dari komponen ini, kita dapat menghitung berbagai metrik evaluasi yang penting untuk menilai efektivitas model.

Rumus Akurasi mengukur proporsi prediksi yang benar dari total prediksi yang dibuat, dan dapat dihitung dengan rumus:

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Precision menunjukkan seberapa tepat prediksi positif yang dihasilkan oleh model, dihitung dengan rumus:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Recall (atau sensitivitas) mengukur kemampuan model dalam mendeteksi semua kasus positif yang ada, dengan rumus:

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

F1-Score adalah metrik yang menggabungkan precision dan recall menjadi satu nilai tunggal, yang sangat berguna ketika kita perlu menyeimbangkan antara keduanya. F1-Score dihitung dengan rumus:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

Macro Average menghitung rata-rata metrik evaluasi (seperti precision, recall, dan F1-Score) untuk setiap kelas secara terpisah. Macro Average dihitung dengan rumus:

$$Macro - F1 = \frac{\sum_{i=1}^N F1_i}{N} \quad (5)$$

Weighted Average menghitung rata-rata metrik evaluasi dengan mempertimbangkan jumlah contoh di setiap kelas, sehingga kelas yang lebih besar memiliki pengaruh lebih besar terhadap nilai rata-rata. Weighted Average dihitung dengan rumus:

$$Weighted - F1 = \frac{\sum_{i=1}^N F1_i \times w_i}{\sum_{i=1}^N w_i} \quad (6)$$

3. METODE PENELITIAN

Penelitian ini dilakukan melalui enam tahapan utama, yaitu Pengumpulan Data, Pra-pemrosesan Data, Implementasi Arsitektur Model CNN, Optimasi Model, Evaluasi Model, dan Perbandingan Hasil. Optimasi model dilakukan dengan menggunakan dua metode, yaitu SGD dan ADAM, yang hasilnya dievaluasi untuk perbandingan performa. Gambar 1 menunjukkan tahapan-tahapan dalam penelitian ini.

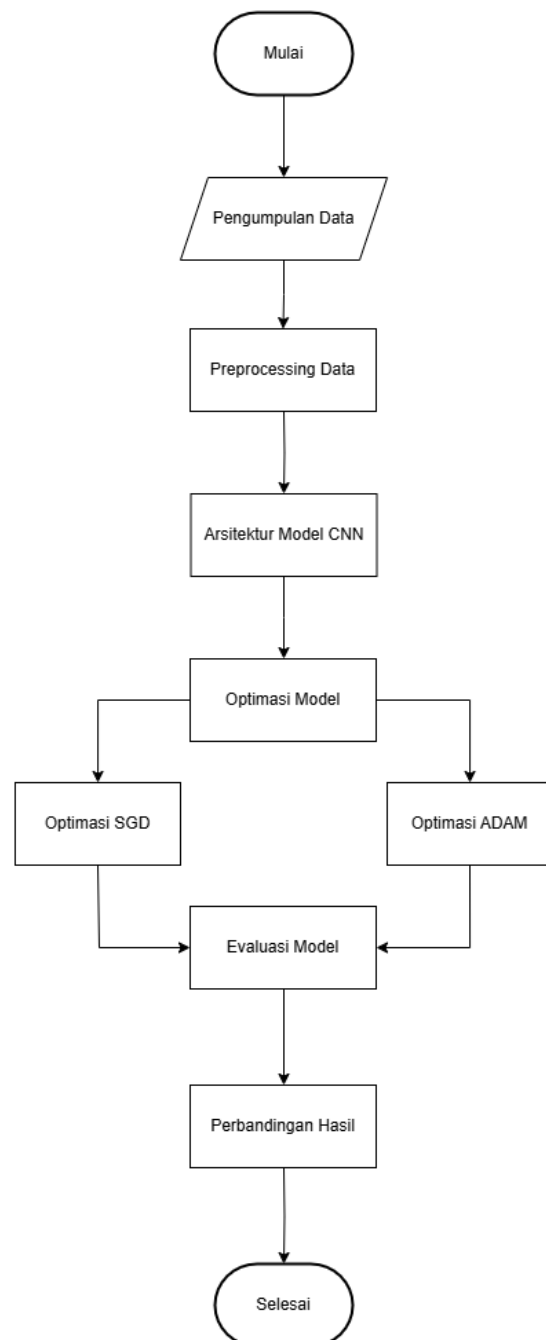
3.1. Pengumpulan Data

Penelitian ini menggunakan dataset yang diperoleh dari platform *Kaggle*, yang secara khusus menyediakan gambar jamur dengan dua kategori utama, yaitu *edible* (dapat dimakan) dan *poisonous* (beracun). Dataset ini dipilih karena relevansi dan kelengkapannya dalam mendukung penelitian yang berfokus pada klasifikasi visual menggunakan metode deep learning.

Dataset yang digunakan berisi gambar dari tiga jenis jamur yang sering dijadikan objek penelitian, yaitu *Apioperdon pyriforme*, *Amanita*, dan *Amanita citrina*. Untuk kategori *edible*, digunakan 1.000

gambar *Apioperdon pyriforme*. Sementara itu, untuk kategori *poisonous*, digunakan 750 gambar *Amanita* dan 250 gambar *Amanita citrina*. Pemilihan jenis-jenis ini dilakukan karena karakteristik visualnya yang signifikan dan jelas dalam membedakan kategori *edible* dan *poisonous*, sehingga dapat mempermudah proses pelatihan model.

Dataset awal diunduh dalam format terstruktur dengan gambar-gambar yang telah dikelompokkan berdasarkan kelasnya. Untuk menjaga keseimbangan data, jumlah gambar yang digunakan untuk setiap kelas adalah 1.000, sehingga total data yang digunakan dalam penelitian ini adalah 2.000 gambar.



Gambar 1. Flowchart Metode

3.2. Pra-proses Data

Dataset dibagi menjadi tiga subset utama secara manual dengan memastikan distribusi kelas yang seimbang di setiap subset. Proses pembagian dilakukan dengan cara menyortir dataset utama berdasarkan label kelas untuk memastikan distribusi yang seragam, kemudian dataset dibagi ke dalam tiga folder terpisah dengan proporsi 80% untuk data pelatihan, 10% untuk data validasi, dan 10% untuk data pengujian. Pembagian ini dilakukan secara acak namun tetap menjaga proporsi setiap kelas di setiap subset menggunakan skrip Python sederhana.

Langkah awal dalam pra-proses data adalah melakukan analisis distribusi ukuran gambar dalam dataset. Seluruh gambar dievaluasi dimensinya (lebar dan tinggi) menggunakan pustaka Pillow (PIL). Statistik deskriptif, seperti median, mean, persentil 25%, dan persentil 75% dihitung untuk memberikan gambaran karakteristik dimensi gambar mentah. Distribusi ukuran ini divisualisasikan dalam bentuk histogram untuk mengidentifikasi ukuran target yang optimal. Berdasarkan analisis statistik, ukuran target dipilih dari beberapa kandidat standar (128×128, 160×160, 192×192, atau 224×224). Penelitian ini memilih ukuran 160×160 karena berada dalam rentang distribusi persentil 25%-75% untuk lebar dan tinggi gambar. Pemilihan ukuran ini memastikan keseimbangan antara informasi visual yang dipertahankan dan efisiensi komputasi.

Setelah ukuran target dipilih, seluruh gambar dalam dataset diubah ukurannya ke dimensi 160×160 menggunakan metode *interpolasi LANCZOS*. Metode ini dipilih karena menjaga kualitas gambar selama proses scaling, terutama untuk detail gambar yang kompleks seperti tekstur jamur. Gambar yang telah diubah ukurannya disimpan dalam format terkompresi untuk mempermudah proses pelatihan model. Untuk meningkatkan variabilitas data dan mengurangi risiko *overfitting*, diterapkan teknik data augmentation yang dilakukan secara acak pada setiap batch selama proses pelatihan. Augmentasi ini mencakup rotasi hingga 40 derajat, translasi horizontal dan vertikal hingga 20% dari dimensi gambar, shear transform hingga 20%, zoom hingga 20%, dan flipping horizontal. Teknik augmentasi ini dilakukan menggunakan pustaka seperti *TensorFlow* (dengan *ImageDataGenerator*) atau *PyTorch* (dengan *torchvision.transforms*) untuk mensimulasikan berbagai kondisi nyata dan membantu model menjadi lebih robust terhadap variasi gambar di dunia nyata.

Untuk memastikan proses pra-proses data berjalan optimal, langkah-langkah ini dilakukan melalui pipeline manual yang terstruktur menggunakan skrip Python modular untuk mempermudah reproduksi dan meminimalkan kesalahan manual. Gambar yang telah diubah ukurannya disimpan dalam format yang efisien, seperti *TFRecord* (*TensorFlow*) atau *pickle* (*Python*), untuk mempercepat akses selama pelatihan. Selain itu, pemrosesan gambar dilakukan secara paralel menggunakan pustaka seperti *multiprocessing* atau

fungsi bawaan GPU untuk mempercepat waktu eksekusi.

3.3. Arsitektur Model CNN

Model Convolutional Neural Network (CNN) yang digunakan dalam penelitian ini dirancang untuk klasifikasi dua kelas, yaitu *edible* dan *poisonous*. Model terdiri dari tiga lapisan konvolusi dengan jumlah filter yang meningkat secara progresif (32, 64, 64). Setiap lapisan konvolusi menggunakan kernel berukuran 3×3 dengan fungsi aktivasi ReLU untuk memperkenalkan non-linearitas. Setelah setiap lapisan konvolusi, diterapkan lapisan *pooling* (*MaxPooling*) dengan ukuran 2×2 untuk mengurangi dimensi fitur sambil mempertahankan informasi penting.

Lapisan konvolusi diikuti oleh lapisan *Flatten* untuk meratakan output ke bentuk vektor satu dimensi, yang kemudian diteruskan ke lapisan *Dense*. Model memiliki dua lapisan *Dense*: lapisan pertama dengan 64 neuron dan fungsi aktivasi ReLU, serta lapisan output dengan jumlah neuron sesuai jumlah kelas (2 neuron) menggunakan fungsi aktivasi softmax. Lapisan output ini menghasilkan probabilitas untuk setiap kelas, yang kemudian digunakan untuk prediksi.

3.4. Optimasi Model

Model dikompilasi menggunakan dua jenis optimizer, yaitu *Stochastic Gradient Descent* (SGD) dan Adam, untuk membandingkan kinerja keduanya. Optimizer SGD digunakan dengan *learning rate* sebesar 0.01 dan momentum 0.9 untuk mempercepat konvergensi sambil menghindari osilasi. Di sisi lain, optimizer Adam digunakan dengan *learning rate* sebesar 1e-4, yang secara adaptif menyesuaikan langkah pembaruan parameter berdasarkan rata-rata momentum gradien. Kedua optimizer ini diuji dalam kondisi yang sama untuk mengevaluasi efektivitasnya dalam melatih model pada dataset ini.

Model dilatih selama maksimal 50 *epoch* dengan batch size sebesar 32. Untuk mencegah *overfitting*, diterapkan *callback Early Stopping* dengan parameter *patience* sebesar 10. Hal ini memastikan bahwa pelatihan dihentikan secara otomatis jika tidak ada peningkatan performa pada *validasi loss* selama 10 *epoch* berturut-turut.

3.5. Evaluasi Model

Kinerja model dievaluasi menggunakan data validasi yang dipisahkan sebelumnya (20% dari dataset). Setelah pelatihan, model menghasilkan prediksi untuk semua gambar dalam data validasi. Prediksi ini kemudian dibandingkan dengan label sebenarnya untuk menghitung metrik evaluasi, seperti akurasi, *F1-score*, dan *confusion matrix*. *Confusion Matrix* adalah sebuah tabel yang menampilkan jumlah data uji yang diklasifikasikan dengan nilai yang benar dan salah [6]. Selain itu, laporan klasifikasi juga dihasilkan untuk menilai performa model pada masing-masing kelas (*edible* dan *poisonous*).

Laporan Klasifikasi:				
	precision	recall	f1-score	support
edible	0.94	0.97	0.96	100
poisonous	0.97	0.94	0.95	100
accuracy			0.95	200
macro avg	0.96	0.95	0.95	200
weighted avg	0.96	0.95	0.95	200

Gambar 1. Laporan Evaluasi optimizer SGD

Laporan Klasifikasi:				
	precision	recall	f1-score	support
edible	0.94	0.95	0.95	100
poisonous	0.95	0.94	0.94	100
accuracy			0.94	200
macro avg	0.95	0.94	0.94	200
weighted avg	0.95	0.94	0.94	200

Gambar 2. Laporan Evaluasi optimizer ADAM

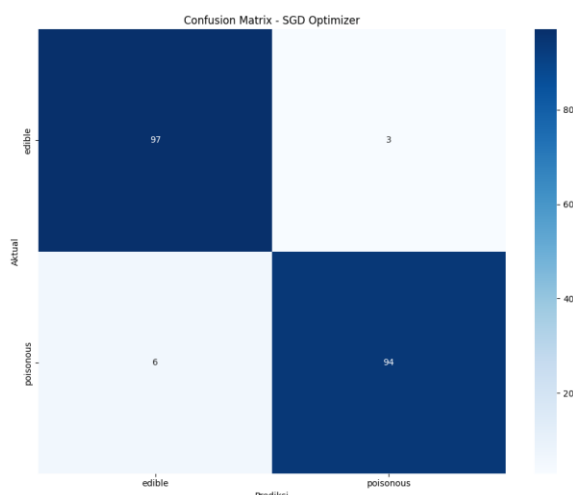
Kurva pelatihan dan validasi model untuk metrik akurasi dan loss diplot untuk memvisualisasikan performa selama pelatihan. Confusion matrix divisualisasikan menggunakan heatmap untuk memberikan gambaran detail tentang distribusi prediksi model, termasuk kesalahan klasifikasi.

Akhirnya, hasil model dengan *optimizer* SGD dan Adam dibandingkan berdasarkan akurasi dan F1-score. Optimizer dengan kinerja terbaik dipilih, dan model yang dihasilkan disimpan dalam format Keras model file (.keras) untuk digunakan di masa depan.

4. HASIL DAN PEMBAHASAN

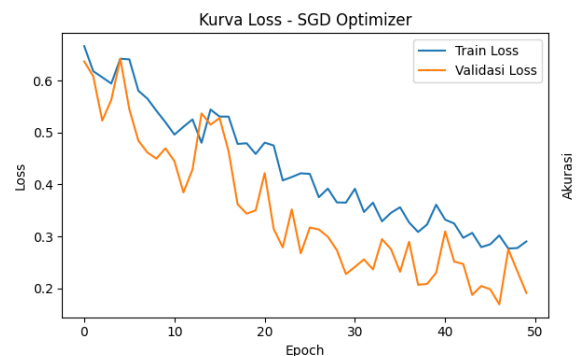
4.1 Hasil Eksperimen dengan Optimizer SGD

Model CNN dengan *optimizer* SGD menunjukkan performa yang sangat memuaskan dalam klasifikasi jamur. Hasil evaluasi pada data testing mencapai akurasi 95.50% dengan F1-Score 0.9550, yang mengindikasikan keseimbangan yang sangat baik antara presisi dan *recall*. Performa ini menunjukkan kemampuan model yang konsisten dalam mengidentifikasi kedua kelas jamur.



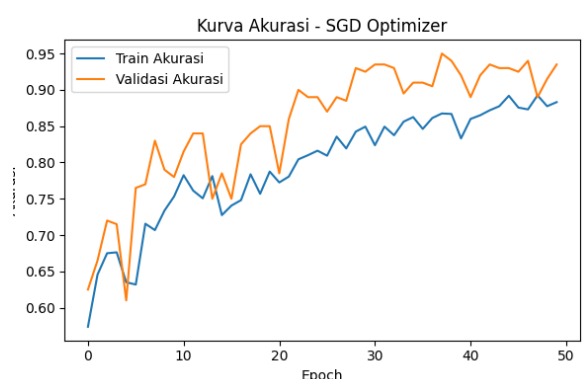
Gambar 3. Confusion Matrix Model dengan Optimizer SGD

Berdasarkan *confusion matrix* pada Gambar 1, dapat dianalisis bahwa dari 100 sampel jamur edible, 97 sampel (97%) berhasil diklasifikasikan dengan benar sebagai *edible* (*True Positive*) dan hanya 3 sampel (3%) yang salah diklasifikasi sebagai *poisonous* (*False Negative*). Sementara itu, dari 100 sampel jamur poisonous, 94 sampel (94%) berhasil diklasifikasi dengan benar sebagai *poisonous* (*True Negative*) dan 6 sampel (6%) salah diklasifikasi sebagai *edible* (*False Positive*). Hasil ini menunjukkan tingkat kesalahan yang sangat rendah, terutama untuk klasifikasi jamur *edible* yang hanya memiliki *error rate* 3%.



Gambar 4. Kurva Loss Model dengan Optimizer SGD

Kurva *loss* menggambarkan perubahan nilai kerugian (*loss*) selama proses pelatihan model menggunakan *optimizer* SGD. Pada kurva ini, terlihat bahwa *training loss* menunjukkan penurunan yang gradual dan konsisten, dari nilai awal sekitar 0.65 hingga mencapai 0.29 pada epoch terakhir. Di sisi lain, *validation loss* juga mengalami pola penurunan yang serupa, dari 0.62 menjadi 0.20. Pola ini menunjukkan bahwa model terus belajar dengan baik dari data yang diberikan, mengurangi kesalahan prediksi seiring bertambahnya *epoch*. Selain itu, jarak antara *training loss* dan *validation loss* relatif kecil, yang menandakan model memiliki generalisasi yang baik tanpa *overfitting*.



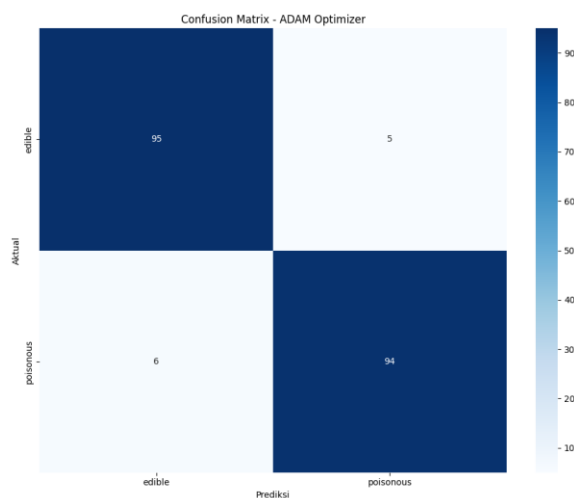
Gambar 5. Kurva Akurasi Model dengan Optimizer SGD

Kurva akurasi menunjukkan peningkatan performa klasifikasi model selama proses pelatihan. *Training accuracy* meningkat secara bertahap dari nilai awal 57% hingga mencapai 89% pada epoch ke-

50, menunjukkan bahwa model semakin baik dalam mengenali pola pada data latih. Sementara itu, *validation accuracy* menunjukkan hasil yang lebih tinggi, yaitu mencapai 93% di akhir pelatihan. Performa *validation accuracy* yang konsisten lebih tinggi daripada *training accuracy* mengindikasikan bahwa model mampu menangani data baru dengan baik, menunjukkan proses pembelajaran yang stabil dan efektif.

4.2 Hasil Eksperimen dengan Optimizer Adam

Implementasi optimizer Adam menghasilkan performa yang juga sangat baik dengan akurasi 94.50% dan *F1-Score* 0.9450 pada data *testing*. Meskipun sedikit lebih rendah dari SGD, hasil ini tetap menunjukkan kemampuan model yang excellent dalam klasifikasi jamur.



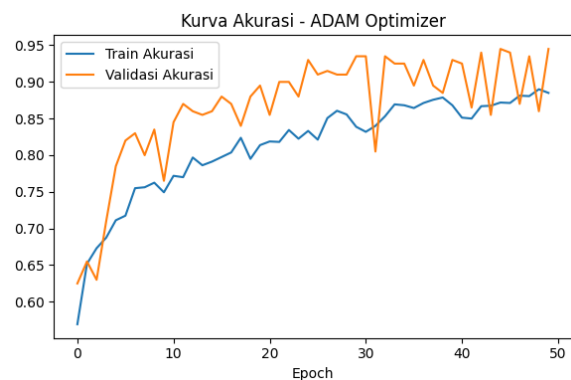
Gambar 6. Confusion Matrix Model dengan Optimizer Adam

Detail analisis confusion matrix Adam menunjukkan distribusi yang sedikit berbeda. Dari 100 sampel jamur *edible*, 95 sampel (95%) terklasifikasi dengan benar sebagai *edible* (*True Positive*) dan 5 sampel (5%) salah klasifikasi sebagai *poisonous* (*False Negative*). Untuk sampel jamur *poisonous*, 94 sampel (94%) terklasifikasi dengan benar sebagai *poisonous* (*True Negative*) dan 6 sampel (6%) salah klasifikasi sebagai *edible* (*False Positive*).



Gambar 7. Kurva Loss Model dengan Optimizer Adam

Kurva loss pada model dengan optimizer Adam menunjukkan penurunan yang cepat di awal pelatihan, dari nilai awal sekitar 0.70 hingga mencapai 0.30 pada epoch terakhir. Hal ini menandakan bahwa model dengan cepat belajar untuk mengurangi kesalahan pada data latih. Namun, *validation loss* terlihat lebih fluktuatif dengan beberapa lonjakan (*spike*) yang cukup signifikan. Selain itu, terdapat gap yang lebih besar antara *training loss* dan *validation loss* dibandingkan dengan optimizer SGD. Pola ini mengindikasikan bahwa model cenderung sedikit *overfit* pada data latih, sehingga memengaruhi konsistensi performa pada data validasi.



Gambar 8. Kurva Akurasi Model dengan Optimizer Adam

Kurva akurasi menunjukkan bahwa *training accuracy* dengan optimizer Adam meningkat dengan cepat sejak awal pelatihan, mencapai sekitar 90% pada epoch terakhir. Sementara itu, *validation accuracy* mengalami fluktuasi dengan tren yang cenderung mendatar (*plateau*) di sekitar 90%. Beberapa penurunan tajam (*drop*) pada *validation accuracy* menunjukkan bahwa proses pembelajaran model kurang stabil dibandingkan dengan hasil yang diperoleh menggunakan optimizer SGD. Meskipun demikian, model masih menunjukkan performa yang baik dengan akurasi yang tetap tinggi di akhir pelatihan.

4.3 Analisis Perbandingan Kedua Optimizer

Tabel di bawah ini menyajikan perbandingan hasil evaluasi model menggunakan optimizer SGD dan Adam pada data *testing*:

Tabel 1. Perbandingan Performansi Model Menggunakan Optimizer SGD dan Adam pada Data Testing

Metrik	SGD	Adam
Akurasi	95.50%	94.50%
<i>F1-Score</i>	0.9550	0.9450
<i>Precision (Edible)</i>	94%	94%
<i>Recall (Edible)</i>	97%	95%
<i>Precision (Poisonous)</i>	97%	95%
<i>Recall (Poisonous)</i>	94%	94%
<i>Macro Average</i>	0.95	0.94
<i>Weighted Average</i>	0.95	0.94

Dari tabel tersebut, terlihat bahwa optimizer **SGD** unggul sedikit dibandingkan optimizer **Adam** dalam hal akurasi dan F1-Score. Optimizer SGD memiliki keunggulan dalam recall untuk kelas jamur edible, yang mencapai **97%** dibandingkan **95%** pada optimizer Adam. Secara keseluruhan, kedua optimizer memberikan hasil yang baik, namun optimizer SGD menunjukkan performa yang lebih stabil.

5. KESIMPULAN

Berdasarkan hasil penelitian, model Convolutional Neural Network (CNN) menunjukkan performa yang sangat baik dalam klasifikasi jamur edible dan poisonous dengan menggunakan dua jenis optimizer, yaitu Stochastic Gradient Descent (SGD) dan Adam, di mana optimizer SGD menghasilkan akurasi tertinggi sebesar 95,50% dan F1-Score 0,9550, sedangkan optimizer Adam mencapai akurasi 94,50% dan F1-Score 0,9450. Optimizer SGD unggul dalam recall pada kelas edible yang mencapai 97% dibandingkan Adam yang memiliki recall sebesar 95%, sementara analisis kurva loss dan akurasi menunjukkan bahwa model dengan optimizer SGD lebih stabil dan memiliki kemampuan generalisasi yang lebih baik dibandingkan Adam yang cenderung mengalami fluktuasi dan sedikit overfitting pada data latih. Kedua model memiliki tingkat false positive yang rendah, hanya 6%, yang sangat penting dalam mengurangi risiko kesalahan klasifikasi jamur beracun, sehingga CNN yang dioptimalkan menggunakan SGD menjadi pilihan yang lebih unggul untuk identifikasi jamur secara akurat dan andal. Untuk penelitian selanjutnya, disarankan untuk menambah variasi spesies jamur dalam dataset untuk meningkatkan kemampuan model dalam mengidentifikasi lebih banyak jenis jamur yang ada di alam.

DAFTAR PUSTAKA

- [1] I. P. Putra, "Studi taksonomi dan potensi beberapa jamur liar di Pulau Belitung," *Justek: Jurnal Sains dan Teknologi*, vol. 3, no. 1, pp. 24–31, May 2020, doi: <https://doi.org/10.31764/justek.v3i1.3534>.
- [2] D. Kawasan Hutan Liang Bukal, M. HULU SUMBAWA Indah Dwi Lestari, and U. Tsanyiah Fauziah, "IDENTIFIKASI KEANEKARAGAMAN JENIS FUNGI MAKROSKOPIS," *NN-NN*, 2022. Accessed: Dec. 26, 2024. [Online]. Available: <https://www.e-journalppmunsa.ac.id/index.php/kependidikan/article/view/1096>
- [3] I. P. Putra, "PANDUAN KARAKTERISASI JAMUR MAKROSKOPIS DI INDONESIA: BAGIAN 1-DESKRIPSI CIRI MAKROSKOPIS (Guide for Indonesian Macroscopic fungi Characterization: Part I-Description of Macroscopic Characteristics)," 2021. doi: <https://doi.org/10.18330/jwallacea.2021.vol10iss1pp25-37>.
- [4] U. Sri Rahmadhani and N. Lysbetti Marpaung, "Klasifikasi Jamur Berdasarkan Genus Dengan Menggunakan Metode CNN," vol. 8, no. 2, May 2023, Accessed: Dec. 25, 2024. [Online]. Available: <https://ejournal.poltekharber.ac.id/index.php/informatika/article/view/5229/2460>
- [5] B. Nugroho, E. Y. Puspaningrum, and M. S. Munir, "Kinerja Algoritma Optimasi Root-Mean-Square Propagation dan Stochastic Gradient Descent pada Klasifikasi Pneumonia Covid-19 Menggunakan CNN," *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, 2021, [Online]. Available: <https://api.semanticscholar.org/CorpusID:245695016>
- [6] A. Sandy Wardhani, F. Tri Anggraeny, and A. Mustika Rizki, "PENERAPAN MODEL HIBRIDA CNN-KNN UNTUK KLASIFIKASI PENYAKIT MATA," 2024. doi: <https://doi.org/10.36040/jati.v8i3.9774>.
- [7] D. Oleh, : Shidqi, and A. Hauzan, "PENERAPAN CONVOLUTIONAL NEURAL NETWORK DALAM PENGKLASIFIKASIAN CITRA GAMBAR JAMUR BERACUN," Jakarta, Jul. 2023. Accessed: Dec. 25, 2024. [Online]. Available: <https://repository.uinjkt.ac.id/dspace/handle/123456789/72929>
- [8] N. Zatadinia and H. Dwi Bhakti, "IDENTIFIKASI JAMUR YANG DAPAT DIKONSUMSI DAN BERACUN MENGGUNAKAN METODE LOGISTIC REGRESSION," 2024. doi: <https://doi.org/10.36040/jati.v8i6.11601>.
- [9] N. Fajria, dan Yulindon, J. Teknik Elektro, P. Negeri Padang, J. Limau Manih Padang, and C. Author, "Deteksi Jamur Beracun dan Tidak Beracun Menggunakan CNN dan YOLO," 2024, doi: 10.30630/eji.0.0.453.
- [10] Sandy Andika Maulana, Shabrina Husna Batubara, Tasya Ade Amelia, and Yohanna Permata Putri Pasaribu, "Penerapan Metode CNN (Convolutional Neural Network) Dalam Mengklasifikasi Jenis Ubur-Ubur," *Jurnal Penelitian Rumpun Ilmu Teknik*, vol. 2, no. 4, pp. 122–130, Dec. 2023, doi: 10.55606/juprit.v2i4.3084.
- [11] 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.
- [12] I. Perlindungan, *PENGENALAN TANAMAN CABAI DENGAN TEKNIK KLASIFIKASI MENGGUNAKAN METODE CNN*. 2020. Accessed: Dec. 25, 2024. [Online]. Available: <https://conference.upnvj.ac.id/index.php/senamika/article/view/694>
- [13] K. G. Putra, "Penentuan Penerima Bantuan Program Keluarga Harapan menggunakan Algoritma Random Forest di Desa Kebonrejo," *Innovative: Journal Of Social Science Research*, vol. 4, no. 5, pp. 7242–7257, 2024. doi: 10.31004/innovative.v4i5.15557.