

Optimasi Hyperparameter CNN dengan Arsitektur VGG16 Menggunakan Grid Search Untuk Klasifikasi Penyakit Buah Delima

Muhammad Fawzan¹, Daniel Udjulawa²

¹² Program Studi Informatika, Universitas Multi Data Palembang, Palembang, Indonesia

Email : fawzan_mhd@mhs.mdp.ac.id , daniel@mdp.ac.id

Article Information

Article history

Received 19 October 2025

Revised 25 November 2025

Accepted 5 December 2025

Available 30 December 2025

Keywords

CNN

Disease Classification

Grid Search

Pomegranate

VGG16

Corresponding Author:

Name of corresponding author,

Affiliation,

Email

fawzan_mhd@mhs.mdp.ac.id

Abstract

Early detection of pomegranate fruit diseases is crucial to reduce yield losses and improve harvest quality; however, visual identification in the field is often subjective and difficult due to the similarity of symptoms among different diseases. This study aims to develop a pomegranate fruit disease classification model using a Convolutional Neural Network (CNN) based on the VGG16 architecture, optimized through the Grid Search method. The dataset consists of five classes: 886 *Alternaria* samples, 116 *Anthraco* samples, 966 *Bacterial Blight* samples, 631 *Cercospora* samples, and 1,450 Healthy samples, resulting in a total of 5,099 images. The dataset underwent preprocessing and data augmentation to increase variability and prevent overfitting. After balancing the dataset, it was split into 70% training data, 20% validation data, and 10% testing data. Hyperparameters such as epoch, batch size, learning rate, and optimizer were evaluated using Grid Search to determine the optimal configuration. The results indicate that the best performance was achieved using 100 epochs, a batch size of 32, a learning rate of 0.0001, and the Adam optimizer. The proposed model achieved a testing accuracy of 99.59%, with precision, recall, and F1-score values of 0.996. These findings demonstrate that the optimized VGG16-based CNN model is highly effective in accurately classifying pomegranate fruit diseases.

Keywords : CNN, Disease Classification, Grid Search, Pomegranate, VGG16.

Abstrak

Deteksi dini penyakit pada buah delima penting untuk menekan kehilangan hasil dan meningkatkan kualitas panen, namun identifikasi visual di lapangan cenderung subjektif dan sulit dibedakan antar penyakit yang bergejala mirip. Tujuan penelitian ini adalah untuk membuat model klasifikasi penyakit buah delima menggunakan Convolutional Neural Network (CNN) dengan arsitektur VGG16 yang dioptimalkan melalui metode Grid Search. Dataset ini terdiri dari 5 class yaitu 886 data *Alternaria*, 116 data *Anthraco*, 966 data *Bacterial Blight*, 631 data *Cercospora*, dan 1450 data *Healthy* dengan jumlah keseluruhan mencapai 5099 data, kemudian diproses melalui tahapan preprocessing dan peningkatan data untuk meningkatkan variasi dan mencegah overfitting. Setelah *dataset* memiliki jumlah data yang seimbang, *dataset* dilakukan *split* dengan proporsi 70% data *training*, 20% data *validation* dan 10% data *testing*. Epoch, batch size, learning rate, dan optimizer termasuk dalam kombinasi hyperparameter terbaik yang ditemukan melalui Grid Search. Hasil penelitian menunjukkan bahwa konfigurasi optimal dicapai dengan kombinasi epoch 100, ukuran batch 32, tingkat pembelajaran 0.0001, dan optimizer Adam. Model menghasilkan tingkat akurasi pengujian sebesar 99,59%, dengan nilai akurasi, recall, dan f1-score masing-masing 0,996. Temuan ini menunjukkan bahwa CNN berbasis VGG16 yang dioptimalkan mampu mengklasifikasikan penyakit buah delima dengan sangat baik.

Kata Kunci : Buah Delima, CNN, Grid Search, Klasifikasi Penyakit, VGG16.

Copyright@2025 Muhammad Fawzan, Daniel Udjulawa

This is an open access article under the [CC-BY-NC-SA](https://creativecommons.org/licenses/by-nc-sa/4.0/) license.



1. Pendahuluan

Buah delima (*Punica Granatum L.*) merupakan tanaman perdu atau perdu gugur yang tingginya mencapai 5-8 meter (Saputro & Khairudin, 2022). Buah delima juga merupakan salah satu komoditas hortikultura yang memiliki nilai ekonomi dan gizi tinggi (Abbas et al., 2025). Menurut Sodré dkk, buah delima memiliki kandungan senyawa bioaktif seperti *polifenol*, *flavonoid*, dan *antosianin*, yang menjadikannya sebagai buah dengan segudang manfaat bagi kesehatan manusia (Sodré et al., 2025). Konsumsi buah delima tidak hanya sebatas sebagai buah segar, tetapi juga berkembang pesat ke dalam industri pengolahan makanan, farmasi, dan kosmetik.

Namun, produktivitas dan kualitas buah delima sangat dipengaruhi oleh serangan penyakit tanaman. Menurut Paul dkk, hal ini dikarenakan secara visual, gejala penyakit yang disebabkan oleh penyakit yang berbeda seperti *Alternaria*, *Cercospora* atau penyakit lainnya yang seringkali menunjukkan tanda yang mirip, misalnya berupa bercak-bercak pada permukaan kulit buah delima (Paul et al., 2024). Lalu, menurut studi yang dilakukan oleh Jaiman dkk pada tahun 2024, mengenai penyakit pascapanen pada buah delima, identifikasi yang keliru oleh petani dapat berujung pada penanganan yang tidak tepat, seperti penggunaan pestisida yang tidak sesuai (Jaiman et al., 2024). Oleh karena itu, kesalahan yang terjadi ini tidak hanya menyebabkan kerugian ekonomi akibat gagal panen, tetapi juga menimbulkan dampak lingkungan dan potensi residu kimia pada buah yang berbahaya bagi konsumen.

Menurut Opara dkk, bahwa kehilangan buah delima berkisar antara 15,3 hingga 20,1% dari buah yang dipanen di kebun studi kasus. Ini setara dengan kehilangan rata-rata 117,76 ton buah delima yang dipanen per musim panen. Jumlah buah ini dikeluarkan dari rantai nilai dan dijual terutama dengan harga rendah untuk dijadikan jus dan keperluan lainnya. Penyebab utama kehilangan buah di lahan pertanian adalah faktor lingkungan. Pada tiga kultivar delima yang diteliti, sengatan matahari dan keretakan diidentifikasi sebagai penyebab utama kehilangan buah, yang mencakup sekitar 43,9% dari seluruh kehilangan di lahan pertanian. Lebih lanjut, kerusakan mekanis, termasuk memar dan cedera serta faktor mikroba (menyebabkan pembusukan dan kerusakan) berkontribusi terhadap kehilangan buah yang signifikan. Besarnya kehilangan buah di lahan pertanian untuk varietas 'Acco' yang dipanen adalah 15,8 hingga 16,5%, 15,3% hingga 16,2% untuk 'Hershkawitz' dan 19,7% hingga 20,1% untuk 'Wonderful'. Penggunaan dua metode panen yang berbeda merupakan keterbatasan penelitian ini—baik 'Acco' maupun 'Herskawitz' dipanen berdasarkan ukuran buah yang seragam, sementara 'Wonderful' dipanen secara strip-harvest karena pembatasan perjalanan selama pandemi COVID-19 (Opara dkk., 2021).

Keterbatasan mengklasifikasi penyakit buah delima secara manual yang dilakukan oleh petani ini bersifat subjektif dan memerlukan waktu yang cukup lama, serta bergantung pada keahlian individu menjadi masalah lapangan yang mendesak

untuk dicarikan solusinya. Teknologi berbasis *machine learning* dan *deep learning* khususnya dalam bidang *computer vision* memberikan potensi solusi yang cukup untuk mengatasi permasalahan yang ada dan dapat berguna pada bidang agrikultur khususnya petani buah delima.

Deep learning merupakan bagian dari *machine learning* yang memiliki kemampuan untuk mengenali pola gambar, suara, teks dan data lainnya yang kompleks sehingga dapat menghasilkan prediksi yang akurat (Dewi Linda & Dwi Hartanto, 2024). Salah satu kemampuan *deep learning* adalah klasifikasi citra pada objek. *Convolutional Neural Network* (CNN) merupakan salah satu metode dalam *deep learning* yang digunakan untuk mengklasifikasikan citra objek, di mana algoritma CNN adalah jenis jaringan saraf tiruan yang saat ini banyak digunakan untuk pengenalan suatu citra (Dewi Linda & Dwi Hartanto, 2024). Seiring dengan kemajuan teknologi, implemmentasi pengolahan citra dan *deep learning* seperti CNN telah menunjukkan hasil performa yang cukup baik dalam mengklasifikasi penyakit tanaman atau buah.

Karena termasuk kedalam *deep learning*, CNN memerlukan pelatihan model yang cukup lama dan bergantung pada perangkat yang digunakan. CNN memiliki beberapa arsitektur umum yang sering digunakan seperti *MobileNet*, *ResNet*, *LeNet*, VGG-16, *AlexNet*, *EfficientNet*, dan masih banyak lagi yang lainnya, di mana masing-masing arsitektur memiliki kelebihan dan kekurangan masing-masing (Fitra Maulana & Rochmawati, 2019).

Dalam beberapa tahun terakhir, *Convolutional Neural Network* (CNN) telah berkembang sebagai salah satu metode yang efektif dalam analisis dan klasifikasi gambar. CNN memiliki kemampuan dalam mengenali pola visual secara otomatis dan telah terbukti memberikan tingkat akurasi tinggi dalam berbagai tugas klasifikasi. Dengan keunggulan tersebut, CNN berpotensi untuk diterapkan dalam klasifikasi penyakit buah delima, sehingga dapat meningkatkan proses identifikasi secara otomatis.

Arsitektur yang akan digunakan dalam klasifikasi penyakit buah delima adalah VGG-16. *Visual Geometry Group* atau yang biasa dikenal dengan VGG-16 adalah arsitektur yang dikembangkan oleh Simonyan dan Zisserman di Universitas Oxford pada tahun 2013 (Great Learning, 2025). Secara umum VGG-16 adalah jaringan saraf tiruan konvolusional yang memiliki lapisan sebanyak 16 lapisan yang mampu mengklasifikasikan citra dengan sangat baik melalui penggunaan berulang filter konvolusi berukuran 3x3 untuk mengekstraksi fitur-fitur gambar yang kompleks. Arsitektur VGG-16 sendiri memiliki 13 *layer* konvolusi, dan 3 *layer fully connected*. Arsitektur ini cocok untuk mengklasifikasikan citra karena arsitektur VGG-16 terbukti memberikan hasil yang baik dalam berbagai aplikasi pengenalan objek dan klasifikasi citra. Penggunaan VGG-16 ini sangat relevan untuk mendapatkan hasil yang sesuai (Great Learning, 2025).

Selain itu, penelitian yang dilakukan oleh Pradana dkk, model VGG-16 berhasil melakukan klasifikasi citra tanaman tomat ke dalam 8 kelas yang meliputi beberapa jenis penyakit dan kondisi sehat dengan tingkat akurasi 91%, sehingga berpotensi besar untuk digunakan dalam proses klasifikasi penyakit buah secara otomatis dan akurat (Pradana et al., 2023). VGG-16 hadir dalam dua versi, yakni VGG-16 dan VGG-19, di mana masing-masing versi memiliki perbedaan dari sisi kedalaman layer dan kompleksitas model.

Celah penelitian pada studi ini terletak pada kebutuhan evaluasi yang lebih terukur dan dapat direplikasi untuk optimasi hyperparameter VGG16 pada klasifikasi penyakit buah delima, terutama ketika ketersediaan dataset terstandar menjadi prasyarat penting. Dataset terstandarisasi untuk penyakit buah delima telah dipublikasikan dan dapat dimanfaatkan untuk pelatihan model secara lebih konsisten (Pakruddin & Hemavathy, 2025), namun kajian yang secara khusus memformulasikan dan menguji kombinasi hyperparameter VGG16 melalui grid search pada domain ini masih terbatas.

Berdasarkan gap tersebut, penelitian ini bertujuan mengoptimasi hyperparameter CNN berarsitektur VGG16 menggunakan grid search untuk meningkatkan kinerja klasifikasi penyakit buah delima. Ruang lingkup penelitian dibatasi pada klasifikasi multikelas berbasis citra buah delima dengan evaluasi menggunakan metrik kinerja, tanpa membahas estimasi tingkat keparahan atau rekomendasi perlakuan. Kontribusi penelitian diharapkan memberikan baseline optimasi yang transparan dan dapat direplikasi, serta bukti empiris pengaruh konfigurasi hyperparameter terhadap kinerja VGG16 pada kasus penyakit buah delima untuk mendukung pengembangan sistem diagnosis yang lebih andal di bidang pertanian digital.

Ruang lingkup dan batasan pada penelitian berjudul “Optimasi Hyperparameter CNN Dengan Arsitektur VGG-16 Menggunakan Grid Search Untuk Klasifikasi Penyakit Buah Delima” mencakup penggunaan dataset publik yang diperoleh dari platform Kaggle dengan judul “Pomegranate Fruit Diseases [Image] Dataset”, yang terdiri dari 5.099 citra buah delima dan terbagi ke dalam lima kelas, yaitu Alternaria, Anthracnose, Bacterial Blight, Cercospora, dan Healthy. Dataset tersebut dibagi ke dalam tiga subset dengan proporsi 70% sebagai data training, 20% sebagai data validation, dan 10% sebagai data testing. Penelitian ini berfokus pada penerapan serta optimasi hyperparameter pada model Convolutional Neural Network (CNN) dengan arsitektur VGG-16 untuk melakukan klasifikasi penyakit buah delima. Optimasi hyperparameter dilakukan menggunakan metode Grid Search guna memperoleh kombinasi parameter terbaik yang mampu meningkatkan performa model. Proses pelatihan model klasifikasi menggunakan Adam Optimizer sebagai algoritma utama dalam pembaruan bobot jaringan. Seluruh tahapan pengembangan, pelatihan, dan optimasi model dilakukan melalui platform Kaggle Notebook dengan bahasa pemrograman Python versi 3.11.13. Evaluasi kinerja model klasifikasi dilakukan

menggunakan confusion matrix yang menghasilkan metrik evaluasi berupa accuracy, precision, recall, dan f1-score.

2. Kajian Terdahulu

Pada penelitian terdahulu yang dilakukan oleh Adisaputra dan Weny pada tahun 2023, menjelaskan bahwa lapisan yang ada pada VGG-16 dan VGG-19 terdapat modifikasi pada lapisan konvolusi yang ada pada VGG-19, di mana VGG-19 menambahkan tiga lapisan konvolusi tambahan pada blok tertentu, sehingga VGG-19 lebih dalam (Noorizki & Kusumawati, 2023). Dari hasil simulasi menggunakan VGG-16 dan VGG-19 diperoleh 98% untuk akurasi VGG-16 dengan waktu training 73.405 detik dan 97% untuk akurasi VGG-19 dengan waktu training 78.098 detik. Hal ini menunjukkan bahwa arsitektur VGG-16 dari segi akurasi dan waktu training secara signifikan lebih unggul dibandingkan arsitektur VGG-19. Hal ini dikarenakan VGG-19 melakukan penambahan kedalaman yang membuat waktu training menjadi lebih lama (Noorizki & Kusumawati, 2023).

Kemudian dalam penelitian terdahulu yang telah dilakukan oleh Shah dkk, menjelaskan tentang klasifikasi citra penyakit beras yang membandingkan beberapa arsitektur seperti *InceptionV3*, VGG-16, VGG-19, CNN, and ResNet-50 (Shah et al., 2023). Hasil akurasi yang didapatkan menggunakan arsitektur VGG-16 adalah *Overall Accuracy* (OA) sebesar 97,45%, akurasi validasi sebesar 97,91%, akurasi *f1-score* sebesar 97%, *Area Under Curve* (AUC) sebesar 98,01% dan akurasi *precision* sebesar 97,66%. Sedangkan VGG-19 menghasilkan akurasi *Overall Accuracy* (OA) sebesar 96,49%, akurasi validasi sebesar 75,76%, akurasi *f1-score* sebesar 94,17%, *Area Under Curve* (AUC) sebesar 98,91% dan akurasi *precision* sebesar 96,30% (Shah et al., 2023). Dari penelitian tersebut, dapat dilihat bahwa arsitektur VGG-16 memiliki performa *Overall Accuracy* (OA) yang lebih baik dari pada performa arsitektur VGG-19.

Berdasarkan hal tersebut, ini menunjukkan bahwa proses klasifikasi penyakit pada buah delima memiliki peranan yang sangat penting, di mana tidak hanya memastikan kondisi kesehatan dari buah, namun juga sebagai langkah preventif dalam mendukung kualitas produksi dan meminimalisasi adanya kerugian. Dengan demikian, penelitian mengenai klasifikasi penyakit pada buah delima menggunakan metode CNN dengan arsitektur VGG16 ini diperlukan untuk menghadirkan solusi yang lebih cepat dan akurat dalam memastikan kualitas buah delima yang baik.

Selain pentingnya penerapan arsitektur Convolutional Neural Network (CNN) seperti VGG16 dalam klasifikasi citra penyakit pada buah delima, tantangan yang sering dihadapi dalam penerapan model deep learning adalah pemilihan *hyperparameter* yang optimal. *Hyperparameter* seperti learning rate, batch size, jumlah epochs, serta jenis *optimizer* memiliki pengaruh besar terhadap performa model (Hong & Chen, 2023). Penentuan nilai hyperparameter yang tidak sesuai dapat menyebabkan model mengalami

underfitting atau *overfitting*, sehingga menurunkan akurasi klasifikasi. Oleh karena itu, proses optimasi *hyperparameter* menjadi langkah penting dalam pengembangan model deep learning yang baik.

Salah satu metode yang banyak digunakan untuk menemukan kombinasi *hyperparameter* terbaik adalah *Grid Search*, yaitu teknik pencarian sistematis yang menguji berbagai kombinasi parameter secara menyeluruh untuk menemukan konfigurasi dengan performa terbaik berdasarkan metrik evaluasi tertentu (Wowor, 2025). Menurut penelitian yang dilakukan oleh Gülmez, penerapan *Grid Search* pada CNN terbukti dapat meningkatkan akurasi hingga 3% dibandingkan konfigurasi seperti *random search*, *genetic algorithms*, dan *Bayesian optimization* pada kasus klasifikasi citra tanaman (Gülmez, 2025). Hal ini menunjukkan bahwa pemilihan parameter yang tepat melalui optimasi terstruktur mampu meningkatkan kemampuan generalisasi model terhadap data baru.

Penelitian lain oleh Simanjuntak dkk. (2024) juga menegaskan bahwa tuning *hyperparameter* menggunakan metode *Grid Search* menghasilkan model CNN dengan stabilitas yang lebih baik, terutama dalam menangani dataset citra dengan variasi tinggi. Dengan demikian, penerapan *Grid Search* tidak hanya meningkatkan nilai akurasi, tetapi juga mempercepat proses konvergensi model. Dalam konteks klasifikasi penyakit buah delima, pendekatan ini dapat membantu memperoleh model CNN berbasis arsitektur VGG-16 yang tidak hanya akurat, tetapi juga efisien dalam waktu pelatihan dan penggunaan sumber daya komputasi.

Sejalan dengan hal tersebut, penelitian terbaru oleh Jha dkk, menunjukkan bahwa kombinasi antara *optimasi hyperparameter* dan teknik *augmentasi* data mampu meningkatkan *robustness* model CNN terhadap variasi citra tanaman di lingkungan nyata (Jha et al., 2025). Maka dari itu, penelitian ini tidak hanya berfokus pada penerapan CNN dengan arsitektur VGG-16, tetapi juga pada *optimasi hyperparameter* menggunakan metode *Grid Search* dan *Optimizer* ADAM untuk menghasilkan model klasifikasi penyakit buah delima yang lebih optimal, stabil, dan akurat dalam membantu petani mengidentifikasi penyakit secara otomatis.

Buah Delima

Delima termasuk tanaman yang memiliki ketahanan relatif tinggi terhadap lingkungan kering, sehingga dapat dibudidayakan di lahan yang kering maupun lahan dataran tinggi, sehingga produktivitas buah delima dapat bertumbuh dengan cukup baik (Valero-Mendoza et al., 2023). Meskipun demikian, tanaman ini rentan terhadap serangan penyakit, terutama pada tahap pembentukan dan pematangan buah. Buah delima dapat dilihat pada Gambar 1.



Gambar 1. Buah Delima

Sumber: (Valero-Mendoza et al., 2023)

Delima dikenal sebagai buah dengan kandungan nutrisi senyawa bioaktif yang melimpah seperti vitamin C, vitamin K, asam folat, kalium, *polifenol*, *antosianin*, *flavonoid*, hingga *punicalagin* (Mansur et al., 2022). Kandungan tersebut berfungsi sebagai antioksidan, antiinflamasi, serta memiliki efek antimikroba yang bermanfaat bagi kesehatan. Konsumsi rutin delima terbukti dapat menurunkan risiko penyakit kardiovaskular, membantu pencegahan kanker, dan mendukung reproduksi (Jang et al., 2024). Tidak hanya itu, ekstrak delima juga banyak digunakan pada industri kosmetik karena dapat memperbaiki elastisitas kulit, mempercepat generasi sel, serta melindungi kulit dari penuaan diri (Chan et al., 2021). Kandungan yang ada dalam buah delima dipengaruhi oleh faktor genetik, lingkungan, metode budidaya, serta perlakuan pascapanen.

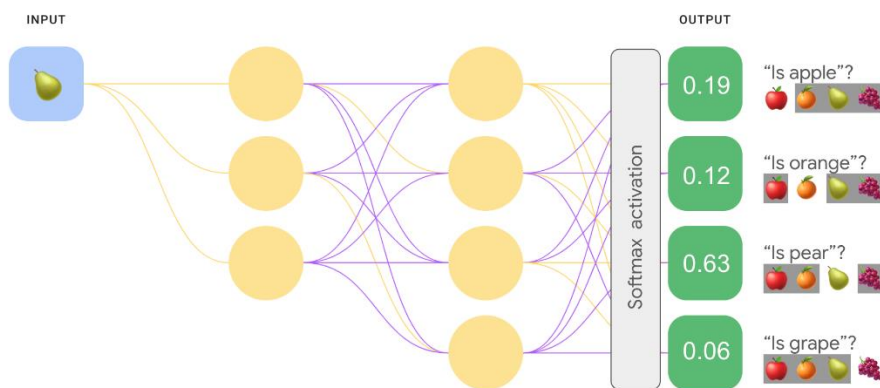
Buah delima tidak hanya bernilai dari sisi gizi, tetapi juga memiliki peran penting bagi kesehatan, industri dan ekonomi. Produk yang dihasilkan dari buah delima dapat berupa jus, sirup, ekstrak, hingga bahan baku kosmetik (*skincare*) dan farmasi, semakin meningkatkan ekonominya di pasar global. Namun, keberhasilan komoditas ini sangat dipengaruhi oleh kualitas buah yang dipasarkan. Penyakit tanaman menjadi tantangan besar, terutama serangan jamur *Alternaria alternata* yang menyebabkan bercak hitam serta *Colletotrichum gloeosporioides* penyebab *anthracnose*. Kedua patogen tersebut dilaporkan menyebabkan kerusakan serius pada buah, bahkan mengakibatkan kehilangan hasil dan menurunkan nilai jual secara signifikan (Tirrò et al., 2024). Selain menurunkan produktivitas, beberapa infeksi jamur diketahui menghasilkan mikotoksin berbahaya, yang dapat berdampak negatif terhadap kesehatan konsumen.

Di tingkat lapangan, petani umumnya masih mengandalkan pengamatan visual untuk mengidentifikasi penyakit delima. Gejala yang mirip antar patogen seringkali membingungkan, sehingga diagnosis manual cenderung tidak konsisten dan berpotensi menyebabkan penggunaan pestisida yang berlebihan atau tidak tepat sasaran (Manjunatha et al., 2022). Kesalahan ini tidak hanya menambah biaya produksi, tetapi

juga dapat meninggalkan residu berbahaya pada buah (Math et al., 2024). Untuk mengatasi keterbatasan tersebut, oleh karena itu penerapan sistem klasifikasi berbasis CNN dengan arsitektur VGG-16, akan membantu petani lebih terbantu dalam menjaga kualitas panen, meningkatkan daya saing produk di pasar global, serta mendukung keamanan pangan bagi konsumen.

Classification

Secara umum, klasifikasi adalah proses pengelompokan yang artinya mengumpulkan benda/entitas yang sama serta memisahkan benda/entitas yang tidak sama (Abdel Rahman Shalaby et al., 2019). Dalam pengertian ilmu komputer, klasifikasi adalah proses evaluasi data untuk mengkategorikannya ke dalam salah satu atau beberapa kelas yang telah ditentukan, dimana hal ini dimulai dengan membangun model sebagai representasi prototipe yang akan disimpan sebagai referensi yang kemudian model tersebut digunakan untuk mengenali atau mengklasifikasikan objek data lain, sehingga dapat ditempatkan dalam salah satu kelas yang ada dalam model yang telah disimpan (Wibawa Rahayu et al., 2024). Gambar *Classification Object* dapat dilihat pada Gambar 2.



Gambar 2. Classification Object

Sumber : <https://developers.google.com/machine-learning/crash-course/neural-networks/multi-class?hl=id>

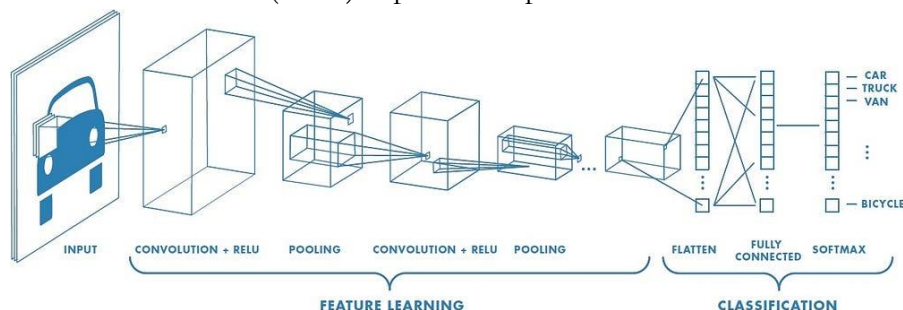
Tujuan dari klasifikasi adalah memahami sebuah *dataset* sehingga kita dapat membuat suatu aturan atau model yang mampu mengklasifikasikan data baru yang belum pernah dilihat sebelumnya (Wibawa Rahayu et al., 2024), sehingga dapat membantu kita dalam mempercepat proses kategorisasi dengan akurat.

Jadi secara singkat, *Classification* atau klasifikasi merupakan suatu proses pengelompokan yang bertujuan untuk mengkategorikan suatu *input* / data ke dalam kelas atau kategori tertentu berdasarkan model yang telah dibangun dengan *dataset*

tertentu, sehingga model tersebut mampu mengenali serta mengklasifikasikan data baru secara lebih cepat dan akurat.

Convolution Neural Network (CNN)

Convolution Neural Network (CNN) adalah salah satu algoritma dalam *deep learning* (Azmi et al., 2023) yang diusulkan pertama kali pada tahun 1960-an (M. Wu et al., 2023) dan termasuk ke dalam jenis spesifik dari *Artificial Neural Network* (ANN) (Bezdan & Baćanin Džakula, 2019) berbasiskan *multi-layer neural network* yang dapat melakukan identifikasi, pengenalan, klasifikasi objek serta mendeteksi dan mengelompokkan objek yang ada didalam gambar (Taye, 2023). CNN mampu memahami detail pada gambar dengan baik karena memiliki arsitektur yang sesuai dengan cara kerja otak manusia dalam memproses informasi visual (Anhar & Adi Putra, 2023). Terkait arsitektur dari *Convolutional Neural Network* (CNN) dapat dilihat pada Gambar 3.



Gambar 3. Arsitektur *Convolutional Neural Network* (CNN)

Sumber: (Ghandi et al., 2024)

Secara umum, arsitektur CNN terdiri dari dua bagian utama yaitu *feature learning* dan *classification layer* (Pratama et al., 2024). *Feature learning* merupakan lapisan awal yang bertugas dalam menerima input berupa gambar / citra secara langsung diawal dan memprosesnya hingga menghasilkan output data (Azmi et al., 2023). *Classification Layer* merupakan lapisan yang menerima *input* dari *output layer* bagian *feature learning*, dimana lapisan ini berisikan banyak *neuron* yang saling terkoneksi penuh (*fully connected*) dengan lapisan lain (Azmi et al., 2023).

Dalam *feature learning* terdapat beberapa *layer* yang disusun secara berurutan, diantaranya:

a. Convolution Layer

Convolution Layer adalah elemen kunci dalam arsitektur *Convolutional Neural Network* yang terdiri dari kumpulan filter yang diterapkan melalui operasi konvolusi pada *input* yang disajikan untuk menghasilkan *feature map* yang mana dalam *context convolution layer* setiap filter diwakilkan oleh matriks yang mengandung serangkaian nilai unik yang akan mengalami proses pembaruan sepanjang fase training sesudah

dilakukan konvolusi, dimana operasi *convolution* sendiri dilakukan secara berulang dan melibatkan proses perkalian matriks antara *filter* dan *convolution layer* yang kemudian *output* dari operasi ini akan diakumulasikan untuk menghasilkan *feature map* yang diinginkan pada citra (Ghandi et al., 2024).

b. *Activation Layer (ReLU)*

Activation Layer merupakan suatu *layer* yang memproses perubahan nilai-nilai *feature map* pada jarak tertentu tergantung pada fungsi aktivasi yang digunakan dan biasanya pada bagian *feature learning* CNN, fungsi aktivasi yang digunakan adalah ReLu (*Rectified Linear Unit*) (Azmi et al., 2023). ReLu merupakan operasi untuk mengenalkan non-linearitas dan meningkatkan representasi dan model, di mana fungsi aktivasi ReLu akan menghilangkan *vanishing gradient* dengan cara menerapkan fungsi $f(x) = \max(0, x)$ atau aktivasi elemen akan dilakukan saat berada pada ambang batas 0 (Paraijun et al., 2022).

c. *Pooling Layer*

Pooling Layer bisanya sering digunakan pada CNN setelah *convolution layer* yang berfungsi untuk mengurangi dimensi atau ukuran panjang dan lebar pada citra dengan teknik *subsampling* atau *downsampling*, dimana *pooling layer* umumnya menggunakan *filter* 2 dan *stride* 2 dengan jenis *max pooling* yaitu nilai *maximum* yang akan diambil secara berurutan pada citra (Bezdan & Baćanin Džakula, 2019).

Sedangkan, didalam *classification layer* juga terdapat beberapa *layer* yang tersusun, diantaranya:

a. *Flatten*

Flatten merupakan suatu *layer* yang membentuk ulang fitur (*reshape feature map*) menjadi sebuah *vector* agar bisa digunakan sebagai *input* dari *fully-connected layer* (Paraijun et al., 2022).

b. *Fully Connected Layer*

Fully Connected Layer adalah struktur yang menghubungkan setiap neuron aktivasi dari lapisan sebelumnya secara langsung dengan setiap neuron pada lapisan berikutnya, dimana perbedaan mendasar antara *fully connected layer* dan *convolution layer* adalah bahwa dalam *convolution layer*, neuron-neuron hanya terhubung dengan wilayah tertentu dari *input*, sementara dalam *fully connected layer*, setiap neuron terhubung dengan semua neuron pada lapisan sebelumnya yang mana pada titik ini semua fitur yang telah dipelajari oleh *network* dari gambar akan digunakan untuk melakukan klasifikasi (Ghandi et al., 2024).

c. *Activation Layer (Softmax)*

Pada bagian terakhir dalam *classification layer* terdapat fungsi aktivasi dengan jenis *softmax* yang berfungsi untuk mendapatkan hasil klasifikasi, di mana fungsi ini menghitung probabilitas dari setiap *class* atas semua *class* target yang memungkinkan

dan akan membantu untuk menentukan class target untuk *input* yang diberikan (Paraijun et al., 2022).

Hyperparameter Tuning

Hyperparameter tuning adalah metode yang dapat digunakan untuk mengoptimalkan parameter yang digunakan dalam algoritma *Machine Learning/Deep Learning* seperti *Convolutional Neural Network* (CNN) dengan tujuan untuk meningkatkan kinerja dari model tersebut (Auliaddina & Arifin, 2024). *Hyperparameter tuning* terdiri dari beberapa jenis, salah satunya adalah *Grid Search*. *Grid search* adalah metode tradisional yang mengoptimalkan *hyperparameter* dengan mencari secara mendalam subset tertentu dari ruang *hyperparameter* algoritma pelatihan (Anugerah Simanjuntak et al., 2024).

Hyperparameter yang ditentukan untuk model umumnya terdiri dari *learning rate*, *batch size*, jumlah *epoch* (Anugerah Simanjuntak et al., 2024), dan *optimizer* (Morales-Hernández et al., 2023). *Hyperparameter* ini memainkan peranan penting dalam mengoptimalkan kinerja model selama proses *training* (Anugerah Simanjuntak et al., 2024).

a. Learning Rate

Learning rate atau laju pembelajaran adalah salah satu parameter *training* yang ditetapkan untuk menghitung nilai koreksi bobot pada waktu proses *training* (Rochmawati, Hidayati, Yamasari, Agustin Tjahyaningtijas, et al., 2021), dimana apabila *learning rate* terlalu tinggi, maka model mungkin gagal untuk konvergen dan sebaliknya jika *learning rate* terlalu rendah, maka model mungkin akan membutuhkan waktu terlalu lama untuk konvergen (Anugerah Simanjuntak et al., 2024).

b. Batch Size

Batch Size atau ukuran *batch* adalah istilah yang digunakan dalam *machine learning* dan mengacu pada jumlah contoh *training* yang digunakan dalam satu iterasi dan merupakan salah satu *hyperparameter* terpenting untuk disesuaikan dengan sistem *deep learning* (Rochmawati, Hidayati, Yamasari, Peni, et al., 2021), dimana ketika *batch size* yang ditetapkan besar dapat menghasilkan waktu pelatihan yang lebih cepat, tetapi juga dapat menyebabkan kemampuan dan akurasi generalisasi menjadi buruk (Anugerah Simanjuntak et al., 2024).

c. Epoch

Epoch merupakan salah satu *hyperparameter* yang dapat digunakan untuk menentukan banyaknya *training* yang dilakukan terhadap model di seluruh *dataset* (Anugerah Simanjuntak et al., 2024).

d. Optimizer

Optimizer merupakan algoritma yang digunakan untuk meminimalisir *loss function*, dimana *optimizer* akan mempelajari bagaimana menyesuaikan parameter model agar model dapat memberikan prediksi yang lebih baik dan sesuai dengan data yang

digunakan selama pelatihan (Wardani et al., 2023). Biasanya optimizer yang paling banyak digunakan yakni *Adam Optimizer*, karena memiliki kemampuan yang dapat memperbarui bobot dan *learning rate*-nya secara otomatis (*adaptive*) (Asy Syifa & Amelia Dewi, 2022).

ADAM Optimizer

Optimizer ADAM merupakan salah satu algoritma optimasi yang terbaik dalam pelatihan model *deep learning* (Dede Kurniadi et al., 2025). Adam adalah singkatan dari *Adaptive Moment Estimation* yang pertama kali diperkenalkan oleh Kingma dan Ba pada tahun 2015, di mana Adam merupakan algoritma optimasi berbasis *Stochastic Gradient Descent* (SGD) yang menggabungkan dua gagasan penting yakni *momentum* (rata-rata bergerak dari *gradient*) dan penyesuaian laju pembelajaran per parameter berdasarkan varians *gradient* (*learning rate*) (Bock & Weis, 2019). *Optimizer* Adam menghitung dua statistik bagi setiap parameter dengan rata-rata momen pertama (rata-rata *gradient*) dan momen kedua (*varians gradient* yang tidak berpusat) dari *gradient* historis. Menurut Sun dkk, menyatakan bahwa keunggulan utama Adam *Optimizer* terletak pada kemampuannya menggabungkan manfaat dari dua metode optimisasi lainnya, yaitu Momentum (seperti SGD) dan RMSprop, yang membuatnya sering kali konvergen lebih cepat pada berbagai arsitektur jaringan saraf yang kompleks (Sun et al., 2019). Untuk mengatasi bias pada iterasi awal, Adam memperkenalkan *bias correction* yang menghasilkan estimasi *non-bias* dari kedua momen tersebut.

Adam *Optimizer* menjadi salah satu pilihan *default* dalam pelatihan model *deep learning* karena keandalannya dalam menangani *gradient* dengan skala berbeda, *gradient* jarang (*sparse*), dan lanskap *loss function* yang *non-konveks* (Abdulkadirov et al., 2023). Menurut Liu dkk, menyebutkan bahwa Adam menunjukkan performa lebih stabil dibanding algoritma optimasi lainnya seperti SGD, AdaGrad dan RMSprop, khususnya pada pelatihan model dengan jumlah parameter besar seperti CNN (Liu et al., 2023). Dalam konteks pelatihan model CNN, Adam terbukti sangat efektif dalam mempercepat proses pembelajaran dan menjaga stabilitas pelatihan, terutama ketika data yang digunakan memiliki variasi tinggi seperti pada citra tanaman atau citra medis (Dede Kurniadi et al., 2025).

Augmentation

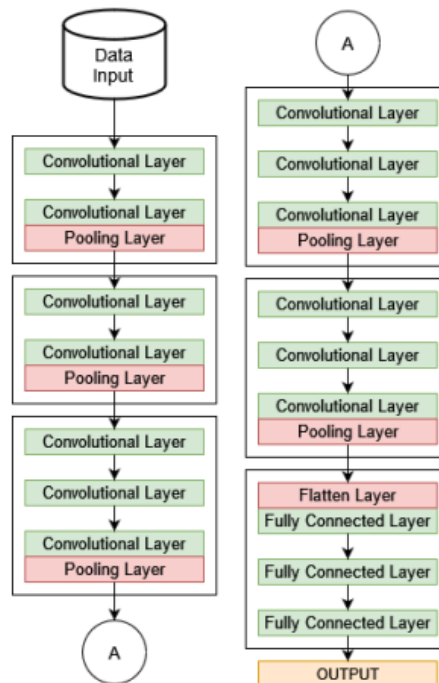
Data augmentation merupakan bagian dari kumpulan teknik regularisasi yang bertujuan untuk meningkatkan kinerja model (Mumuni & Mumuni, 2022). Teknik regularisasi dalam *augmentation* bekerja dengan menambahkan informasi tambahan ke yang mendasarinya untuk menangkap sifat-sifat umum dari masalah yang sedang dimodelkan, di mana hal tersebut memungkinkan model yang dilatih memiliki respons yang lebih stabil dan halus terhadap berbagai variasi data input (pada saat pengujian).

Pada saat augmentasi, *batch normalization* yang ada pada saat *augmentation* akan menormalisasi keluaran berbagai lapisan untuk menghilangkan efek pergeseran kernel internal pada input ke lapisan dalam. Regularisasi juga dapat dicapai dengan mengubah ukuran citra dan struktur model *deep learning*, mengubah jumlah parameter yang dapat dipelajari. Hal tersebut bertujuan untuk mengurangi kompleksitas model sehingga pola yang tidak perlu tidak dipelajari dari data.

Visual Geometry Group (VGG) 16

VGG-16 merupakan salah satu arsitektur *Convolutional Neural Network* (CNN) yang dikembangkan oleh tim Visual Geometry Group di *University of Oxford* (Great Learning, 2025). Arsitektur ini diperkenalkan oleh Simonyan dan Zisserman pada tahun 2014, namun hingga saat ini tetap menjadi model yang banyak digunakan dalam berbagai penelitian karena kesederhanaan desain dan performa yang konsisten. Model VGG-16 dirancang untuk melakukan tugas deteksi, identifikasi dan klasifikasi gambar. Model ini menonjol karena susunan blok-blok konvolusi yang sederhana yang dipadukan dengan *max-pooling* sehingga mendukung *feature extraction* spasial bertingkat dari citra (Sitaula & Hossain, 2021). Desain VGG-16 memungkinkan jaringan belajar kombinasi fitur dasar (tepi, tekstur) pada lapisan awal dan fitur yang semakin kompleks pada lapisan terdalam, menjadikan VGG-16 sangat berguna sebagai *backbone* untuk *transfer learning* pada banyak aplikasi visi komputer (F. Wu & Lin, 2022).

Secara teknis, VGG-16 pada dasarnya tidak berbeda jauh dengan uraian yang telah dijelaskan sebelumnya oleh peneliti. VGG-16 memiliki jumlah lapisan sebanyak 16, di mana terdiri dari 13 lapisan konvolusi dan 3 lapisan *fully-connected* (total 16 lapisan). Lapisan *Rectified Linear Unit* (ReLU) tidak digunakan dalam VGG-16 karena lapisan tersebut sudah sekaligus dijalankan prosesnya menjadi satu pada setiap lapisan konvolusi sebelum dilanjutkan ke tahap berikutnya (Noorizki & Kusumawati, 2023). Gambar Arsitektur VGG-16 dapat dilihat pada Gambar 4.



Gambar 4. Arsitektur VGG-16

Sumber: (Noorizki & Kusumawati, 2023)

Perancangan model CNN VGG-16 dalam penelitian ini akan menggunakan *input* berukuran 224x224x3, yang berarti memiliki panjang 224 piksel, lebar 224 piksel, dan *depth* 3, yang mewakili warna RGB. Selanjutnya, untuk jumlah *filter*, digunakan 64, 128, 256 dan 512 *filter*. *Kernel* yang digunakan berukuran 3x3 dan lapisan *max pooling* memiliki ukuran 2x2.

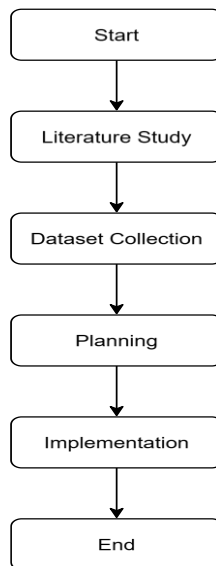
Khusus pada penyakit buah delima, literatur internasional terbaru sudah mulai menyediakan dataset dan model yang lebih spesifik. Misalnya, tersedia dataset terstandar untuk penyakit buah delima beserta kajian model DL yang diuji pada beberapa kelas penyakit, yang dapat dijadikan benchmark (Pakruddin & Hemavathy, 2025). Selain itu, terdapat pendekatan DL yang memadukan mekanisme perhatian/kapsul atau optimasi parameter untuk meningkatkan ketahanan klasifikasi, serta studi hybrid CNN + metaheuristic optimization untuk klasifikasi penyakit buah delima (Sajitha et al., 2024; Sameera & Deshpande, 2024). Temuan-temuan ini menunjukkan arah riset yang menekankan bukan hanya akurasi, tetapi juga ketahanan terhadap variasi citra dan pemilihan parameter yang lebih sistematis.

Berdasarkan peta riset tersebut, celah penelitian yang perlu ditegaskan dalam penelitian ini adalah masih terbatas kajian yang secara eksplisit mengevaluasi VGG16 pada klasifikasi penyakit buah delima dengan Grid Search yang dirancang ketat (ruang parameter jelas, skema validasi transparan, serta pelaporan computational cost), dan

masih kurang pelaporan yang menekankan reproduibilitas (konfigurasi parameter final, seed, dan protokol evaluasi). Dengan demikian, penelitian ini perlu diposisikan sebagai kontribusi yang menyediakan pipeline yang reproduibel untuk hyperparameter optimization berbasis Grid Search pada VGG16, serta pembuktian kinerja yang dibandingkan dengan konfigurasi baseline dan/atau setingan alternatif yang relevan.

3. Metodologi Penelitian

Dalam pelaksanaan penelitian ini, diterapkan perancangan metode terlebih dahulu untuk meminimalisir terjadinya kesalahan dalam proses penelitian. Tahapan metodologi penelitian dapat dilihat pada Gambar 5.



Gambar 5. Tahapan Metodologi Penelitian

Studi Literatur

Pada tahapan ini, permasalahan yang diidentifikasi melalui kajian literatur mencakup sulitnya mengklasifikasikan kualitas buah delima yang sehat dan terjangkit penyakit melalui citra. Selain studi literatur, penelitian ini juga melakukan wawancara secara langsung dengan Balai Perbenihan Tanaman Hutan Wilayah I Palembang sebagai *stakeholder* utama yang membantu validasi *dataset* dan permasalahan yang ada.

Pengumpulan Dataset

Pada tahap ini, dilakukan proses pengumpulan *dataset* berupa citra 2 dimensi yang diperoleh dari *platform* Kaggle dengan judul “Pomegranate Fruit Diseases [Image] Dataset”, di mana *dataset* ini terdiri dari 5 *class* yaitu 886 data *Alternaria*, 116 data *Anthraco*nose, 966 data *Bacterial Blight*, 631 data *Cercospora*, dan 1450 data *Healthy* dengan jumlah keseluruhan mencapai 5099 data. *Dataset* ini diambil oleh Pakruddin B dan

Hemavathy R pada bulan Juli dan Oktober 2023 dari berbagai daerah seperti Ballari, Bangalore, Bagalkote, dan beberapa daerah sekitar lainnya. Selain itu, *dataset* ini dipakai pada penelitian yang berjudul “*Pomegranate Fruit Diseases Dataset for Deep Learning Models*” pada tahun 2023.

Perancangan

Pada tahapan ini dilakukan proses perancangan sistem yang dibutuhkan untuk penelitian klasifikasi penyakit pada buah delima menggunakan CNN dan arsitektur VGG-16. Perancangan diawali dengan mempersiapkan *dataset* penyakit buah delima yang dibutuhkan seperti *Alternaria*, *Anthracoze*, *Bacterial Blight*, *Cercospora*, dan *Healthy*. Kemudian, *dataset* yang telah terkumpul tersebut dilakukan *preprocessing* seperti *resizing* yaitu menyamakan ukuran citra agar konsisten dengan *input* model pada arsitektur VGG-16 yaitu 224×224 px. Setelah itu, *dataset* yang telah melalui tahap *preprocessing* dilakukan data *augmentation* dengan melakukan *rotation* (30°), *flipping* (*horizontal* dan *vertical*), *cropping* (20%) dan *scaling* (20%) pada *class dataset* dengan jumlah data yang sedikit seperti, *Alternaria*, *Anthracoze*, *Bacterial Blight*, *Cercospora* dan *Healthy*. Setelah *dataset* memiliki jumlah data yang seimbang, *dataset* dilakukan *split* dengan proporsi 70% data *training*, 20% data *validation* dan 10% data *testing*, seperti yang dapat dilihat pada Tabel 4. Selanjutnya, *dataset* tersebut akan digunakan untuk proses *training* dan *validation* menggunakan model klasifikasi CNN dengan arsitektur VGG-16.

Tabel 1. *Splitting Dataset*

Data	Proporsi	Jumlah Data
<i>Training</i>	70%	5075
<i>Validation</i>	20%	1450
<i>Testing</i>	10%	725

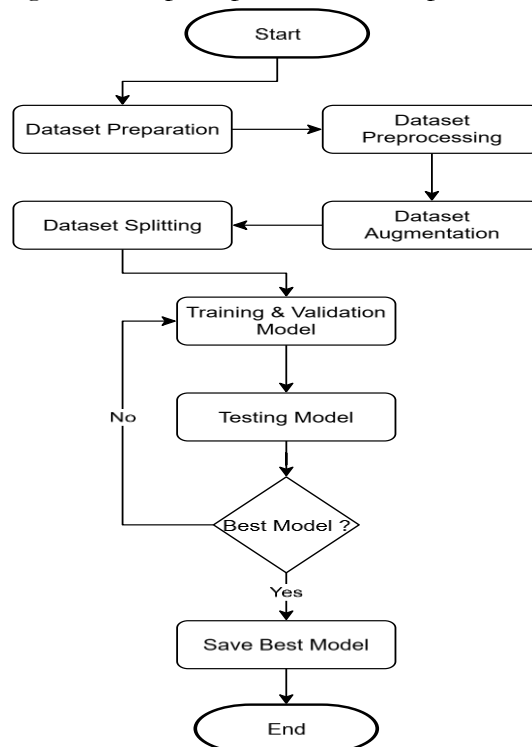
Dalam proses *training* model klasifikasi CNN dengan arsitektur VGG-16 akan dilakukan penyesuaian parameter model menggunakan metode *hyperparameter tuning*. Beberapa konfigurasi *hyperparameter* pada penelitian ini dapat dilihat pada Tabel 4.

Tabel 2. *Konfigurasi Hyperparameter Tuning*

No	<i>Hyperparameter</i>	Konfigurasi <i>Hyperparameter</i>
1	<i>Epoch</i>	50 dan 100
2	<i>Batch Size</i>	16 dan 32
3	<i>Learning Rate</i>	0.001 dan 0.0001
4	<i>Optimizer</i>	<i>Adam</i>

Setelah selesai proses *training* dan *validation*, model klasifikasi buah delima diuji dengan menggunakan data *testing* untuk melihat performa model klasifikasi saat bertemu

dengan data baru. Apabila model berhasil mengklasifikasikan penyakit buah delima dengan *accuracy* yang tinggi, baik dan benar, maka model klasifikasi tersebut akan disimpan dan menjadi *best model final*. Namun, apabila model gagal mengklasifikasikan penyakit buah delima, model tersebut akan dilakukan proses *training* dan *validation* ulang dengan *hyperparameter tuning* yang berbeda hingga model dapat mengklasifikasikan penyakit buah delima dengan baik, benar dan memiliki performa *accuracy* klasifikasi yang tinggi. Skema perancangan sistem pada penelitian ini dapat dilihat pada Gambar 6.



Gambar 6. Skema Perancangan Model

Implementasi

Pada tahapan ini dilakukan implementasi dari sistem yang telah dirancang sebelumnya agar sistem dapat melakukan klasifikasi penyakit pada buah delima dengan memanfaatkan data *training* sebanyak 70% untuk proses pelatihan model, data *validation* sebanyak 20% untuk mengukur performa dan optimalisasi parameter selama proses pelatihan, serta data *testing* sebanyak 10% untuk menguji kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya.

Pengujian

Pada tahapan ini, sistem yang telah dibuat sebelumnya akan melakukan uji coba terhadap data uji. Setelah tahapan uji coba, hasil pengujian dihitung untuk mendapatkan tingkat keberhasilan metode yang digunakan dengan *Confusion Matrix* dalam menghitung

nilai *precision*, *recall*, *accuracy*, dan *f1-score* yang dapat dilihat pada Persamaan (1), (2), (3), dan (4).

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (2)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (3)$$

$$F1 - score = 2 \times \frac{precision \times recall}{precision + recall} \quad (4)$$

Keterangan:

TP: Jumlah data positif (*defect*) yang terklasifikasi dengan benar oleh sistem

TN: Jumlah data negatif (*non-defect*) yang terklasifikasi dengan benar oleh sistem

FN: Jumlah data negatif (*non-defect*) yang terklasifikasi dengan salah oleh sistem

FP: Jumlah data positif (*defect*) yang terklasifikasi dengan salah oleh sistem.

4. Hasil dan Pembahasan

Bagian ini bertujuan untuk mengoptimasi *Hyperparameter* CNN dengan arsitektur VGG16 menggunakan *Grid Search* untuk klasifikasi penyakit pada buah delima. Pengujian ini dilakukan dengan delapan skenario pengujian yang menggunakan variasi seperti *learning rate* 0.001 dan 0.0001, *Batch size* 16 dan 32, *Epoch* 50 dan 100, serta Optimizer Adam, dan dievaluasi akurasi, presisi, *recall*, dan *f1-score* untuk masing masing scenario.

Sebelum memasuki tahap pelatihan model, semua citra di dalam dataset akan dilakukan augmentasi dan *pre-processing* yang Dimana ukuran gambar diubah menjadi 224x224 *pixel*. Pembagian dataset dibagi menjadi tiga subset yaitu 70% *training*, 20 % *validation*, 10% *testing*. Hasil dari pengujian dapat dilihat pada Tabel 3.

Tabel 3. Hasil Pengujian VGG16

Nomor	Learning Rate	Batch Size	Epoch	Train Accuracy	Validation Accuracy	Testing Accuracy	Precision	Recall	F1-Score
1	0.001	16	50	97.93%	98%	97.95%	0.980	0.979	0.979
2	0.0001	16	50	99.33%	98.76%	99.18%	0.992	0.992	0.992
3	0.001	32	50	98.93%	98.41%	98.77%	0.988	0.988	0.988
4	0.0001	32	50	99.72%	99.1%	98.77%	0.988	0.988	0.988
5	0.001	16	100	98.36%	98.55%	98.36%	0.984	0.984	0.984
6	0.0001	16	100	99.35%	98.69%	99.18%	0.992	0.992	0.992
7	0.001	32	100	98.56%	98.62%	98.08%	0.981	0.981	0.981
8	0.0001	32	100	99.84%	99.59%	99.59%	0.996	0.996	0.996

Berdasarkan hasil percobaan dengan 8 skenario pelatihan model CNN menggunakan arsitektur VGG16 dengan konfigurasi *hyperparameter*, maka untuk skenario

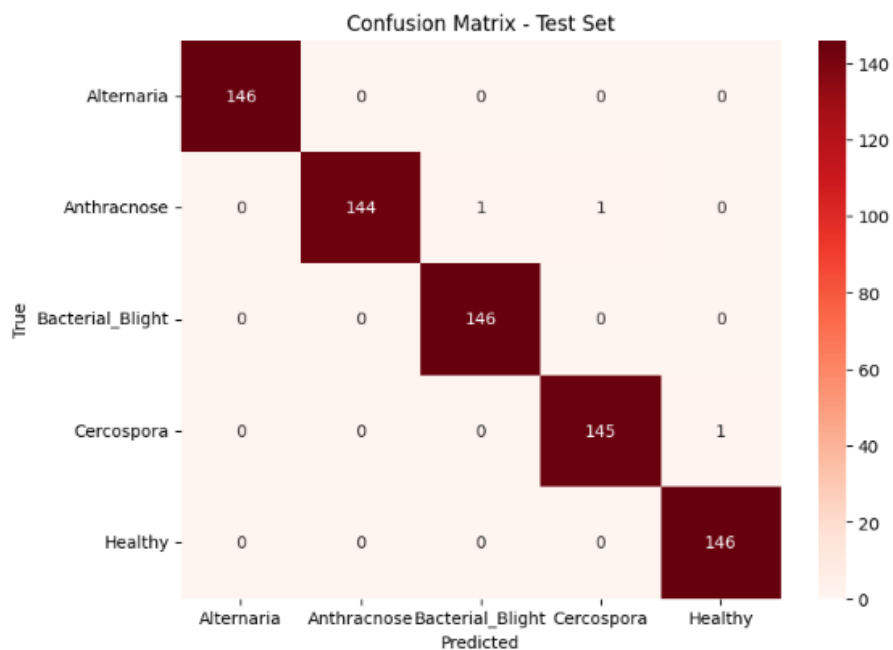
yang terbaik ada pada skenario ke delapan dengan konfigurasi *hyperparameter* (Epoch = 100, Batch Size = 32, Learning Rate = 0.0001, Optimizer = Adam) yang menghasilkan performa yang sangat baik pada klasifikasi lima jenis penyakit buah delima yaitu *Alternaria*, *Anthracnose*, *Bacterial Blight*, *Cercospora* dan sehat. Berdasarkan dari hasil training pada gambar 7, model mencapai akurasi validasi sebesar 99.59% pada epoch ke-91, dengan *train accuracy* sebesar 99.84% dan *train loss* yang sangat rendah yaitu 0.0046. hal ini dapat menunjukkan bahwa model mampu mempelajari pola penyakit buah delima dengan sangat baik tanpa mengalami *overfitting* yang signifikan.

```
Epoch 91/100 - Train: 100%|██████████| 159/159 [00:32<00:00, 4.87it/s]
Epoch 91/100 - Val: 100%|██████████| 46/46 [00:05<00:00, 8.78it/s]

New best model saved at epoch 91 - val_acc: 0.9959
Epoch 91/100 | Train Loss: 0.0046 | Train Acc: 0.9984 | Val Loss: 0.0208 | Val Acc: 0.9959 | Epoch Time: 37.9s
```

Gambar 7. Hasil Training Skenario 8 Epoch 91

Confusion matrix dengan hasil terbaik dapat dilihat pada gambar 8, berdasarkan *confusion matrix* terlihat bahwa model mampu mengklasifikasikan penyakit buah delima dengan benar sebagian sampel pada *test set*. Terdapat 146 *true positive* yang berhasil di prediksi benar pada penyakit *alternaria*, terdapat 144 *true positive* dan 2 yang salah pada penyakit *Anthracnose*, terdapat 146 *true positive* yang berhasil di prediksi benar pada penyakit *Bacterial Blight*, terdapat 145 *true positive* dan 1 yang salah pada penyakit *Cercospora*, dan terdapat 146 *true positive* yang berhasil di prediksi benar pada buah delima sehat.



Gambar 8. Confusion Matrix Skenario 8

Analisis berbasis confusion matrix (Gambar 8) menunjukkan sebagian besar sampel uji terklasifikasi benar pada seluruh kelas. Kesalahan yang muncul relatif kecil (misalnya pada kelas Anthracnose terdapat 2 prediksi yang salah, dan pada kelas Cercospora terdapat 1 prediksi yang salah), sementara kelas lain menunjukkan prediksi benar yang dominan. Untuk memperkuat pembahasan, perlu disajikan confusion matrix ternormalisasi dan contoh citra yang salah klasifikasi untuk mengidentifikasi pola kesalahan (misalnya kemiripan visual antar gejala penyakit, pencahayaan, atau latar belakang).

Implikasi praktis dari hasil ini adalah potensi penggunaan model sebagai komponen sistem bantu keputusan untuk identifikasi dini penyakit buah delima, sehingga petani dapat melakukan tindakan penanganan lebih cepat dan tepat. Keterbatasan yang perlu diakui mencakup ketergantungan pada dataset publik dan potensi perbedaan distribusi citra dengan kondisi lapangan nyata (domain shift), serta kebutuhan verifikasi bahwa augmentasi hanya diterapkan pada data latih agar tidak terjadi kebocoran data antar subset. Penelitian lanjutan dapat menguji generalisasi pada data lapangan (multi-lokasi/multi-perangkat), mengevaluasi arsitektur yang lebih ringan untuk implementasi mobile/edge, serta memperluas ruang hyperparameter (misalnya optimizer lain, regularisasi, dan scheduler) untuk memperoleh konfigurasi yang lebih robust.

5. Kesimpulan

Penelitian ini menunjukkan bahwa model Convolutional Neural Network (CNN) dengan arsitektur VGG16 mampu mengklasifikasikan penyakit buah delima secara efektif. Optimasi hyperparameter menggunakan Grid Search menghasilkan konfigurasi terbaik pada epoch 100, batch size 32, learning rate 0,0001, dan optimizer Adam, dengan kinerja pengujian mencapai accuracy 99,59% serta precision, recall, dan F1-score masing-masing sebesar 0,996, dengan tingkat kesalahan yang rendah pada seluruh kelas.

Hasil ini menunjukkan penggunaan teknik augmentasi pada data berhasil meningkatkan variasi data sehingga model dapat mengenali fitur citra dengan lebih baik dan menghindari *overfitting*. Serta optimasi *Hyperparameter* CNN pada arsitektur VGG16 menggunakan *Grid Search* untuk mengklasifikasi kan penyakit buah delima, berdasarkan konfigurasi terbaik dengan hasil tertinggi dalam tabel adalah konfigurasi dengan (Epoch = 100, Batch Size = 32, Learning Rate = 0.0001, Optimizer = Adam), yang mencapai akurasi sebesar 99.59%.

Secara praktis, temuan ini mengindikasikan bahwa penalaan hyperparameter yang terstruktur dapat meningkatkan reliabilitas model klasifikasi berbasis citra untuk mendukung identifikasi dini penyakit buah delima. Namun, penelitian ini memiliki keterbatasan, terutama ketergantungan pada dataset publik dan potensi perbedaan karakteristik citra dengan kondisi lapangan (variasi pencahayaan, latar belakang, dan perangkat akuisisi). Selain itu, prosedur augmentasi perlu dipastikan hanya diterapkan pada data pelatihan untuk menghindari kebocoran data antar subset.

Penelitian lanjutan disarankan untuk melakukan validasi pada data lapangan multi-lokasi dan multi-perangkat untuk menguji generalisasi, mengevaluasi arsitektur yang lebih ringan untuk implementasi mobile/edge, serta memperluas ruang optimasi hyperparameter (misalnya variasi optimizer/regularisasi/scheduler) agar model lebih robust dan reproduksibel.

6. Ucapan Terima Kasih

Saya mengucapkan puji dan syukur kepada Tuhan yang Maha Esa. serta terimakasih kepada semua pihak yang terlibat pada penelitian ini terutama co-autor dari Universitas Multi Data Palembang karena telah mendukung dan membimbing dalam penelitian ini.

7. Pernyataan Penulis

Penulis menyatakan bahwa tidak ada konflik kepentingan terkait publikasi artikel ini. Penulis menyatakan bahwa data dan makalah bebas dari plagiarisme serta penulis bertanggung jawab secara penuh atas keaslian artikel.

Bibliografi

- Abbas, R., Aamir, M., Saeed, F., Shankar, A., Kaur, J., Nadeem, R., Chauhan, A. S., Imran, A., Afzaal, M., & Kinki, A. B. (2025). Development and nutritional evaluation of pomegranate peel enriched bars. *PLOS ONE*, 20(1). <https://doi.org/10.1371/journal.pone.0315830>
- Abdel Rahman Shalaby, F., Misroni, & Wahidi, A. (2019). Pengklasifikasian dan penataan ulang buku di Perpustakaan SDN 15 Gelumbang. 1(1).
- Abdulkadrirov, R., Lyakhov, P., & Nagornov, N. (2023). Survey of optimization algorithms in modern neural networks. *Mathematics*, 11(11), 2466. <https://doi.org/10.3390/math11112466>
- Agustina, A., Yanto, F., Budianita, E., Iskandar, I., & Syafria, F. (2024). Klasifikasi penyakit tanaman padi menggunakan metode CNN arsitektur DenseNet-121 dan augmentasi data. *JOISIE (Journal of Information Systems and Informatics Engineering)*, 8(1), 124-134. <https://doi.org/10.35145/joisie.v8i1.4256>
- Anggraeni, A. M., Hermanto, T. I., & Nugroho, I. M. (2025). Klasifikasi penyakit daun singkong menggunakan convolutional neural network (CNN) dengan arsitektur VGG16 berbasis Android. *Jurnal Teknologi Terpadu*, 11(1), 53-59.
- Anhar, & Adi Putra, R. (2023). Perancangan dan implementasi self-checkout system pada toko ritel menggunakan convolutional neural network (CNN). *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 11(2), 466. <https://doi.org/10.26760/elkomika.v11i2.466>
- Simanjuntak, A., Lumbantoruan, R., Sianipar, K., Gultom, R., Simaremare, M., Situmeang, S., & Panggabean, E. (2024). Studi dan analisis hyperparameter tuning IndoBERT dalam pendeteksian berita palsu. *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, 13(1), 60-67. <https://doi.org/10.22146/jnteti.v13i1.8532>
- Asy Syifa, S., & Amelia Dewi, I. (2022). Arsitektur ResNet-152 dengan perbandingan optimizer Adam dan RMSProp untuk mendeteksi penyakit paru-paru. *MIND (Multimedia Artificial Intelligent Networking Database)*, 7(2), 139-150. <https://doi.org/10.26760/mindjournal.v7i2.139-150>
- Auliaddina, S., & Arifin, T. (2024). Penggunaan data augmentasi dan hyperparameter tuning dalam klasifikasi jenis batik menggunakan model CNN. *SISTEMASI: Jurnal Sistem Informasi*, 13(1), 114-128. <http://sistemasi.ftik.unisi.ac.id>
- Azmi, K., Defit, S., & Sumijan. (2023). Implementasi convolutional neural network (CNN) untuk klasifikasi batik tanah liat Sumatera Barat. *Jurnal Unitek*, 16(1).
- Bezdan, T., & Bacanin Dzakula, N. (2019). Convolutional neural network layers and architectures. *SINTEZA 2019*, 445-451. <https://doi.org/10.15308/sinteza-2019-445-451>
- Bock, S., & Weis, M. (2019). A proof of local convergence for the Adam optimizer. *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*. <https://doi.org/10.1109/IJCNN.2019.8852239>

- Chan, L., Tseng, Y., Liu, C., & Liang, C. (2021). Fermented pomegranate extracts protect against oxidative stress and aging of skin. *Journal of Cosmetic Dermatology*, 21. <https://doi.org/10.1111/jocd.14379>
- Kurniadi, D., Shidiq, R. M., & Mulyani, A. (2025). Perbandingan penggunaan optimizer dalam klasifikasi sel darah putih menggunakan convolutional neural network. *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, 14(1), 77-86. <https://doi.org/10.22146/jnteti.v14i1.17162>
- Dewi Linda, K. S., & Dwi Hartanto, A. (2024). Studi literatur mengenai klasifikasi citra kucing dengan menggunakan deep learning: Convolutional neural network (CNN). *Journal of Electrical Engineering and Computer (JEECOM)*, 129-137. <https://doi.org/10.33650/jeeecom.v4i2>
- Fitra Maulana, F., & Rochmawati, N. (2019). Klasifikasi citra buah menggunakan convolutional neural network. *JINACS (Journal of Informatics and Computer Science)*, 1(2), 104-108.
- Ghandi, S., Nugroho, I. M., & Ramadhan, Y. (2024). Penerapan metode convolutional neural network (CNN) dalam aplikasi pendeteksi penyakit daun tanaman kentang berbasis Android. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(5).
- Great Learning. (2025, February 12). Introduction to VGG16 - What is VGG16? <https://www.mygreatlearning.com/blog/introduction-to-vgg16/>
- Gulmez, B. (2025). A comprehensive review of convolutional neural networks based disease detection strategies in potato agriculture. *Potato Research*, 68(2), 1295-1329. <https://doi.org/10.1007/s11540-024-09786-1>
- Hong, T.-Y., & Chen, C.-C. (2023). Hyperparameter optimization for convolutional neural network by opposite-based particle swarm optimization and an empirical study of photomask defect classification. *Applied Soft Computing*, 148, 110904. <https://doi.org/10.1016/j.asoc.2023.110904>
- Jaiman, R. S., Purohit, J., Nakrani, B. R., Elangbam, P. D., & Pandya, K. S. (2024). Efficacy of fluopyram 250 g/L + trifloxystrobin 250 g/L SC against anthracnose disease of pomegranate. *Asian Research Journal of Agriculture*, 17(4), 1189-1201. <https://doi.org/10.9734/arja/2024/v17i4634>
- Jang, J. Y., Kim, D., Im, E., & Kim, N. D. (2024). Therapeutic potential of pomegranate extract for women's reproductive health and breast cancer. *Life*, 14(10), 1264. <https://doi.org/10.3390/life14101264>
- Jha, S. K., Misra, Y., Nyayadhis, R., Rawat, M. K., K, K. K., & Neelam, N. (2025). A CNN-driven image analysis approach for accurate detection of plant leaf diseases. *International Journal of Engineering, Science and Information Technology*, 5(4). <https://doi.org/10.52088/ijesty.v5i4.1299>
- Laksono, F. B., & Fathurohman, A. (2024). Plant disease detection with convolution neural network: Combination of VGG16 and ResNet34 architectures for leaf classification. *Jurnal Komputer dan Teknologi Informasi*, 2(2). <https://doi.org/10.26714/jkti.v2i2.13932>
- Liu, M., Yao, D., Liu, Z., Guo, J., & Chen, J. (2023). An improved Adam optimization algorithm combining adaptive coefficients and composite gradients based on randomized block coordinate descent. *Computational Intelligence and Neuroscience*, 2023, 4765891. <https://doi.org/10.1155/2023/4765891>
- Madhavan, M. V., Thanh, D. N. H., Khamparia, A., Pande, S., Malik, R., & Gupta, D. (2021). Recognition and classification of pomegranate leaves diseases by image processing and machine learning techniques. *Computers, Materials & Continua*, 66(3), 2939-2955. <https://doi.org/10.32604/cmc.2021.012466>
- Manjunatha, N., Sharma, J., Pokhare, S. S., Agarwal, R., Patil, P. G., Sirsat, J. D., Chakranarayan, M. G., Bicchal, A., Ukale, A. S., & Marathe, R. A. (2022). Characterization of Alternaria and

- Colletotrichum species associated with pomegranate (*Punica granatum* L.) in Maharashtra State of India. *Journal of Fungi*, 8(10), 1040. <https://doi.org/10.3390/jof8101040>
- Mansur, S. A., Deroyeen, A. F., Indriyanti, M. N., Annisak, A. K., Fajriati, D. R., & Amiruddin, M. (2022). Kandungan buah delima (*Punica granatum* L.) dalam perspektif Al-Qur'an, Sunnah, dan sains. *Proceedings of International Pharmacy Ulul Albab Conference and Seminar (PLANAR)*, 2, 69. <https://doi.org/10.18860/planar.v2i0.2128>
- Math, R. M., Dharwadkar, N. V., Sajjanar, V., & Deshmukh, M. N. (2024). Deep learning for precision grade-quality classification of pomegranates. *2024 International Conference on Innovation and Novelty in Engineering and Technology (INNOVA)*, 1-6. <https://doi.org/10.1109/INNOVA63080.2024.10846982>
- Morales-Hernandez, A., Van Nieuwenhuyse, I., & Rojas Gonzalez, S. (2023). A survey on multi-objective hyperparameter optimization algorithms for machine learning. *Artificial Intelligence Review*, 56(8), 8043-8093. <https://doi.org/10.1007/s10462-022-10359-2>
- Mumuni, A., & Mumuni, F. (2022). Data augmentation: A comprehensive survey of modern approaches. *Array*, 16, 100258. <https://doi.org/10.1016/j.array.2022.100258>
- Ningsih, N. P., Suryadi, E., Bakti, L. D., & Imran, B. (2022). Klasifikasi penyakit early blight dan late blight pada tanaman tomat berdasarkan citra daun menggunakan metode CNN berbasis website. *Jurnal Kecerdasan Buatan dan Teknologi Informasi (JKBTI)*, 1(3), 27-35.
- Noorizki, A. Z., & Kusumawati, W. I. (2023). Perbandingan performa algoritma VGG16 dan VGG19 melalui metode CNN untuk klasifikasi varietas beras. *Journal of Computer, Electronic, and Telecommunication*, 4(2). <https://doi.org/10.52435/complete.v4i2.387>
- Oktafanda, E. (2022). Klasifikasi citra kualitas bibit dalam meningkatkan produksi kelapa sawit menggunakan metode convolutional neural network (CNN). *Jurnal Informatika Ekonomi Bisnis*, 72-77. <https://doi.org/10.37034/infkeb.v4i3.143>
- Opara, I. K., Fawole, O. A., Kelly, C., & Opara, U. L. (2021). Quantification of on-farm pomegranate fruit postharvest losses and waste, and implications on sustainability indicators: South African case study. *Sustainability*, 13(9), 5168. <https://doi.org/10.3390/su13095168>
- Pakruddin, B., & Hemavathy, R. (2024a). A comprehensive standardized dataset of numerous pomegranate fruit diseases for deep learning. *Data in Brief*, 54, 110284. <https://doi.org/10.1016/j.dib.2024.110284>
- Pakruddin, B., & Hemavathy, R. (2024b). Development of a pomegranate fruit disease detection and classification model using deep learning. *Indian Journal of Agricultural Research*, 58, 1121-1130. <https://doi.org/10.18805/IJARE.A-6281>
- Pakruddin, B., & Hemavathy, R. (2025). Performance evaluation of deep learning models for multiclass disease detection in pomegranate fruits. *Indian Journal of Agricultural Research*. <https://doi.org/10.18805/ijare.a-6396>
- Paraijun, F., Nur Aziza, R., & Kuswardani, D. (2022). Implementasi algoritma convolutional neural network dalam mengklasifikasi kesegaran buah berdasarkan citra buah. *KILAT*, 11(1). <https://doi.org/10.33322/kilat.v11i1.1458>
- Paul, P., Manishankar, S., & Satheesh Kumar, R. (2024). Disease detection and severity prediction of pomegranate fruit. *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 1-7. <https://doi.org/10.1109/ICCCNT61001.2024.10725059>
- Pradana, M. A., Maulana, R. I., Putra, R. S., Subairi, S., & Anggraeny, F. T. (2023). Klasifikasi penyakit tanaman tomat menggunakan metode convolutional neural network (CNN) VGG16. *KERNEL*:

- Jurnal Riset Inovasi Bidang Informatika dan Pendidikan Informatika*, 4(2), 61-69. <https://doi.org/10.31284/j.kernel.2023.v4i2.6829>
- Pratama, G. A., Puspaningrum, E. Y., & Maulana, H. (2024). Convolutional neural network dan faster region convolutional neural network untuk klasifikasi kualitas biji kopi Arabika. *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, 12(3). <https://doi.org/10.23960/jitet.v12i3.4887>
- Rahman, F. F., & Pratiwi, N. (2025). Identifikasi penyakit tumbuhan tomat dan anggur menggunakan CNN dengan arsitektur VGG-16. *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 10(1), 270-279. <https://doi.org/10.29100/jupi.v10i1.5663>
- Rochmawati, N., Hidayati, H. B., Yamasari, Y., Tjahyaningtijas, A. P., Yustanti, W., & Prihanto, A. (2021). Analisa learning rate dan batch size pada klasifikasi COVID menggunakan deep learning dengan optimizer Adam. *JIEET: Journal Information Engineering and Educational Technology*, 5(2).
- Sajitha, P., Andrushia, A. D., Anand, N., Naser, M. Z., & Lubloy, E. (2024). A deep learning approach to detect diseases in pomegranate fruits via hybrid optimal attention capsule network. *Ecological Informatics*, 102859. <https://doi.org/10.1016/j.ecoinf.2024.102859>
- Sameera, P., & Deshpande, A. A. (2024). Disease detection and classification in pomegranate fruit using hybrid convolutional neural network with honey badger optimization algorithm. *International Journal of Food Properties*, 27(1), 815-837. <https://doi.org/10.1080/10942912.2024.2365927>
- Mashuri, A. S., Sunyoto, A., & Kusnawi. (2024). Klasifikasi penyakit pada daun cabai menggunakan arsitektur VGG16. *Journal of Electrical Engineering and Computer (JEECOM)*, 6(2). <https://doi.org/10.33650/jeeecom.v4i2>
- Saputro, W., & Khairudin, A. (2022). Klasifikasi kualitas mutu buah delima dengan menggunakan ekstraksi gray level co-occurrence matrix (GLCM) dan k-nearest neighbor (KNN). *JINTEKS*, 4(3).
- Shah, S. R., Qadri, S., Bibi, H., Shah, S. M. W., Sharif, M. I., & Marinello, F. (2023). Comparing Inception V3, VGG 16, VGG 19, CNN, and ResNet 50: A case study on early detection of a rice disease. *Agronomy*, 13(6), 1633. <https://doi.org/10.3390/agronomy13061633>
- Sitaula, C., & Hossain, M. B. (2021). Attention-based VGG-16 model for COVID-19 chest X-ray image classification. *Applied Intelligence*, 51(5), 2850-2863. <https://doi.org/10.1007/s10489-020-02055-x>
- Sodre, L. I., Gall, M. E. C., Elias, M. de B., de Oliveira, L. O., Lobo, F. A. T. F., Carias, R. B. V., & Teodoro, A. J. (2025). Osteogenic effects of bioactive compounds found in fruits on mesenchymal stem cells: A review. *Nutrition Reviews*, 83(4), 675-691. <https://doi.org/10.1093/nutrit/nuae209>
- Sun, S., Cao, Z., Zhu, H., & Zhao, J. (2019). A survey of optimization methods from a machine learning perspective. *arXiv*. <https://arxiv.org/abs/1906.06821>
- Taye, M. M. (2023). Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions. *Computation*, 11(3), 52. <https://doi.org/10.3390/computation11030052>
- Tirro, G., Conti Taguali, S., Pane, A., Riolo, M., Ezra, D., & Cacciola, S. O. (2024). Outbreak of Alternaria black spot of pomegranate (*Punica granatum* L.) in Italy as a consequence of unusual climatic conditions. *Plants*, 13(14), 2007. <https://doi.org/10.3390/plants13142007>
- Valero-Mendoza, A. G., Melendez-Renteria, N. P., Chavez-Gonzalez, M. L., Flores-Gallegos, A. C., Wong-Paz, J. E., Govea-Salas, M., Zugasti-Cruz, A., & Ascacio-Valdes, J. A. (2023). The whole pomegranate (*Punica granatum* L), biological properties and important findings: A review. *Food Chemistry Advances*, 2, 100153. <https://doi.org/10.1016/j.focha.2022.100153>

- Wardani, K. R. R., Suryalim, H., Engel, V. J. L., & Christian, H. (2023). Analisis pemilihan optimizer dalam arsitektur convolution neural network VGG16 dan Inception untuk sistem pengenalan wajah. *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, 9(2).
- Wibawa Rahayu, P., Sudipa, I. G. I., Suryani, Surachman, A., Achmad, R., Darmawiguna, I. G. M., Slamet, I., Harlina, S., & Maysanjaya, I. M. D. (2024). *Buku ajar data mining* (Efitra, Ed.). PT. Sonpedia Publishing Indonesia. <https://www.researchgate.net/publication/377415198>
- Wowor, L. D. (2025). Implementation of grid search method for optimization in convolutional neural network model: Penerapan grid search method untuk optimasi pada model convolutional neural network. *Jurnal Teknik Informatika*.
- Wu, F., & Lin, H. (2022). Effect of transfer learning on the performance of VGGNet-16 and ResNet-50 for the classification of organic and residual waste. *Frontiers in Environmental Science*, 10, 1043843. <https://doi.org/10.3389/fenvs.2022.1043843>
- Wu, M., Liu, X., Gui, N., Yang, X., Tu, J., Jiang, S., & Zhao, Q. (2023). Prediction of the remaining time and time interval of pebbles in pebble bed HTGRs aided by CNN via DEM datasets. *Nuclear Engineering and Technology*, 55(1), 339-352. <https://doi.org/10.1016/j.net.2022.09.019>
- Yang, S., Xiao, W., Zhang, M., Guo, S., Zhao, J., & Shen, F. (2023). Image data augmentation for deep learning: A survey. *arXiv*. <https://arxiv.org/abs/2204.08610>
- Yuliany, S., Aradea, & Nur Rachman, A. (2022). Implementasi deep learning pada sistem klasifikasi hama tanaman padi menggunakan metode convolutional neural network (CNN). *Jurnal Buana Informatika*, 13(1), 54-65.