Journal of Science and Social Research ISSN 2615 – 4307 (Print)
August 2025, VIII (3): 3872 – 3880 ISSN 2615 – 3262 (Online)
Available online at http://jurnal.goretanpena.com/index.php/JSSR

EVALUASI DENSENET-201 UNTUK IDENTIFIKASI BIJI KOPI MENGGUNAKAN HYPERPARAMETER GRIDSEARCH

Yuke Manza[1], Lima Hartimar Rambe[2], Kiki Putri Ani Siregar[3], Rika Rosnelly[4], Adil Setiawan[5]
Universitas Potensi Utama, Medan
e-mail: [1]manzayuke@gmail.com, [2]limahartimarrambe12345@gmail.com, [3]putrianikiki1@gmail.com, [4]rika@potensi-utama.ac.id, [5]adio165@gmail.com

Abstract: *Coffee is one of the most important commodities in the global agricultural sector. However, the manual sorting process of coffee beans, which is still widely applied in the Small and Medium Industry (IKM) sector, tends to be time-consuming and often results in inconsistent quality assessments. This study aims to classify coffee bean quality using the DenseNet-201 deep learning architecture, optimized with the GridSearch method to obtain the best combination of hyperparameters. The dataset used consists of 450 images of coffee beans divided into two classes: good-quality and defective beans. The model was trained for 20 epochs using a transfer learning approach and evaluated using performance metrics such as accuracy, precision, recall, and F1-score. The test results show that the model before optimization achieved an accuracy of only 78.67%, while the model optimized with GridSearch reached a high accuracy of 99.47% with a low loss value. These findings indicate that the application of DenseNet-201 with hyperparameter tuning is capable of producing accurate and stable classification results, and can be relied upon as an automated solution for sorting coffee beans based on their quality.*

Keywords: *Deep Learning, DenseNet201, Hyperparameter, GridSearch, Coffee Bean Classification*

Abstrak: Kopi merupakan salah satu komoditas penting dalam sektor pertanian global. Namun, proses pemilahan biji kopi secara manual yang masih banyak diterapkan pada sektor Industri Kecil dan Menengah (IKM) cenderung memakan waktu dan menghasilkan penilaian kualitas yang tidak konsisten. Penelitian ini bertujuan untuk mengklasifikasikan kualitas biji kopi menggunakan arsitektur Deep Learning DenseNet-201 yang dioptimalkan dengan metode GridSearch untuk memperoleh kombinasi hyperparameter terbaik. Dataset yang digunakan terdiri dari 450 gambar biji kopi dengan dua kelas: biji kopi bagus dan biji kopi rusak. Model dilatih selama 20 epoch dengan pendekatan transfer learning dan dilakukan evaluasi terhadap performa model menggunakan metrik akurasi, precision, recall, dan f1-score. Hasil pengujian menunjukkan bahwa model sebelum optimasi hanya mencapai akurasi sebesar 78,67%, sedangkan model dengan optimasi GridSearch mampu mencapai akurasi tinggi sebesar 99,47% dan nilai loss yang rendah. Hal ini menunjukkan bahwa penerapan DenseNet-201 dengan tuning hyperparameter mampu menghasilkan klasifikasi yang akurat dan stabil, serta dapat diandalkan sebagai solusi otomatis dalam proses sortasi biji kopi berdasarkan kualitasnya.

Kata kunci: Deep Learning, DenseNet201, Hyperparameter, GridSearch, Klasifikasi Biji Kopi

PENDAHULUAN

Kopi merupakan salah satu minuman paling populer di dunia dan memiliki peranan penting dalam sektor pertanian global. Kopi juga menjadi salah satu komoditas utama dengan potensi besar dalam pengembangan pasar dan

perdagangan internasional. Setiap jenis biji kopi memiliki karakteristik unik yang membedakannya satu sama lain, baik dari segi rasa, aroma, kadar kafein, maupun harga pasar.

Di berbagai wilayah, terutama pada sektor Industri Kecil dan Menengah (IKM), proses pemilahan biji kopi masih dilakukan secara manual. Metode tradisional ini biasanya melibatkan tenaga manusia untuk memilah biji kopi yang cacat dari yang berkualitas baik. Selain memerlukan waktu yang lama, proses manual juga rawan terhadap ketidak konsistenan karena adanya faktor subjektif dalam penilaian mutu biji kopi.

Kondisi ini menjadi tantangan serius dalam menjamin kualitas produk akhir. Oleh karena itu, proses sortasi biji kopi sangat penting untuk dilakukan secara sistematis dan objektif guna memastikan pemisahan biji kopi cacat dari biji kopi yang memenuhi standar kualitas. Dengan proses sortasi yang baik, produsen kopi dapat menghasilkan kopi dengan cita rasa yang lebih konsisten serta memenuhi standar mutu yang ditetapkan di pasar.

Perkembangan teknologi deep learning membuka peluang baru dalam klasifikasi kopi secara lebih efektif. Salah satu metode yang dapat diimplementasikan adalah CNN. CNN merupakan singkatan dari Convolutional Neural Network, yaitu salah satu jenis jaringan saraf tiruan (neural network) yang paling banyak digunakan dalam bidang pengolahan citra (image processing) dan visi komputer (computer vision). CNN memiliki banyak varian arsitektur yang dikembangkan untuk meningkatkan akurasi, efisiensi, dan kedalaman jaringan. Salah satunya adalah DenseNet201 . DenseNet memiliki keunggulan mengurangi over fitting dan efisiensi parameter. Karena fitur-fitur lama tidak perlu dipelajari ulang, model ini membutuhkan jumlah parameter yang lebih sedikit dibandingkan arsitektur lain dengan performa setara.

Melakukan Klasifikasi Citra Penyakit Tanaman Pada Daun Paprika Dengan Metode DenseNet-201. Data yang digunakan berupa 4.876 gambar daun paprika. Data tersebut dibagi menjadi data latih, data validasi, dan data uji.Hasil penelitian menunjukkan bahwa model transfer learning mampu mendeteksi penyakit bercak daun pada paprika dengan akurasi keseluruhan sekitar 99.5%. Evaluasi model terhadap kelas “Bacterial Spot” dan “Healthy” menghasilkan precision, recall, dan F1-score rata-rata sekitar 99.5%. Penelitian ini menunjukkan bahwa metode transfer learningdapat digunakan sebagai sistem deteksi penyakit tanaman yang efektif dan efisien Selain itu, menggunakan SVM untuk klasifikasi biji kopi robusta berdasarkan circularity dan eccentricity. Circularity dan eccentricity digunakan dalam proses ekstraksi fitur, Sedangkan algoritme gridsearch digunakan untuk optimasi parameter SVM dalam proses klasifikasi pada 4 kernel berbeda. Kajian ini menghasilkan model klasifikasi terbaik dengan akurasi tertinggi sebesar 94% pada kernel RBF dan Polynomial.

Penelitian-penelitian tersebut memberikan landasan teoritis dan empiris untuk mengembangkan identifikasi biji kopi. Bisa disimpulkan bahwa belum ada implementasi DenseNet-201 menggunakan optimasi Gridsearch untuk klasifikasi biji kopi. Berdasarkan tinjauan tersebut, peneliti tertarik untuk mengevaluasi kinerja DenseNet-201 untuk identifikasi biji kopi menggunakan Hyperparameter Gridsearch untuk mengklasifikasikan jenis biji kopi berdasarkan kualitasnya supaya

Kopi merupakan salah satu tanaman perkebunan yang menjadi sumber penghasilan rakyat dan juga dapat menjadi sumber peningkat devisa negara melalui ekspor biji kopi mentah maupun olahan dari biji kopi.

Algoritma deep learning memiliki kemampuan yang unik, yaitu dapat mengekstraksi fitur dari objek secara otomatis, akan tetapi algoritma ini membutuhkan dataset yang besar dan proses pelatihan yang lama. Oleh karena

itu, kombinasi deep learning dengan gridsearch sering digunakan untuk mengatasi permasalahan tersebut. Kombinasi ini merupakan teknik untuk memanfaatkan model yang telah dilatih sebelumnya dengan dataset yang besar untuk melakukan penyesuaian dengan permasalahan yang serupa. Hyperparameter tuning diperlukan untuk menghasilkan model klasifikasi yang akurat. Penelitian ini bertujuan membangun model klasifikasi biji kopi dengan model DenseNet-201 dan pemilihan hyperparameter menggunakan metode gridsearch.

GridSearch adalah teknik optimasi hyperparameter tuning yang mencari kombinasi parameter terbaik dari sebuah model.

Evaluasi performa model klasifikasi dilakukan dengan menggunakan beberapa metrik pengukuran standar yang terdiri dari accuracy (mengukur ketepatan prediksi secara keseluruhan), precision (mengukur ketepatan prediksi positif), recall (mengukur kemampuan model mendeteksi kelas positif), dan F1-score (rata-rata harmonik antara precision dan recall). Hasil penelitian sebelumnya menunjukkan bahwa proses optimasi hyperparameter menggunakan gridsearch memberikan peningkatan performa yang signifikan. Hasil akurasi sebelum optimasi adalah 73,31%, sedangkan setelah optimasi adalah 94,8%. Hal ini menunjukkan peningkatan akurasi sebesar 21,5% setelah menerapkan metode grid search cv pada klasifikasi pemantauan kualitas udara menggunakan model klasifikasi.

METODE

Penelitian ini menggunakan metode klasifikasi DenseNet-201 untuk menganalisis dataset biji kopi yang diunduh dari *Kaggle* pada tahun 2024. Metode penelitian ini dijelaskan melalui tahapan berikut:

Dataset

Dataset yang digunakan dalam penelitian ini berasal dari Kaggle yang dipublikasikan pada tahun 2024 dan diterbitkan ole Afiyah Musrah. Dataset terdiri dari 450 gambar biji kopi dengan kategori biji bagus dan biji rusak.

Tahapan Penelitian

Penelitian ini dilakukan dalam beberapa tahap dimulai dari:

Pre-processing Data

Pada tahapan ini dilakukan beberapa teknik pre-*processing* yang bertujuan untuk membantu keakuratan dari model yang dijalankan, Teknik yang dilakukan ialah sebagai berikut:

1. Split data
   Data dibagi kedalam 2 bagian utama yaitu data training dan data testing yang masing-masing berisi dua kelas (rusak, bagus)
2. Resize
   Semua gambar diubah ukurannya menjadi 180x180 piksel agar konsisten.
3. Normalisasi
   Menyesuaikan skala piksel dari 0–255 menjadi 0–1 untuk mempercepat pelatihan dan menghindari dominasi nilai besar.
4. Label Binner
   Label dikonversi ke format biner (0 dan 1) karena hanya ada dua kelas: bagus:0, rusak:1
5. Shuffle Data: Digunakan untuk mencegah bias urutan
   Untuk data latih, gambar diacak urutannya agar model tidak belajar berdasarkan urutan file (bias/bias urutan).
6. Optimisasi Pipeline
   a. Cache: Menyimpan dataset di memori agar tidak perlu membaca ulang dari disk.
   b. Shuffle: Mengacak hingga 1000 data (hanya untuk data latih).
   c. Prefetch: Menyiapkan batch selanjutnya saat batch sekarang sedang dilatih untuk mempercepat training**.**

Metode DenseNet-201

DenseNet201 adalah varian dari arsitektur DenseNet yang memiliki 201 lapisan (layers) dalam jaringannya. Yang membedakan DenseNet dengan CNN konvensional adalah cara koneksi antar lapisan: setiap layer dihubungkan secara langsung dengan semua layer sebelumnya. Teknik DenseNet-201 yang dilakukan pada tahapan ini sebagai berikut:

1. Data augmentasion: Teknik augmentasi untuk mencegah overfitting dan memperkaya data latih.
2. Input & Normalisasi: Input layer menerima gambar RGB dan diskalakan ke rentang [0,1] untuk menstabilkan optimasi.
3. Layer Konvolusi + Pooling: Dua lapis CNN dengan jumlah filter yang bisa dituning dilakukan untuk mengurangi overfitting dan beban komputasi.
4. Dropout: Untuk regularisasi dan mencegah overfitting.
5. Output Layer: Karena klasifikasi biner, digunakan sigmoid.
6. Compiler: Optimizer dan learning rate yang dapat dituning. Mengatur bagaimana model mempengaruhi bobot berdasarkan loss.

Optimasi Menggunakan *Gridsearch*

Gridsearch digunakan untuk memperoleh hyperparameter terbaik dari DenseNet-201. Implementasi *gridsearch* dilakukan dengan detail sebagai berikut:

1. Inisialisasi Tuner *Random Search*
   build_model: Fungsi yang sudah Anda definisikan sebelumnya.
   objective='val_accuracy': Metrik yang dioptimalkan adalah akurasi validasi.
   max_trials=10: Maksimal 10 kombinasi hyperparameter akan dicoba.
   executions_per_trial=1: Setiap kombinasi dilatih sekali.
   directory dan project_name: Tempat hasil tuning disimpan *(cache, log,* dan model terbaik).
2. Melihat ruang pencarian: mencetak semua hyperparameter yang bisa dituning beserta rentang nilai.
3. Melakukan Pencarian hyperparameter: Melatih model dengan dataset train_dataset selama 10 epoch, dan memvalidasi pada validation dataset.

Evaluasi Model

Evaluasi performa dilakukan menggunakan *confusion matrix* untuk memvisualisai hasil akurasi dan loss pada data pengujian.

Berikut komponen yang digunakan untuk mengidentifikasi hasil prediksi menggunakan *confusion matrix*:

1. TP (*True Positive*) ini adalah jumlah data yang memiliki nilai aktual positif dan telah diprediksi dengan benar sebagai positif.
2. TN (*True Negative*) adalah merupakan jumlah data yang memiliki nilai aktual negatif dan telah diprediksi dengan benar sebagai negatif.
3. FP (*False Positive)* ini adalah jumlah data yang memiliki nilai aktual negatif, tetapi diprediksi sebagai positif dengan tidak benar.
4. FN (*False Negative*) merupakan jumlah data yang memiliki nilai aktual positif, tetapi diprediksi sebagai negatif dengan tidak benar.

Melalui *confusion matrix*, dapat dilakukan perhitungan untuk mengidentifikasikan performa dari model *machine learning* yang digunakan. Nilai yang dapat dihitung untuk mengidentifikasi performa, yaitu akurasi, *recall*, *precision*, dan *f1-score*.

Akurasi, dihitung dengan cara membagi jumlah data yang diklasifikasikan benar oleh model dengan total data.

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN}$$

Precision, dihitung dengan cara membagi jumlah data *True Positive* dengan jumlah data *True Positive* ditambah data *False Positive*.

Journal of Science and Social Research ISSN 2615 – 4307 (Print)
August 2025, VIII (3): 3872 – 3880 ISSN 2615 – 3262 (Online)
Available online at http://jurnal.goretanpena.com/index.php/JSSR

$$\text{Presisi} = \frac{TP}{TP+FP}$$

Recall, dihitung dengan cara membagi jumlah data *True Positive* dengan jumlah data *True Positive* ditambah data *False Negative*.

$$\text{Recall} = \frac{TP}{TP+FN}$$

F1-score didapatkan dengan cara pembagian hasil perkalian *precision* dan *recall* dengan hasil penjumlahannya lalu dikalikan dua.

$$\text{F1-Score} = \frac{2 . Presisi . Recall}{Presisi + Recall}$$

Setelah evaluasi, peneliti bisa menganalisis kinerja algoritma DenseNet-201 terhadap pengklasifikasin data. Pengaruh optimasi *Gridsearch* dalam mencari parameter terbaik dari algoritma, dan bagaimana proses dan hasil yang diberikan oleh model pengujian.

HASIL DAN PEMBAHASAN

Setelah melalui tahapan *pre-processing* data, ditemukan 300 gambar dari dua kategori pada data latih. 150 gambar dengan kategori yang sama pada data uji. Kategori yang ditemukan: ['bijikopibagus', 'bijikopirusak']

Gambar 1 Dataset Biji Kopi

Bangun Model DenseNet-201

Pengembangan model DenseNet-201 dilakukan dengan membangun arsitektur model sequential_1 untuk melakukan klasifikasi citra berukuran 180x180 piksel dengan 3 saluran warna (RGB). Hasil performa model ditampilkan pada gambar dibawah ini.

Model: "sequential_1"

Layer (type)	Output Shape	Param #
sequential (Sequential)	(None, 180, 180, 3)	0
rescaling (Rescaling)	(None, 180, 180, 3)	0
conv2d (Conv2D)	(None, 178, 178, 32)	896
max_pooling2d (MaxPooling2D)	(None, 89, 89, 32)	0
conv2d_1 (Conv2D)	(None, 87, 87, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 43, 43, 64)	0
conv2d_2 (Conv2D)	(None, 41, 41, 128)	73,856
max_pooling2d_2 (MaxPooling2D)	(None, 20, 20, 128)	0
flatten (Flatten)	(None, 51200)	0
dense (Dense)	(None, 128)	6,553,728
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 1)	129

Gambar 2 Hasil Sequantial_1

Berdasarkan hasil perfoma model yang dicantumkan pada Gambar 2, secara keseluruhan, model ini memiliki total parameter sebanyak 6.647.105, ini menunjukkan bahwa model cukup kompleks.

Untuk hasil pengkasifikasian model belum begitu sempurna karena hasil evaluasi yang di peroleh tidak begitu signifikan, seperti yang dijelaskan pada

Tabel 1 DenseNet Tanpa Optimasi

Label	Precision	Recall	F1-Score	Support
Biji kopibagus	1.00	0.56	0.72	75
Biji kopirusak	0.69	1.00	0.82	75

Berdasarkan Tabel. 1 ini dijelaskan Meskipun precision untuk kelas "bijikopibagus" sangat tinggi (1.00), namun nilai recall-nya rendah (0.56), menandakan banyak biji kopi bagus tidak dikenali dengan benar. Sebaliknya, kelas "bijikopirusak" memiliki recall sempurna (1.00) tetapi precision-nya lebih rendah (0.69), mengindikasikan adanya salah klasifikasi biji kopi bagus sebagai rusak.

Model ini menunjukkan ketidakseimbangan prediksi antar kelas dan indikasi overfitting, di mana model hanya mampu mengingat pola tertentu namun gagal dalam generalisasi.

Optimasi *Gridsearch*

Setelah melalui tahapan klasifikasi menggunakan DenseNet-201 kemudian dilakukan Teknik Optimasi dengan tujuan mencari parameter terbaik dari DenseNet-201. Untuk mencari parameter tersebut digunakan rumus umum *gridsearch* dengan menetapkan parameter yang diuji.
Rumus *Gridsearch*

$$Total\ kombinasi = i = 1 \prod n \mid vi \mid$$

P...p(n) : Jumlah parameter yang diuji
Vi : Jumlah nilai yang diuji untuk parameter p(n)

Pada optimasi *Gridsearch* dihasilkan beberapa parameter terbaik dari DenseNet-201 sebagai berikut:

1. Filter Konv 1: 32
2. Filter Konv 2: 96
3. Unit Dense: 128
4. Dropout Rate: 0.20
5. Learning Rate: 0.001

Latih Model Dengan Hyperparameter terbaik

Setelah parameter terbaik dari DenseNet201 ditemukan, kemudian dilaukan pengujian guna memperoleh hasil yang lebih akurat. Proses yang dilakukan ialah:

1. Pada tahap ini, kombinasi hyperparameter terbaik digunakan
2. Latih Model dengan Dataset Pelatihan Selama 20 Epoch. Model akan memproses seluruh dataset pelatihan sebanyak 20 kali.
3. Simpan Riwayat Pelatihan ke dalam Objek history. Nilai loss dan accuracy untuk setiap epoch pada data latih dan validasi. Dapat digunakan untuk visualisasi: grafik akurasi dan loss terhadap epoch.

Setelah dilakukan pengujian kembali menggunakan parameter terbaik, bisa dilihat hasil terbaru memiliki akurasi yang lebih tinggi dari sebelumnya seperti yang dijelaskan pada Tabel 1 berikut.

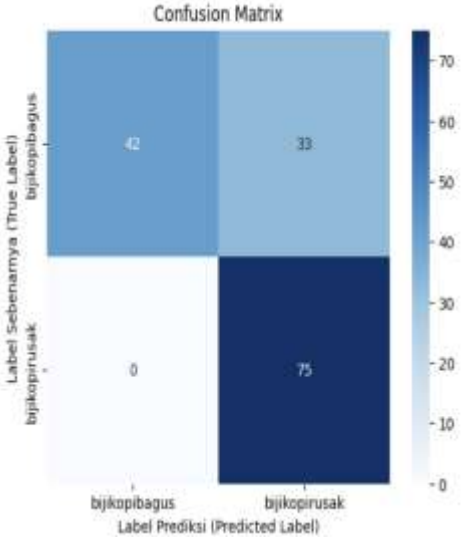
Tabel 2 DenseNet Dengan Parameter Terbaik

Metrik	Biji kopibagus	biji kopirusak	Rata-rata
Precision	1.00	0.98	0.99
Recall	0.99	0.99	0.99
F1-Score	0.99	0.98	0.985
Akurasi Total			99,47%

Berdasarkan hasil pengujian dengan parameter terbaik, model DenseNet-201 yang telah dioptimasi menunjukkan performa yang sangat baik dengan akurasi pada data uji mencapai 99,47%.

Evaluasi Dan Confusion Matriks

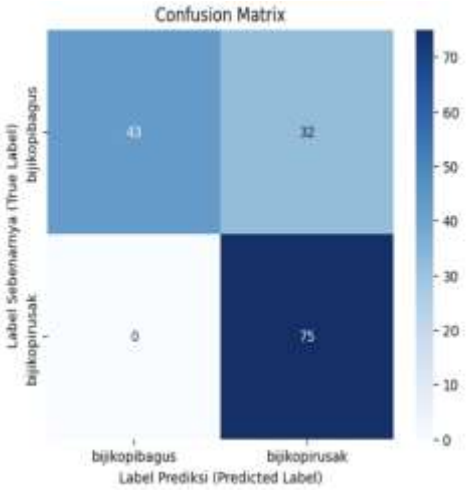
Melalui tahapan *pre-processing* yang dilakukan serta pengujian model dengan kombinasi CNN arsitertuk DenseNet-201 menggunakan optimasi *gridsearch* , model ini memberikan hasil yang lumayan optimal dengan pengklasifikasi dua kategori/kelas. Untuk kelas ‘biji bagus’ diperoleh hasi akurasi 1.00 Artinya, semua prediksi yang diklasifikasikan sebagai bijikopibagus benar (tidak ada false positive). ‘bijikopirusak’: 0.69Artinya, hanya 69% dari prediksi “bijikopirusak” yang benar, sisanya salah (termasuk biji kopi bagus yang diprediksi rusak).



Gambar 3 Matriks CNN Tanpa Optimasi

Pada confusion matrik ini bisa kita analisis hasil Model sangat baik dalam mengenali bijikopirusak (recall 100%), tetapi kurang optimal dalam mengenali bijikopibagus (hanya 56% berhasil dikenali)

Journal of Science and Social Research ISSN 2615 – 4307 (Print)
August 2025, VIII (3): 3872 – 3880 ISSN 2615 – 3262 (Online)
Available online at http://jurnal.goretanpena.com/index.php/JSSR



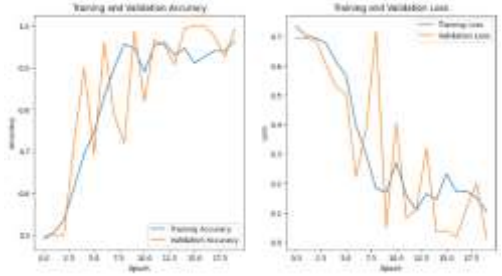
Gamabr 4 Matriks CNN Dengan Optimasi

Berdasarkan *confusion matrix*, dari total 150 data uji, model berhasil mengklasifikasikan 43 gambar *bijikopibagus* dengan benar (True Positive), namun sebanyak 32 gambar 'bijikopibagus' salah diklasifikasikan sebagai 'bijikopirusak' (False Negative). Sementara itu, seluruh 75 gambar *bijikopirusak* berhasil diklasifikasikan dengan tepat tanpa kesalahan (True Negative), dan tidak terdapat prediksi yang salah pada kelas ini (False Positive = 0).

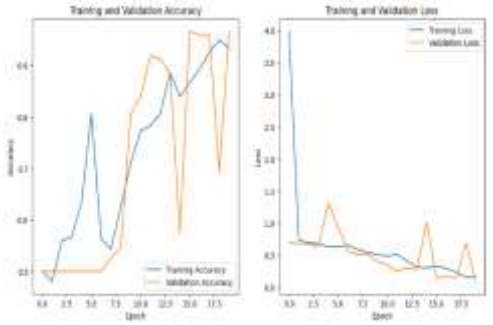
Secara keseluruhan, model menghasilkan akurasi sebesar 78,67%, yang dihitung dari jumlah klasifikasi benar dibagi total data uji, yaitu (43 + 75) / 150. Nilai precision untuk kelas *bijikopibagus* mencapai 1.00, yang berarti semua prediksi untuk kelas ini benar.

Visualisasi Hasil Bentuk Grafik

Untuk memperjelas hasil pengujian, maka visualisasi hasil bentuk grafik digunakan untuk melihat hasil Loss pada model yang diuji seperti pada gambar-gambar dibahwah ini.



Gambar 5 CNN



Gambar 6 CNN dengan optimasi

Berdasarkan analisis kedua grafik tersebut, dapat disimpulkan bahwa optimasi CNN dengan hyperparameter terbaik menghasilkan model yang jauh lebih baik dibandingkan model CNN sebelumnya.
Berikut adalah penjelasan rincinya:

Analisis Gambar 1: CNN

Grafik ini menunjukkan performa model Convolutional Neural Network (CNN) dasar sebelum dilakukan optimasi.

1. Grafik Akurasi kiri
   a) *Training Accuracy*: Akurasi pada data latih cenderung meningkat seiring berjalannya epoch, meskipun grafiknya naik-turun (tidak stabil).
   b) *Validation Accuracy*: Akurasi pada data validasi sangat fluktuatif dan tidak stabil. Grafiknya melonjak tinggi lalu turun drastis berulang kali. Ini adalah pertanda buruk. Meskipun sesekali mencapai akurasi tinggi (mendekati 1.0), performanya tidak bisa diandalkan karena pada epoch lain bisa sangat rendah.
2. Grafik loss kanan
   a) *Training Loss*: Nilai loss pada data latih cenderung menurun, yang merupakan hal yang diharapkan.
   b) *Validation Loss*: Ini adalah indikator masalah yang paling jelas. Nilai *loss* pada data validasi sangat tidak stabil. Terlihat ada lonjakan-lonjakan besar, di mana loss tiba-tiba meningkat tajam. Ketika *validation loss* meningkat sementara *training loss* terus

menurun, ini adalah gejala klasik dari overfitting.

Kesimpulan untuk Gambar 1: Model ini sangat tidak stabil dan mengalami overfitting parah. Artinya, model terlalu "menghafal" data latih sehingga tidak mampu melakukan generalisasi dengan baik pada data baru (data validasi). Performa model ini tidak dapat diandalkan di dunia nyata.

Analisis Gambar 2: CNN dengan Optimasi

Grafik ini menunjukkan performa model CNN setelah dilakukan optimasi, misalnya dengan menyesuaikan *learning rate*, jumlah *layer*, *filter*, atau menggunakan teknik seperti *dropout* dan *batch normalization*.

1. Grafik akurasi kiri
   a) *Training Accuracy*: Akurasi pada data latih meningkat secara lebih mulus dan konsisten dibandingkan Gambar 1.
   b) *Validation Accuracy*: Akurasi pada data validasi juga menunjukkan tren peningkatan yang jauh lebih stabil. Setelah epoch ke-10, akurasi validasi tetap tinggi dan konsisten, tanpa lonjakan atau penurunan drastis seperti pada Gambar 1.
2. Grafik loss kanan
   a) Training Loss: Nilai *loss* pada data latih menurun dengan mulus.
   b) Validation Loss: Nilai *loss* pada data validasi juga menurun secara konsisten dan kemudian stabil di angka yang rendah. Tidak ada lagi lonjakan-lonjakan liar. Jarak (gap) antara *training loss* dan *validation loss* juga jauh lebih kecil dan stabil, menandakan masalah overfitting telah berhasil dikurangi secara signifikan.

SIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa penerapan arsitektur DenseNet-201 dalam proses klasifikasi biji kopi memberikan hasil yang cukup baik. Model awal yang belum dioptimasi menunjukkan performa yang kurang stabil dan mengalami overfitting dengan akurasi hanya sebesar 78,67%. Namun setelah dilakukan optimasi hyperparameter menggunakan metode GridSearch, model menunjukkan peningkatan performa yang signifikan. Akurasi model meningkat menjadi 99,47% dengan nilai loss yang rendah dan stabil, yang menunjukkan kemampuan generalisasi yang baik terhadap data uji.
Optimasi parameter seperti jumlah filter konvolusi, dropout rate, dan learning rate terbukti mampu meningkatkan akurasi dan kestabilan pelatihan model. Model yang dihasilkan tidak hanya akurat, tetapi juga efisien dalam melakukan klasifikasi terhadap dua kelas biji kopi, yaitu biji kopi bagus dan biji kopi rusak. Dengan demikian, penerapan DenseNet-201 yang dioptimasi dengan GridSearch dapat dijadikan solusi efektif dan otomatis dalam proses sortasi biji kopi, khususnya bagi pelaku industri kecil dan menengah untuk meningkatkan efisiensi dan konsistensi kualitas produk.

DAFTAR PUSTAKA

R. D. Yunita, “Implementasi Metode Linear Discriminan Analysis Untuk Klasifikasi Biji Kopi,” Jurnal Teknologi Informatika dan Komputer, vol. 8, no. 1, pp. 27–39, Mar. 2022, doi: 10.37012/jtik.v8i1.664.

D. Gusmaliza and S. Aminah, “Sistem Identifikasi Kualitas Biji Kopi Robusta berbasis Image Processing dengan Support Vector Machine,” Edumatic: Jurnal Pendidikan Informatika, vol. 8, no. 2, pp. 744–753, Dec. 2024, doi: 10.29408/edumatic.v8i2.28008.

I. Alfiantama, M. I. Kresnawan, and A. P. Handoko, “Klasifikasi Tingkat Roasting Biji Kopi Dengan Metode CNN.”

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.

A. Korkmaz, T. Talan, S. Koşunalp, and T. Iliev, "Comparison of deep learning models in automatic classification of coffee bean species," PeerJ Comput Sci, vol. 11, 2025, doi: 10.7717/peerj-cs.2759.

J. Khatib Sulaiman, V. Salim, A. Abdullah, P. Yuli Utami, U. Muhammadiyah Pontianak, and K. Kunci, "Klasifikasi Citra Penyakit Tanaman Pada Daun Paprika Dengan Metode Transfer Learning Menggunakan DenseNet-201," Indonesian Journal of Computer Science Attribution, vol. 13, no. 2, pp. 2024–3001.

R. Dwi Yulian Prakoso, B. Soejono Wiriaatmadja, and F. Wahyu Wibowo, "Sistem Klasifikasi Pada Penyakit Parkinson Dengan Menggunakan Metode K-Nearest Neighbor," Seminar Nasional Teknologi Komputer & Sains (SAINTEKS) , no. 2016, pp. 63–68, 2020.

B. E. Cahyono, A. T. Nugroho, and I. W. Maulinida, "Klasifikasi Jenis Biji Kopi dengan Menggunakan Metode Gray Level Co-occurrence Matrix (GLCM)," TEKNOTAN, vol. 16, no. 3, p. 191, Mar. 2023, doi: 10.24198/jt.vol16n3.9.

I. Irmawan, R. Rendiansyah, G. Gustini, and S. A. Harahap, "IDENTIFIKASI DAN KLASIFIKASI PENYAKIT PADA TANAMAN KOPI ARABICA DENGAN METODE CNN DAN TRANSFER LEARNING DENSENET-201," Transmisi: Jurnal Ilmiah Teknik Elektro, vol. 27, no. 2, pp. 83–91, Apr. 2025, doi: 10.14710/transmisi.27.2.83-91.

A. Michael, J. Rusman, and A. History, "Jurnal Teknologi dan Manajemen Informatika Klasifikasi Cacat Biji Kopi Menggunakan Metode Transfer Learning dengan Hyperparameter Tuning Gridsearch Article Info ABSTRACT," vol. 9, no. 1, pp. 37–45, 2023, [Online]. Available: http://http://jurnal.unmer.ac.id/index.php/jtmi

A. Toha, P. Purwono, and W. Gata, "Model Prediksi Kualitas Udara dengan Support Vector Machines dengan Optimasi Hyperparameter GridSearch CV," Buletin Ilmiah Sarjana Teknik Elektro, vol. 4, no. 1, pp. 12–21, Apr. 2022, doi: 10.12928/biste.v4i1.6079.