

Penerapan Convolutional Neural Network Dan DenseNet121 untuk Identifikasi Penyakit Daun Jagung Di Daerah Toba

Oppir Hutapea^{1*}, Ford Lumban Gaol², Takuro Matsuo³

¹ Fakultas Vokasi, Sarjana Terapan Rekayasa Perangkat Lunak, Institut Teknologi Del, Laguboti, Indonesia

² Computer Science Department, Doctor of Computer Science, Bina Nusantara University, Jakarta, Indonesia

³ Advanced Institute of Industrial Technology, Tokyo, Japan

Email: ¹oppir.hutapea@del.ac.id, ²fgaol@binus.edu, ³matsuo@aiit.ac.jp

Email Penulis Korespondensi: oppir.hutapea@del.ac.id

Submitted 22-05-2025; Accepted 17-07-2025; Published 14-08-2025

Abstrak

Jagung adalah salah satu produk pertanian penting di wilayah Toba, Sumatera Utara. Akan tetapi, hasil tanaman jagung seringkali berkurang karena gangguan penyakit yang menyerang daun sebelum panen. Tiga jenis penyakit daun yang paling sering dijumpai adalah bercak daun, penyakit hawar, dan karat daun. Untuk mendukung petani dalam melakukan deteksi awal terhadap penyakit tersebut, penelitian ini merancang sistem klasifikasi berbasis citra yang memanfaatkan algoritma Convolutional Neural Network (CNN) dan model DenseNet121 sebagai metode transfer learning. Penelitian ini bertujuan untuk secara otomatis mengidentifikasi jenis penyakit pada daun jagung melalui citra, sehingga petani dapat segera mengambil tindakan penanggulangan yang sesuai. Eksperimen dilaksanakan dengan membandingkan beragam konfigurasi model, termasuk pemanfaatan fungsi aktivasi (ReLU dan Tanh), penyesuaian laju pembelajaran, serta penyesuaian hyperparameter lainnya seperti optimizer dan metode preprocessing (normalisasi, augmentasi rotasi, zoom, dan kontras). Temuan menunjukkan bahwa penerapan DenseNet121 dengan tingkat pembelajaran optimal 0,001 menghasilkan akurasi maksimum mencapai 97%, mengungguli model CNN yang dirancang khusus dengan akurasi 96%. Komposisi preprocessing dan penyesuaian parameter memberikan kontribusi besar terhadap peningkatan kinerja model. Studi ini mengungkapkan kemungkinan penerapan teknologi deteksi penyakit tanaman berbasis citra dalam bidang pertanian, terutama untuk membantu pengambilan keputusan petani secara langsung, meningkatkan efisiensi pengelolaan lahan, dan mendukung peningkatan hasil panen jagung.

Kata Kunci: CNN; DenseNet121; Deteksi Penyakit Tanaman; Hyperparameter Tuning; Preprocessing Citra

Abstract

Corn is one of the most important agricultural commodities in the Toba region of North Sumatra. However, its productivity is often reduced due to foliar diseases that appear prior to harvest. The three most commonly observed leaf diseases include leaf spot, blight, and rust. To support early detection efforts among local farmers, this study proposes an image-based classification system employing the Convolutional Neural Network (CNN) algorithm and the DenseNet121 model as a transfer learning approach. The primary objective of this research is to automatically identify the type of disease affecting corn leaves using image data, thereby enabling farmers to promptly implement appropriate countermeasures. A series of experiments were conducted to evaluate various model configurations, including different activation functions (ReLU and Tanh), adjustments to learning rates, and the tuning of other hyperparameters such as optimizers and preprocessing methods (normalization, rotation augmentation, zooming, and contrast adjustments). The results demonstrate that DenseNet121, when trained with an optimal learning rate of 0.001, achieved the highest accuracy of 97%, outperforming the custom-built CNN model which attained an accuracy of 95%. The combination of effective preprocessing techniques and hyperparameter tuning significantly contributed to the improved performance of the models. This study highlights the potential of image-based plant disease detection technologies in agriculture, particularly in aiding real-time decision-making, enhancing land management efficiency, and supporting increased corn yield.

Keywords: CNN; DenseNet121; Deases Detection; Hyperparameter; Image Preprocessing;

1. PENDAHULUAN

Jagung (*Zea mays*) merupakan salah satu tanaman sereal penting yang berperan besar sebagai sumber karbohidrat utama, serta menduduki peringkat ketiga tanaman pangan dunia setelah beras dan gandum. Di Indonesia, jagung merupakan tanaman pangan kedua terbesar setelah padi, dengan kontribusi yang signifikan terhadap ketahanan pangan nasional dan sebagai bahan baku pakan ternak [1][2].

Namun, produktivitas tanaman jagung di Indonesia, termasuk di wilayah Toba, Sumatera Utara, kerap mengalami penurunan akibat serangan penyakit daun sebelum masa panen. Tiga jenis penyakit daun yang sering dijumpai oleh petani adalah bercak daun, hawar daun, dan karat daun [3]. Deteksi penyakit yang selama ini dilakukan secara manual oleh petani memiliki berbagai keterbatasan, seperti ketergantungan pada pengalaman visual, waktu yang lama, dan potensi kesalahan manusia [4]. Oleh karena itu, pendekatan berbasis teknologi informasi, khususnya dalam ranah kecerdasan buatan (AI), menjadi alternatif solusi yang menjanjikan untuk mendukung sistem pertanian modern. Salah satu pendekatan dalam klasifikasi penyakit tanaman berbasis citra adalah menggunakan algoritma Convolutional Neural Network (CNN), sebuah arsitektur Deep Learning yang efektif untuk pengenalan pola visual dan klasifikasi gambar [5][6]. CNN mampu mengenali dan mengekstrak fitur dari gambar secara otomatis, serta membedakan objek berdasarkan pola spasialnya tanpa memerlukan proses ekstraksi fitur manual.

Indonesia merupakan negara agraris, di mana sektor pertanian menjadi sumber utama mata pencaharian bagi sebagian besar penduduknya. Di antara berbagai aspek pertanian, produksi tanaman merupakan salah satu kegiatan yang paling penting dan tersebar luas. Namun, tantangan paling krusial yang dihadapi para petani saat ini adalah ancaman

penyakit tanaman, yang secara signifikan memengaruhi kesehatan dan produktivitas tanaman. Penyakit-penyakit ini tidak hanya menghambat pertumbuhan tanaman, tetapi juga menurunkan kualitas dan kuantitas hasil pertanian, yang pada akhirnya dapat menyebabkan kerugian ekonomi yang besar serta mengancam ketahanan pangan [7]. Keberadaan penyakit tanaman ini disebabkan oleh berbagai faktor biotik, terutama hama dan patogen seperti jamur, bakteri, virus, dan nematoda. Apabila tidak terdeteksi dan ditangani secara tepat waktu, agen penyebab tersebut dapat menyebar dengan cepat, menginfeksi seluruh lahan pertanian, dan memberikan dampak yang merugikan bagi komunitas petani. Oleh karena itu, deteksi dini dan identifikasi penyakit tanaman secara akurat menjadi sangat penting dalam mewujudkan praktik pertanian yang berkelanjutan serta menjaga stabilitas produksi pangan [8].

Beberapa studi sebelumnya telah menerapkan CNN untuk klasifikasi penyakit pada tanaman. Misalnya, patologi daun tomat dengan CNN dan Learning Vector Quantization (LVQ) menunjukkan akurasi tinggi dalam mengidentifikasi bercak bakteri, hawar daun, dan daun menguning [9][10]. Studi lainnya menggunakan AlexNet dan VGG16 untuk klasifikasi penyakit tanaman padi dan berhasil mencapai akurasi di atas 90%. Namun, studi-studi tersebut sebagian besar masih berfokus pada jenis tanaman tertentu dan belum banyak yang mengkaji klasifikasi penyakit daun jagung secara spesifik dengan pendekatan transfer learning [11].

Gap dari penelitian ini adalah belum adanya kajian yang secara spesifik membandingkan efektivitas model CNN kustom dengan arsitektur DenseNet121 untuk klasifikasi penyakit daun jagung dalam konteks lokal, khususnya wilayah Toba. DenseNet121, sebagai salah satu model CNN canggih, dikenal memiliki keunggulan dalam efisiensi parameter dan aliran informasi yang lebih optimal antar lapisan jaringan [12][13]. Penelitian ini bertujuan untuk mengembangkan dan membandingkan dua pendekatan CNN, yaitu CNN kustom dan DenseNet121, dalam mengklasifikasikan empat jenis kondisi daun jagung: bercak daun, hawar daun, karat daun, dan daun sehat. Model dilatih dengan berbagai konfigurasi fungsi aktivasi (ReLU dan Tanh), penyesuaian learning rate, serta teknik preprocessing seperti normalisasi dan augmentasi citra (rotasi, zoom, kontras) [14]. Selain itu, digunakan teknik clustering K-Means untuk menghitung area terdampak penyakit guna memberikan informasi tambahan bagi pengguna dalam pengambilan keputusan [15].

Kontribusi utama dari penelitian ini adalah Pengembangan sistem klasifikasi penyakit daun jagung berbasis CNN dan DenseNet121. Evaluasi komprehensif terhadap pengaruh fungsi aktivasi, learning rate, dan teknik preprocessing terhadap performa model. Penerapan metode segmentasi area terdampak menggunakan K-Means untuk mendukung pengambilan keputusan dalam penanganan penyakit. Penerapan model dalam konteks lokal wilayah pertanian Toba sebagai upaya integrasi teknologi dalam pertanian presisi. Penelitian ini diharapkan dapat menjadi langkah awal menuju sistem pendeteksi penyakit tanaman berbasis teknologi real-time yang mampu meningkatkan efisiensi pertanian, mengurangi kerugian akibat penyakit, dan mendukung pertanian berkelanjutan. Berikut ini adalah gambar yang memberikan wawasan visual terhadap jenis penyakit daun jagung (1,2,3) dibandingkan dengan gambar daun sehat (4) seperti dibawah ini [16][17].



Gambar 1. Bercak Daun



Gambar 2. Daun Hawar



Gambar 3. Karat Daun



Gambar 4. Daun Sehat

2. METODOLOGI PENELITIAN

2.1. Identifikasi Masalah

Penelitian ini bertujuan untuk mengembangkan sebuah model klasifikasi citra dalam mengidentifikasi jenis penyakit pada daun jagung menggunakan pendekatan CNN (*Convolutional Neural Network*) dan DenseNet121. Solusi yang akan diciptakan dalam penelitian ini adalah bagaimana membedakan daun jagung yang sehat terhadap tiga jenis penyakit utamanya yaitu bercak daun, karat daun dan hawar daun berdasarkan visualisasi gambar yang dihasilkan. Kategori daun menjadi hal yang ditargetkan sehingga klasifikasi yang dimaksud adalah banyak kelas penyakit jagung (*multiclass image classification*). Dataset yang terdiri dari berbagai penyakit jagung sesuai dengan penyakitnya dilakukan pelabelan sehingga diharapkan dapat mempelajari masing-masing citra dari daun-daun tersebut. Dengan tujuan penjaminan kualitas data yang optimal maka terdapat beberapa langkah yang dilakukan yaitu pembersihan data (*cleansing*), normalisasi (*normalization*) dan augmentasi data (*augmentation*). Dalam pembagian dataset yang telah disediakan ditentukan menjadi 70% data pelatihan (*training*), 15% data validasi (*validation set*) dan 15% data testing (*test set*). Beberapa matriks evaluasi yang dilakukan yaitu Akurasi (*Accuration*), Presisi (*Precision*), Recall dan F1-Score. Untuk memastikan data baru terukur dengan baik sesuai dengan ketentuan yang telah dijalankan maka diterapkan *k-fold cross-validation* sehingga performa model menjadi lebih general (data baru menjadi lebih terukur) [18][19].

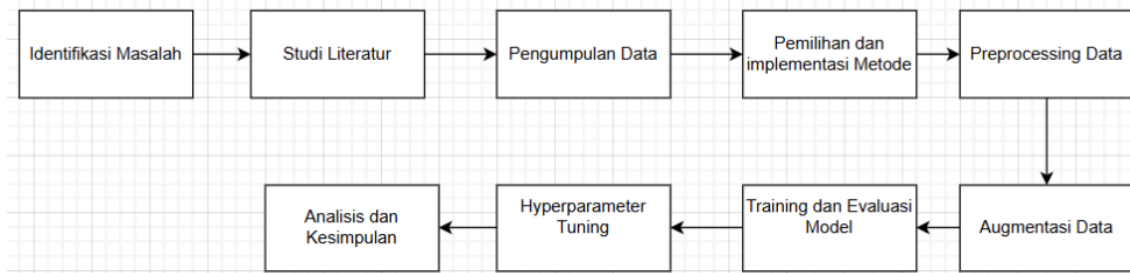
Proses pelatihan menggunakan fungsi kehilangan (*loss function*) *categorical cross-entropy*, dengan optimasi dilakukan melalui algoritma Adam dan penyesuaian *learning rate* serta *dropout* untuk mencegah *overfitting*. Teknik *hyperparameter tuning* dilakukan melalui *grid search* untuk menemukan konfigurasi terbaik [20].

2.2. Tahapan Penelitian

Terdapat beberapa tahapan penelitian yang disusun sebagai panduan dalam pelaksanaan penelitian ini yaitu:

- Pengumpulan data adalah langkah awal yang sangat krusial dalam penelitian ini. Kumpulan data yang digunakan mencakup gambar daun jagung dengan empat kategori: bercak daun, hawar daun, karat daun, dan daun yang sehat. Foto-foto ini didapatkan melalui pengamatan langsung di lokasi dan juga dari sumber dataset terbuka yang telah ada di internet. Variasi sumber data ini mendukung peningkatan representasi keadaan nyata penyakit jagung.
- Tahap preprocessing bertujuan untuk mempersiapkan gambar agar sesuai dengan format masukan model CNN. Proses ini meliputi perubahan ukuran gambar menjadi seragam, penyesuaian nilai piksel agar berada dalam rentang spesifik, serta konversi ke format tensor sebagai syarat input untuk model deep learning. Augmentasi juga dilakukan untuk menambah variasi data, termasuk rotasi acak, penyesuaian brightness, contrast, hue, exposure, serta penerapan teknik histogram equalization dan median filtering untuk memperbaiki kualitas serta ketahanan model terhadap noise.
- Pada fase ini, dilakukan penerapan dua metode: CNN Kustom dan Pembelajaran Transfer dengan menggunakan arsitektur DenseNet121. CNN Custom dirancang dari nol dengan struktur yang disesuaikan untuk mengidentifikasi fitur pada gambar daun. Sementara itu, Transfer Learning menggunakan model DenseNet121 yang telah dilatih sebelumnya pada dataset besar (seperti ImageNet), kemudian disesuaikan (*fine-tune*) untuk klasifikasi penyakit pada daun jagung. Pendekatan ini bertujuan untuk membandingkan kinerja antara model yang dibuat sendiri dan model yang telah dilatih sebelumnya.
- Model dilatih dengan memanfaatkan fungsi aktivasi ReLU dan Tanh untuk menciptakan representasi non-linear yang rumit. Berbagai pengaturan hyperparameter diuji, termasuk perubahan learning rate, untuk menemukan kombinasi optimal yang mempercepat konvergensi tanpa mengurangi akurasi. Dropout diterapkan dalam desain jaringan untuk mencegah *overfitting*, yang merupakan situasi dimana model tampil sangat baik pada data pelatihan tetapi buruk pada data baru (data uji). Proses validasi dilakukan dengan mengawasi kinerja model selama pelatihan guna menghindari pelatihan berlebihan.
- Model dievaluasi dengan memanfaatkan berbagai metrik kinerja: akurasi, presisi, recall, serta F1-score. Akurasi menilai seberapa sering model menghasilkan prediksi yang tepat. Presisi dan recall digunakan untuk menganalisis kemampuan model dalam mengidentifikasi masing-masing kelas penyakit secara rinci. F1-score menyediakan rata-rata harmonis antara presisi dan recall, sangat bermanfaat ketika distribusi kelas tidak seimbang. Penilaian ini berfungsi untuk mengidentifikasi model yang paling terpercaya untuk diterapkan secara lebih luas.
- Model paling baik yang diperoleh dari proses pelatihan dan validasi selanjutnya diterapkan pada data uji untuk mengevaluasi kinerjanya yang sesungguhnya. Analisis hasil prediksi dilakukan dengan pendekatan kualitatif dan kuantitatif untuk mengevaluasi kekuatan serta kelemahan model. Perbandingan dilakukan antara kedua metode (CNN Kustom dan DenseNet121) untuk menilai model mana yang paling tepat, konsisten, dan efisien dalam mengidentifikasi jenis penyakit pada daun jagung.
- Evaluasi model divisualisasikan untuk memberikan pemahaman yang lebih jelas. Matriks kebingungan ditampilkan untuk mengamati klasifikasi yang benar dan salah dari setiap kelas. Grafik akurasi dan loss selama proses pelatihan juga ditampilkan untuk memperlihatkan pola pembelajaran model. Prediksi divisualisasikan dalam bentuk gambar daun yang disertai label prediksi dan label actual, untuk menunjukkan kinerja model secara visual pada situasi nyata.

Secara garis besar tahapan penelitian yang telah dijelaskan pada poin satu sampai dengan tujuh dapat diringkas menjadi bentuk alur penelitian seperti gambar 5 dibawah ini:



Gambar 5. Tahapan Penelitian

2.3. Image Classification

Proses mengidentifikasi dan mengelompokkan gambar ke dalam kategori tertentu berdasarkan karakteristik visual, seperti pola, warna, tekstur, atau elemen lain yang mencerminkan data visual. Pada proyek ini, klasifikasi gambar dilakukan menggunakan *model Convolutional Neural Network (CNN)*, termasuk arsitektur CNN Custom dan *Transfer Learning DenseNet121* [13], [15]. Model ini dilatih dengan dataset gambar yang telah diproses untuk mengekstraksi pola penting yang membedakan kategori satu dengan lainnya, seperti bentuk, tekstur, dan warna. Dalam konteks proyek, klasifikasi diterapkan untuk mengenali jenis daun mangga dan mendeteksi penyakitnya berdasarkan ciri visual unik. Proses ini lebih efektif dibandingkan metode manual karena model secara otomatis mempelajari fitur signifikan dari gambar tanpa perlu rekayasa fitur manual. Selain itu, tahapan image processing dilakukan untuk memastikan gambar dipersiapkan sebagai input optimal bagi model CNN, sehingga meningkatkan akurasi dan efisiensi pengenalan. Klasifikasi citra adalah proses pengenalan dan pengelompokan gambar berdasarkan fitur visual seperti bentuk, warna, dan terstruktur [21]. CNN dapat mengekstraksi dan mengenali fitur-fitur penting secara otomatis, berbeda dengan pendekatan konvensional yang membutuhkan rekayasa fitur manual.

Model CNN Custom dibangun dari awal, sementara DenseNet121 digunakan sebagai arsitektur transfer learning. DenseNet121 memungkinkan *feature reuse* melalui koneksi antar lapisan yang padat, meningkatkan efisiensi representasi citra [22]. Gambar diproses sebagai input dengan ukuran 640x640 piksel, kemudian dinormalisasi agar sesuai dengan standar model.

2.4. Arsitektur Model

CNN custom dan DenseNet121 menjadi solusi dalam perancangan arsitektur penelitian ini dimana keduanya adalah pendekatan jaringan konvolusional. Keduanya dirancang untuk mendeteksi dan mengklasifikasi penyakit daun jagung berdasarkan ciri visual citra.

a. CNN Custom

Arsitektur CNN Custom dirancang dari awal dengan mempertimbangkan efisiensi dan kemampuan ekstraksi fitur yang baik. Struktur model ini terdiri dari beberapa komponen utama:

1. Lapisan Konvolusi dan Aktivasi:

Model menggunakan filter konvolusi kecil berukuran 3×3 yang berfungsi untuk mengekstraksi fitur spasial dari gambar secara mendalam dan rinci. Lapisan ini efektif dalam mengenali pola-pola dasar seperti tepi, sudut, dan garis, yang kemudian diproses pada lapisan berikutnya untuk menangkap pola yang lebih kompleks dan terstruktur.

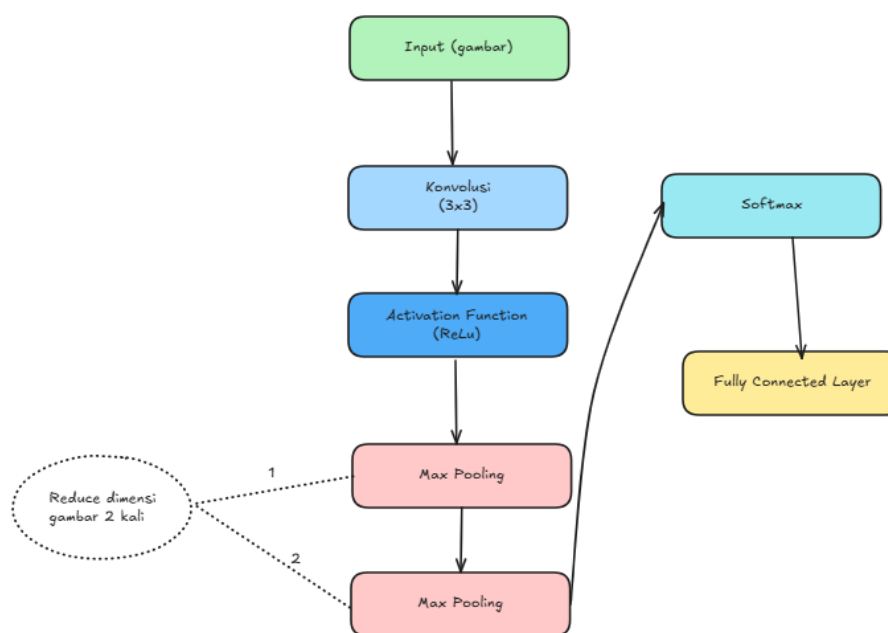
2. Pooling Layer (Max Pooling):

Untuk mengurangi dimensi dari *feature map*, digunakan teknik *Max Pooling*. Pendekatan ini tidak hanya menurunkan kompleksitas komputasi, tetapi juga membantu mengurangi risiko *overfitting* tanpa kehilangan informasi penting dari fitur yang telah diekstraksi artinya mempertahankan fitur yang lebih tidak akan mengurangi nilai dari matriks yang dikelola.

3. Lapisan Fully Connected:

Setelah fitur diekstraksi (fitur penting dalam gambar) dan direduksi (pengurangan dimensi matriks), jaringan menggunakan beberapa lapisan *fully connected* untuk menghasilkan representasi numerik. Representasi ini digunakan dalam proses klasifikasi akhir menggunakan fungsi aktivasi *softmax* untuk menentukan jenis penyakit daun jagung yang terdeteksi.

Sehingga design yang diterapkan dalam perancangan model CNN Custom adalah sebagai berikut :



Gambar 6. Desain CNN

b. DenseNet121 (Transfer Learning)

Arsitektur DenseNet121 merupakan model *transfer learning* yang telah terbukti efektif dalam berbagai tugas klasifikasi citra. Model ini menawarkan efisiensi dan akurasi tinggi dengan jumlah parameter yang relatif rendah. Komponen penting dalam arsitektur ini adalah sebagai berikut:

1. Dense Connectivity:

DenseNet121 mengadopsi pendekatan *Dense Connectivity*, di mana setiap lapisan menerima masukan tidak hanya dari lapisan sebelumnya, tetapi juga dari semua lapisan sebelumnya. Mekanisme ini memungkinkan penggunaan fitur kembali, memperkuat propagasi gradien, mengurangi redundansi, dan meningkatkan efisiensi dalam menangkap pola yang kompleks secara terstruktur.

2. Transition Layer dan Dense Block:

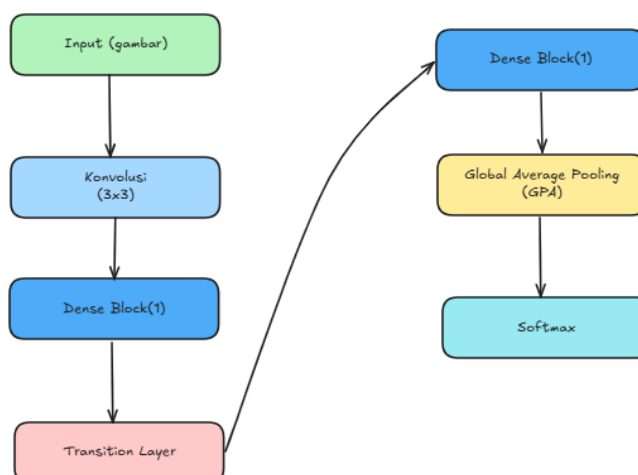
Arsitektur terdiri dari *dense block* yang dihubungkan dengan *transition layer*. *Dense block* berisi beberapa lapisan konvolusi yang saling terhubung, sedangkan *transition layer* berfungsi untuk mengatur dimensi dan kompleksitas jaringan.

3. Pooling Layer (Global Average Pooling):

Berbeda dari CNN Custom yang menggunakan *max pooling* dan *fully connected* secara intensif, DenseNet121 menggunakan GAP (*global average pooling*) di lapisan akhir untuk menggantikan banyaknya lapisan *fully connected*. GAP memberikan representasi yang lebih ringkas dan efisien, serta mengurangi jumlah parameter dalam model.

4. Klasifikasi Akhir (Softmax):

Setelah fitur diringkas melalui GAP, hasilnya diproses oleh *softmax layer* untuk melakukan klasifikasi ke dalam kategori penyakit daun. Proses ini memberikan hasil klasifikasi yang akurat dengan kompleksitas yang lebih rendah dibandingkan arsitektur tradisional.



Gambar 7. Desain DenseNet

2.5. Image Processing

Proses pengolahan gambar sebelum masuk ke CNN meliputi *Resizing* yaitu menyesuaikan ukuran gambar menjadi 640x640 piksel. Normalisasi untuk Skala piksel 0–1. Cropping untuk Horizontal 25%–75%, vertikal 27%–75%. Data Augmentation yang dilakukan seperti Flip horizontal, perubahan hue, saturasi, brightness, exposure, rotasi 90°, histogram equalization, dan median filtering. Teknik augmentasi meningkatkan variasi data dan membantu mengurangi overfitting [24].

2.6. Hyperparameter Tuning

Hyperparameter Tuning adalah proses yang penting dalam pengembangan model *machine learning* untuk meningkatkan performa model dengan menyesuaikan berbagai *hyperparameter* yang mempengaruhi proses pelatihan. *Hyperparameter* merujuk pada parameter yang ditetapkan sebelum pelatihan dimulai dan tidak dipelajari selama proses pelatihan, seperti learning rate, jumlah lapisan konvolusi, jumlah filter per lapisan, ukuran batch, dan dropout. Menyesuaikan hyperparameter ini bertujuan untuk menemukan kombinasi optimal yang dapat meningkatkan akurasi model, mengurangi overfitting, dan mempercepat proses pelatihan. Tujuan Hyperparameter Tuning

- Meningkatkan akurasi model Mengoptimalkan kombinasi hyperparameter untuk meningkatkan performa pada dataset validasi atau testing, menghasilkan prediksi yang lebih akurat.
- Menghindari overfitting dan underfitting Menyesuaikan hyperparameter agar model dapat menggeneralisasi pola dengan baik pada data yang belum terlihat.
- Mengotomatiskan eksperimen Menggunakan algoritma pencarian seperti Grid Search, Random Search, atau Hyperband untuk mempercepat pencarian kombinasi hyperparameter terbaik.

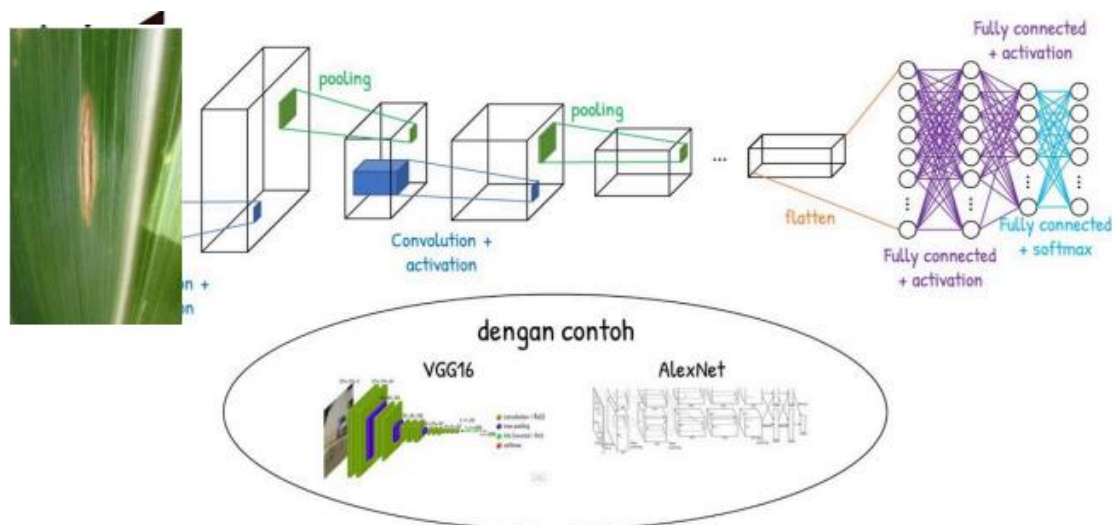
2.7. Evaluasi

Evaluasi dilakukan dengan beberapa hal yaitu Akurasi untuk perbedaan prediksi benar terhadap total prediksi. Presisi untuk proporsi prediksi positif yang benar. Recall untuk kemampuan model mengenali seluruh instance kelas positif. F1-Score untuk Harmonik antara presisi dan recall. Metode confusion matrix juga digunakan untuk menggambarkan performa klasifikasi tiap kelas penyakit daun [27].

3. HASIL DAN PEMBAHASAN

Sesuai dengan jumlah kelas penyakit daun jagung yang diklasifikasikan pada penelitian ini, digunakan 4 jenis dataset, yaitu Hawar, Karat, Bercak, dan Sehat. Masing-masing kelas dataset tersebut dimasukkan ke dalam direktori test, train, dan valid. Tujuannya agar model yang dilatih dan dikembangkan dapat mengetahui letak dari data latih, data validasi, dan data test yang digunakan oleh model. Dataset tersebut juga sudah termasuk dari hasil augmentasi yang dilakukan sebelum masuk ke kode implementasi. Pada kode implementasi, dilakukan juga *augmentasi*, *zoom*, *shear*, dan *rotation*.

Desain arsitektur yang dikembangkan dalam penelitian ini mengacu pada arsitektur neural network seperti gambar dibawah ini yang mengilustrasikan proses gambar di input sampai menjadi data flatten dijadikan menjadi input neural network, setiap proses filtering dilakukan perhitungan masing-masing neuron yang masuk dan menghasilkan neural yang menjadi input ke neuron berikutnya :



Gambar 8. Gambaran Desain Arsitektur

Sesuai dengan jumlah kelas penyakit daun jagung yang diklasifikasikan pada penelitian ini digunakan 4 jenis daun jagung yaitu Hawar, Karat, Bercak dan Sehat. Terdapat jumlah yang cukup balance antar masing-masing kelas. Setiap

Tabel 1. Pengelompokan Dataset

Data Train	
Bercak	990
Hawar	1158
Karat	1173
Sehat	1151
Data Test	
Bercak	111
Hawar	128
Karat	130
Sehat	148
Data Validasi	
Bercak	110
Hawar	128
Karat	130
Sehat	147

3.1 Augmentasi Data

Berikut adalah rincian dari augmentasi yang dipakai oleh penelitian flip vertical dan horizontal dimana Hue Antara -25 dan +25 (derajat), Saturation Antara -25% dan +25%, Brightness Antara -25% dan +25%, Exposure Antara -15% dan +15%. Berikut ini adalah hasil detail arsitektur dalam CNN yang digunakan. Selanjutnya, dalam kode implementasi dilakukan penambahan teknik augmentasi data guna meningkatkan variasi citra latih dan memperkuat generalisasi model. Augmentasi tersebut meliputi pengaturan parameter zoom_range sebesar 0.2, yang berfungsi untuk mengontrol tingkat pembesaran atau pengecilan citra hingga 20% dari ukuran aslinya. Selain itu, diterapkan pula shear_range dengan nilai 0.2, yang memungkinkan dilakukannya transformasi shear atau kemiringan terhadap citra hingga sebesar 20%, sehingga menghasilkan perubahan sudut pandang secara linier. Adapun rotation_range disetel pada nilai 30, yang memberikan fleksibilitas bagi citra untuk mengalami rotasi hingga 30 derajat, baik searah maupun berlawanan arah jarum jam. Penyesuaian parameter-parameter ini bertujuan untuk mensimulasikan berbagai kemungkinan variasi citra di dunia nyata, sehingga dapat meningkatkan robustnes model dalam menghadapi data yang tidak terstruktur secara seragam. Dapat dilihat pada pembentukan layer yang digunakan dilengkapi dengan jumlah lapisannya pada tabel

Tabel 2. Layer Network

Layer(type)	Output Shape	Param #
Conv2d (Conv2D)	(None, 256, 256, 32)	896
Batch_normalization (BatchNormalization)	(None, 256, 256, 32)	128
max_pooling2d(MaxPooling2D)	(None, 128, 128, 32)	0
conv2d_1(Conv2D)	(None, 128, 128, 64)	18.496
batch_normalization_1(BatchNormalization)	(None, 128, 128, 64)	256
max_pooling2d_1(MaxPooling2D)	(None, 64, 64, 64)	0
conv2d_2(Conv2D)	(None, 64, 64, 128)	73.856
batch_normalization_2(BatchNormalization)	(None, 64, 64, 128)	512
max_pooling2d_2(MaxPooling2D)	(None, 32, 32, 128)	0
conv2d_3(Conv2D)	(None, 32, 32, 256)	295.168
batch_normalization_3(BatchNormalization)	(None, 32, 32, 256)	1.024
max_pooling2d_3(MaxPooling2D)	(None, 16, 16, 256)	0
global_average_pooling2d(GlobalAveragePooling)	(None, 256)	0
Flatten(Flatten)	(None, 256)	0
dropout(Dropout)	(None, 256)	0
dense(Dense)	(None, 512)	131.584
dropout_1(Dropout)	(None, 512)	0
dense_1(Dense)	(None, 128)	65.664
dense_2(Dense)	(None, 4)	516

3.2 Sebelum Hyperparameter Tuning

Pada penelitian ini, digunakan 3 teknik dalam melakukan hyperparameter tuning untuk masing-masing model CNN maupun DenseNet121, yaitu menggunakan teknik Grid Search, Random Search, dan Hyperband. Masing-masing teknik mempunyai kombinasi parameter yang dicoba. Berikut adalah rincian hyperparameter tuning pada masing-masing model:

a. Grid Search

Kombinasi parameter yang dicoba pada teknik ini adalah learning rate dan optimizer. Terdapat 4 kali trial/percobaan pada teknik ini. Berikut adalah rinciannya :

Tabel 3. Grid Search Kombinasi

Trial Ke-	Optimizer	Learning Rate	Validation Accuracy	Validation Loss
1	Adam	0.001	0.9747	0.0618
2	Adamax	0.001	0.9825	0.0707
3	Adam	0.0005	0.9844	0.0469
4	Adamax	0.0005	0.97281	0.0985

b. Random Search

Kombinasi parameter yang dicoba pada teknik ini adalah dropout rate, fungsi aktivasi, learning rate, dan optimizer. Terdapat 5 kali trial/percobaan pada teknik ini. Berikut adalah rinciannya :

Tabel 4. Random Search Kombinasi

Trial ke-	Optimizer	Learning Rate	Dropout Rate	Activation Function	Validation Accuracy	Validation Loss
1	Adamax	0.0005	0.3	ReLu	0.9669	0.100
2	Adamax	0.0001	0.4	Tanh	0.9320	0.251
3	Sgd	0.001	0.3	ReLu	0.9087	0.298
4	Sgd	0.0001	0.4	Tanh	0.4349	1.335
5	Sgd	0.0001	0.5	ReLu	0.4621	1.220

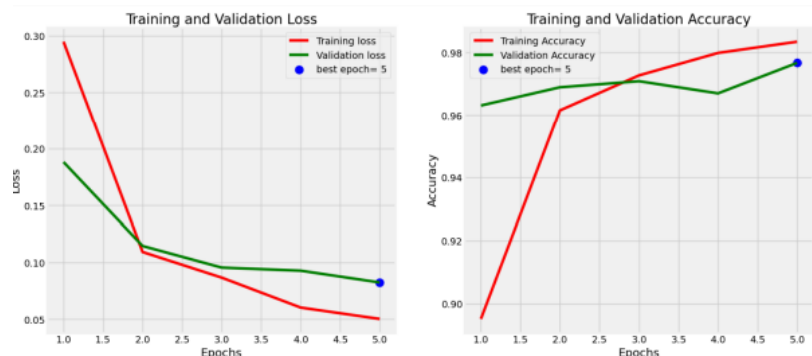
c. Hyperband

Kombinasi parameter yang dicoba pada teknik ini adalah dense unit, fungsi aktivasi, dropout rate, epoch, dan optimizer. Terdapat 18 kali trial/percobaan pada teknik ini. Berikut adalah rinciannya :

Tabel 5. Hyperband Kombinasi

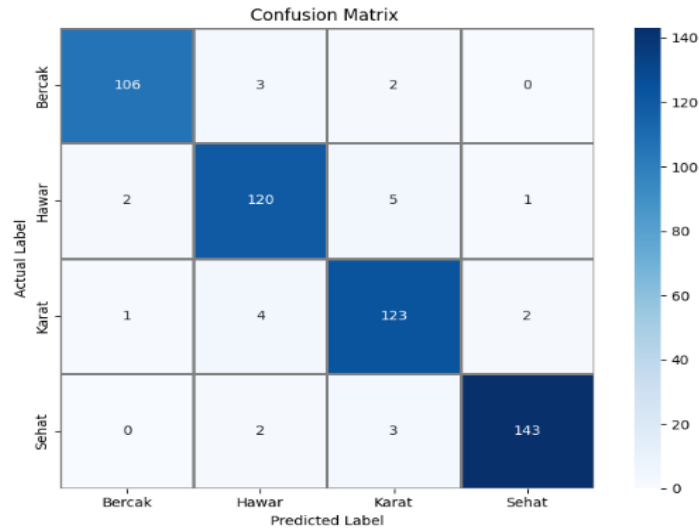
Trial Ke -	Dense Unit	Activation Function	Dropout Rate	Epoch	Optimizer	Validation Accuracy
1	128	Tanh	0.3	2	Adam	0.9766
2	128	Relu	0.4	2	Sgd	0.9165
3	128	Relu	0.3	2	Adamax	0.9631
4	512	Relu	0.3	2	Sgd	0.9456
5	512	Tanh	0.5	2	Adamax	0.9631
6	512	Tanh	0.5	2	Sgd	0.9456
7	512	Tanh	0.4	2	Adam	0.9708
8	256	Tanh	0.4	2	Adamax	0.9669
9	512	Relu	0.3	2	Adamax	0.9766
10	256	Relu	0.4	2	Adam	0.9728
11	512	Tanh	0.3	2	Sgd	0.9475
12	256	Relu	0.5	2	Sgd	0.9320
13	128	Tanh	0.3	4	Adam	0.9786
14	512	Relu	0.3	4	Adamax	0.9805
15	256	Relu	0.4	4	Adam	0.9766
16	512	Tanh	0.4	4	Adam	0.9825
17	512	Tanh	0.4	10	Adam	0.9864
18	512	Relu	0.3	10	Adamax	0.9922

Maka dari data tersebut menghasilkan nilai grafik seperti gambar dibawah ini yang menjelaskan training loss sama-sama mengalami penurunan dimana proses learningnya berjalan dengan baik sementara pada training dan validasi untuk akurasi mengalami peningkatan yang linear walaupun ada titik dimana training akurasi meningkat tetapi validasi akurasi berkurang, tetapi ini sangatlah kecil dilihat dari visualisasi yang dihasilkan pada gambar tersebut.



Gambar 9. Loss dan Accuracy pada Training

Berikut adalah visualisasi confusion matrix berdasarkan data uji yang dilakukan dan hasil prediksi. Warna biru yang lebih gelap menunjukkan jumlah prediksi yang lebih tinggi. Diagonal (kiri atas ke kanan bawah) menunjukkan jumlah prediksi yang benar per kelas. Sel di luar diagonal menunjukkan kesalahan klasifikasi antar kelas. Percobaan data Bercak daun bahwa diprediksi benar 106 dan sisanya diprediksi salah, Hawar daun diprediksi benar 120 sementara 8 diantaranya diprediksi salah. Sementara Karat daun diprediksi benar sebanyak 123 dan diprediksi salah 7, dan yang terakhir adalah kelas sehat diprediksi benar 143 dan jumlah yang tidak tepat diprediksi adalah 5. Hal ini benar apabila dibandingkan dengan validasi terhadap jumlah data test yang digunakan dalam penelitian ini.



Gambar 10. Confusion Matriks Sebelum Hyperparameter

Pada gambar 9 tersebut dapat dilakukan perhitungan untuk akurasi prediksi dari data train yang dilakukan adalah 95.16%. Hal ini diperoleh dari :

Akurasi = Jumlah Prediksi Benar / Total Prediksi

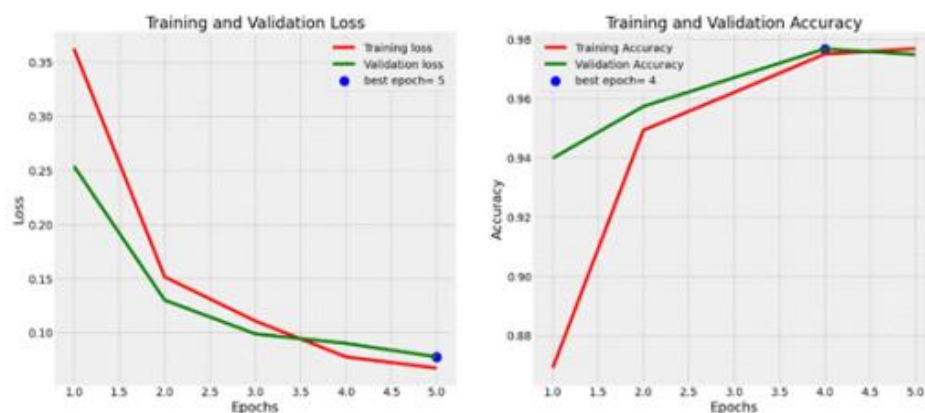
Akurasi = $(106 + 102 + 123 + 143) / (111 + 128 + 130 + 148)$

Akurasi = 492/517

Akurasi = 0,9516 (95.16%)

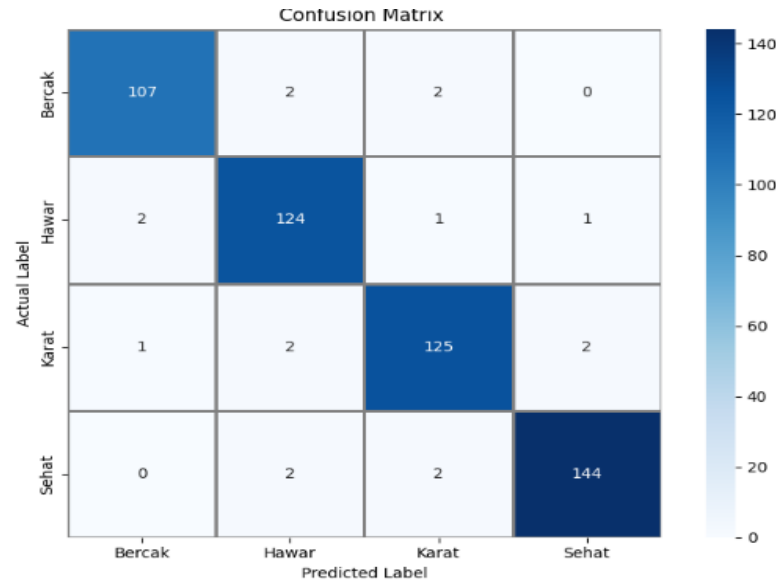
3.3 Setelah Hyperparameter Tuning

Setelah melakukan hyperparameter tuning bahwa pada model belajar dengan baik yang diperlihatkan ke konsistenan antara training dan validation seperti pada gambar dibawah. Sementara pada akurasi pembelajaran yang dilakukan training accuracy dan validation accuracy meningkat kedua-duanya yang menunjukkan bahwa model sudah dapat berjalan dengan baik dan linearitasnya terbukti bekerja dengan baik.



Gambar 11. Loss dan Accuracy pada Training

Setelah menggunakan hyperparameter bahwa untuk Bercak daun benar diprediksi 107 sementara yang lainnya salah, Hawar daun diprediksi benar sebanyak 124 sementara 4 diantaranya diprediksi salah. Karat daun diprediksi benar 125 sementara 5 diantaranya salah diprediksi. Sementara yang terakhir untuk kelas sehat bahwa dari 148 yang dicoba untuk dites diperoleh 144 tepat diprediksi sebagai daun sehat.



Gambar 12. Confusion Matriks Setelah Hyperparameter

Pada gambar 9 tersebut dapat dilakukan perhitungan untuk akurasi prediksi dari data train yang dilakukan adalah 95.16%. Hal ini diperoleh dari :

Akurasi = Jumlah Prediksi Benar/Total Prediksi

Akurasi = $(107+124+125+144)/(111+128+130+148)$

Akurasi = 500/517

Akurasi = 0,9516 (96.71%)

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa penerapan model *Convolutional Neural Network* (CNN) dengan arsitektur DenseNet121 sebagai pendekatan transfer learning terbukti lebih efektif untuk mengidentifikasi penyakit pada tanaman jagung melalui citra daun. Dalam penelitian ini, CNN digunakan sebagai kerangka utama untuk menganalisis dan mengklasifikasikan empat kategori penyakit daun jagung yaitu bercak daun, hawar daun, karat, dan daun sehat. Arsitektur DenseNet121 menunjukkan keunggulan dalam menangani kompleksitas fitur pada gambar dengan performa yang lebih tinggi dibandingkan model lain, seperti CNN Custom, yang juga diuji pada tugas yang sama. DenseNet121 mencapai akurasi sebesar 97% lebih baik dibandingkan CNN Custom yang memiliki akurasi sebesar 95%. Nilai loss DenseNet121 yang lebih rendah dan stabil menunjukkan kemampuannya dalam mempelajari pola dengan baik tanpa mengalami overfitting. Penggunaan teknik hyperparameter tuning, seperti penyesuaian learning rate, dan optimizer dan parameter lainnya juga berkontribusi terhadap peningkatan performa model. Learning rate optimal sebesar 0,001 ditemukan memberikan hasil terbaik selama pelatihan. Selain itu, preprocessing data seperti normalisasi dan augmentasi gambar termasuk rotasi, zoom dan perubahan kontras membantu meningkatkan generalisasi model terhadap data uji, yang berperan penting dalam mendukung performa kedua pendekatan. Arsitektur CNN Custom, meskipun memiliki akurasi yang sedikit lebih rendah dibandingkan DenseNet121, tetap memberikan hasil yang baik dan menunjukkan fleksibilitas dalam menangani data yang bervariasi. Hal ini menjadikannya pilihan yang layak untuk aplikasi serupa, terutama pada kasus dimana sumber daya komputasi lebih terbatas. Secara keseluruhan, penelitian ini membuktikan bahwa DenseNet121 sebagai model transfer learning unggul dalam tugas klasifikasi citra, khususnya deteksi penyakit tanaman jagung. Sementara itu kombinasi teknik preprocessing yang baik dan optimisasi hyperparameter juga merupakan faktor penting dalam meningkatkan performa model. Temuan ini membuka peluang untuk pengembangan lebih lanjut, seperti implementasi sistem deteksi berbasis teknologi di sektor pertanian, yang dapat membantu petani dalam memantau kesehatan tanaman secara real-time, meningkatkan efisiensi pengelolaan lahan, dan mendukung peningkatan hasil produksi pertanian.

REFERENCES

- [1] S. Safruddin, J. W. Sitopu, A. A. Manurung, I. Satria, and A. Wanto, "Pengelompokan Produksi Tanaman Jagung di Sumatera Utara Menggunakan Algoritma K-Medoids," *J. Media Inform. Budidarma*, vol. 7, no. 1, p. 484, 2023, doi: 10.30865/mib.v7i1.5562.
- [2] S. F. Y. Away, D. Darnetti, and R. Afrizal, "Penanganan Pascapanen Benih Jagung Hibrida di PT Citra Nusantara Mandiri Kota Solok Provinsi Sumatera Barat," *J. Agribus. Community Empower.*, vol. 5, no. 2, pp. 96–107, 2022, doi: 10.32530/jace.v5i2.496.

- [3] R. Adenia, A. E. Minarno, and Y. Azhar, "Implementasi Convolutional Neural Network Untuk Ekstraksi Fitur Citra Daun Dalam Kasus Deteksi Penyakit Pada Tanaman Mangga Menggunakan Random Forest," *REPOSITOR*, vol. 4, no. 4, pp. 473–482, 2022.
- [4] M. Farij Amrulloh *et al.*, "Klasifikasi Penyakit Daun Bawang Menggunakan Algoritma CNN Xception Penulis Korespondensi," Online, 2024.
- [5] T. Ayu, V. Dwi, and A. E. Minarno, "PENDIAGNOSA DAUN MANGGA DENGAN MODEL CONVOLUTIONAL NEURAL NETWORK," 2021.
- [6] K. Qi, C. Yang, C. Hu, Y. Shen, and H. Wu, "Deep Object-Centric Pooling in Convolutional Neural Network for Remote Sensing Scene Classification," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 14, pp. 7857–7868, 2021, doi: 10.1109/JSTARS.2021.3100330.
- [7] S. Y. Yadhav, T. Senthilkumar, S. Jayanthi, and J. J. A. Kovilpillai, "Plant Disease Detection and Classification using CNN Model with Optimized Activation Function," *Proc. Int. Conf. Electron. Sustain. Commun. Syst. ICESC 2020*, no. Icesc, pp. 564–569, 2020, doi: 10.1109/ICESC48915.2020.9155815.
- [8] R. Lumbantoruan, N. Rajagukguk, A. U. Lubis, M. Claudia, and H. Simanjuntak, "Two-step convolutional neural network classification of plant disease," *IAES Int. J. Artif. Intell.*, vol. 14, no. 1, p. 584, Feb. 2025, doi: 10.11591/ijai.v14.i1.pp584-591.
- [9] V. Tiwari, C. Pandey, A. Dwivedi, and V. Yadav, "Image Classification Using Deep Neural Network," in *Proceedings - IEEE 2020 2nd International Conference on Advances in Computing, Communication Control and Networking, ICACCCN 2020*, Institute of Electrical and Electronics Engineers Inc., Dec. 2020, pp. 730–733. doi: 10.1109/ICACCCN51052.2020.9362804.
- [10] S. Panichella and M. Ruiz, "Requirements-collector: Automating requirements specification from elicitation sessions and user feedback," in *Proceedings of the IEEE International Conference on Requirements Engineering*, IEEE Computer Society, Aug. 2020, pp. 404–407. doi: 10.1109/RE48521.2020.00057.
- [11] S. U. Rehman and V. Gruhn, "A Sequential VGG16+CNN-Based Automated Approach With Adaptive Input for Efficient Detection of Knee Osteoarthritis Stages," *IEEE Access*, vol. 12, no. February, pp. 62407–62415, 2024, doi: 10.1109/ACCESS.2024.3395062.
- [12] D. Chakraborty, U. Sur, and P. K. Banerjee, "Random forest based fault classification technique for active power system networks," *2019 5th IEEE Int. WIE Conf. Electr. Comput. Eng. WIECON-ECE 2019 - Proc.*, pp. 17–20, 2019, doi: 10.1109/WIECON-ECE48653.2019.9019922.
- [13] H. Tejaswini, M. M. Manohara Pai, and R. M. Pai, "Automatic Estuarine Fish Species Classification System Based on Deep Learning Techniques," *IEEE Access*, 2024, doi: 10.1109/ACCESS.2024.3468438.
- [14] F. Guo, R. He, and J. Dang, "Implicit Discourse Relation Recognition via a BiLSTM-CNN Architecture with Dynamic Chunk-Based Max Pooling," *IEEE Access*, vol. 7, pp. 169281–169292, 2019, doi: 10.1109/ACCESS.2019.2954988.
- [15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, no. July, pp. 2261–2269, 2017, doi: 10.1109/CVPR.2017.243.
- [16] U. P. Singh, S. S. Chouhan, S. Jain, and S. Jain, "Multilayer Convolution Neural Network for the Classification of Mango Leaves Infected by Anthracnose Disease," *IEEE Access*, vol. 7, pp. 43721–43729, 2019, doi: 10.1109/ACCESS.2019.2907383.
- [17] M. F. Almufareh, "An Edge Computing-Based Factor-Aware Novel Framework for Early Detection and Classification of Melanoma Disease Through a Customized VGG16 Architecture With Privacy Preservation and Real-Time Analysis," *IEEE Access*, vol. 12, no. August, pp. 113580–113596, 2024, doi: 10.1109/ACCESS.2024.3444050.
- [18] N. Nurtiwi, R. Ruliana, and Z. Rais, "Convolutional Neural Network (CNN) Method for Classification of Images by Age," *JINAV J. Inf. Vis.*, vol. 3, no. 2, pp. 126–130, Dec. 2022, doi: 10.35877/454ri.jinav1481.
- [19] R. Yati, T. Rohana, and A. R. Pratama, "Klasifikasi Jenis Mangga Menggunakan Algoritma Convolutional Neural Network," *J. MEDIA Inform. BUDIDARMA*, vol. 7, no. 3, p. 1265, Jul. 2023, doi: 10.30865/mib.v7i3.6445.
- [20] K. Stefanus and H. Leong, "Comparison of Random Forest Algorithm Accuracy With Xgboost Using Hyperparameters," *Proxies J. Inform.*, vol. 7, no. 1, pp. 15–23, 2024, doi: 10.24167/proxies.v7i1.12464.