



## Evaluasi dan Perbandingan Model CNN Dan *Transfer Learning* Dalam Klasifikasi Kematangan Buah Kelapa Sawit

Naufal Budiman<sup>1✉</sup>, Ringga Chandra Perdana<sup>2</sup>

<sup>1,2</sup> Fakultas Sains dan Teknologi, Universitas Adiwangsa Jambi, Kota Jambi, 36139, Indonesia

[Naufalbdmn@gmail.com](mailto:Naufalbdmn@gmail.com)

### Abstract

*The identification of oil palm fruit ripeness is an important factor in maintaining harvest quality and improving palm oil productivity. Manual identification processes still have limitations, including subjectivity and inconsistency in assessment results. This study aims to evaluate and compare the performance of a Baseline Convolutional Neural Network (CNN) model with several transfer learning architectures, namely EfficientNetB3, ResNet50, and DenseNet121, for oil palm fruit ripeness classification. The dataset consisted of 302 original images of oil palm fruits categorized into three classes: ripe, unripe, and rotten. To prevent data leakage, the dataset was first divided using a stratified split into training, validation, and testing sets before data augmentation was applied exclusively to the training set. The augmentation techniques included rotation, translation, zooming, brightness adjustment, and horizontal flipping to increase data variability and reduce overfitting. All models were trained using an input size of  $224 \times 224$  pixels, the Adam optimizer, and categorical cross-entropy as the loss function. Model performance was evaluated using accuracy, precision, recall, and F1-score metrics to assess the classification capability of each model. In addition, confusion matrix analysis was conducted to identify classification error patterns across the ripeness categories. The results indicate that transfer learning models outperformed the Baseline CNN model. DenseNet121 achieved the best overall performance, followed by EfficientNetB3 and ResNet50. These findings demonstrate that transfer learning is an effective approach for oil palm fruit ripeness classification, particularly when working with limited datasets. Nevertheless, further studies using larger and more diverse datasets are recommended to improve model generalization capabilities.*

**Keywords:** *Convolutional Neural Network, Transfer Learning, EfficientNetB3, ResNet50, DenseNet121*

### Abstrak

Identifikasi kematangan buah kelapa sawit merupakan salah satu faktor penting dalam menjaga kualitas hasil panen dan meningkatkan produktivitas minyak kelapa sawit. Proses identifikasi manual masih memiliki keterbatasan berupa subjektivitas dan inkonsistensi hasil. Penelitian ini bertujuan untuk mengevaluasi dan membandingkan performa model Baseline CNN dengan beberapa arsitektur transfer learning, yaitu EfficientNetB3, ResNet50, dan DenseNet121 dalam klasifikasi kematangan buah kelapa sawit. Dataset yang digunakan terdiri dari 302 citra asli buah kelapa sawit yang dibagi ke dalam tiga kelas, yaitu matang, mentah, dan busuk. Untuk menghindari data leakage, dataset terlebih dahulu dibagi menggunakan stratified split menjadi data training, validation, dan testing sebelum augmentasi diterapkan hanya pada data training. Teknik augmentasi yang digunakan meliputi rotasi, pergeseran, zoom, penyesuaian kecerahan, dan horizontal flipping untuk meningkatkan variasi data serta mengurangi overfitting. Seluruh model dilatih menggunakan ukuran input  $224 \times 224$  piksel dengan optimizer Adam dan fungsi loss categorical crossentropy. Evaluasi model dilakukan menggunakan accuracy, precision, recall, dan F1-score untuk mengukur kemampuan klasifikasi masing-masing model. Selain itu, analisis confusion matrix digunakan untuk mengidentifikasi pola kesalahan klasifikasi pada setiap kelas kematangan buah kelapa sawit. Hasil penelitian menunjukkan bahwa model transfer learning memberikan performa yang lebih baik dibandingkan baseline CNN. DenseNet121 menghasilkan performa terbaik dibandingkan model lainnya, diikuti oleh EfficientNetB3 dan ResNet50. Hasil tersebut menunjukkan bahwa transfer learning efektif digunakan pada klasifikasi kematangan buah kelapa sawit dengan dataset terbatas. Namun demikian, penelitian lebih lanjut dengan dataset yang lebih besar dan beragam masih diperlukan untuk meningkatkan kemampuan generalisasi model.

**Kata kunci:** *Convolutional Neural Network, Transfer Learning, EfficientNetB3, ResNet50, DenseNet121.*

*Jutekom is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.*

## 1. Pendahuluan

Industri kelapa sawit merupakan salah satu sektor strategis yang berperan penting dalam perekonomian Indonesia [1]. Sebagai salah satu produsen utama minyak kelapa sawit di dunia, kualitas hasil panen menjadi faktor krusial yang sangat menentukan nilai ekonomi dan daya saing produk di pasar global [2]. Salah satu aspek penting dalam menjaga kualitas tersebut adalah ketepatan dalam menentukan tingkat kematangan buah kelapa sawit pada saat panen. Penentuan kematangan yang tidak tepat dapat berdampak pada rendahnya rendemen minyak serta menurunnya kualitas produk akhir [3].

Dalam praktik di lapangan, proses identifikasi kematangan buah kelapa sawit masih banyak dilakukan secara manual oleh tenaga kerja berdasarkan pengamatan visual, seperti perubahan warna dan jumlah buah yang terlepas dari tandan. Metode ini memiliki sejumlah keterbatasan, di antaranya bersifat subjektif, bergantung pada pengalaman pekerja, serta rentan terhadap inkonsistensi hasil, terutama dalam kondisi lingkungan yang beragam [4]. Selain itu, keterbatasan jumlah tenaga kerja di sektor perkebunan semakin memperkuat kebutuhan akan sistem otomatis yang mampu membantu proses identifikasi secara cepat, akurat, dan konsisten.

Kemajuan teknologi *Artificial Intelligence*, khususnya *Deep Learning* kini memungkinkan otomatisasi analisis tanaman secara signifikan. Melalui jaringan saraf tiruan, informasi visual buah dapat diproses secara akurat untuk mendeteksi tingkat kematangan serta kualitasnya guna meningkatkan standar produksi [5]. Penelitian deteksi objek telah menjadi fokus utama dalam beberapa tahun terakhir. Dengan kekuatan alat pembelajaran yang kuat, kita dapat dengan mudah mendeteksi dan mempelajari fitur-fitur yang lebih dalam [6]. *Convolutional Neural Network* (CNN) adalah salah satu arsitektur *Deep Learning* (DL) yang paling banyak digunakan dalam pengolahan gambar. CNN memiliki tingkat abstraksi yang tinggi dan kemampuan untuk secara otomatis mempelajari pola yang ada dalam gambar [7]. Alasan utama penggunaan CNN adalah kemampuannya dalam melakukan ekstraksi fitur secara otomatis. Berbeda dengan teknik *Machine Learning* tradisional yang memerlukan ekstraksi fitur manual oleh manusia (seperti SIFT atau HOG), CNN mampu menarik fitur-fitur penting langsung dari dataset gambar mentah tanpa

keterlibatan pengguna [8]. Hal ini menjadikan CNN sebagai model pembelajaran yang paling akurat untuk tugas identifikasi, klasifikasi, dan pengenalan objek di berbagai bidang.

Meskipun CNN sangat kuat, melatih model dari awal (*from scratch*) membutuhkan sumber daya komputasi yang besar, dataset yang masif, serta waktu pelatihan yang lama. Oleh karena itu, pendekatan *Transfer Learning* kini semakin populer dalam penelitian kecerdasan buatan [8]. *Transfer learning* adalah strategi di mana model yang telah dikembangkan untuk satu tugas digunakan kembali sebagai titik awal untuk tugas kedua yang terkait [9]. Pendekatan transfer learning memiliki beberapa keunggulan utama, seperti penggunaan model *pretrained* yang telah dilatih pada dataset besar seperti ImageNet sehingga model tidak memulai pelatihan dari awal. Selain itu, metode ini mampu mengurangi waktu pelatihan dan kebutuhan komputasi karena model sudah memahami fitur dasar citra. Transfer learning juga sangat efektif untuk dataset berukuran terbatas, karena pengetahuan dari domain sumber dapat membantu meningkatkan akurasi sekaligus mengurangi risiko *overfitting* pada domain target [10].

Dalam penelitian ini, digunakan tiga arsitektur transfer learning sebagai model pembandingan dalam klasifikasi kematangan buah kelapa sawit, yaitu EfficientNetB3, ResNet50, dan DenseNet121. EfficientNetB3 dikenal mampu menghasilkan akurasi tinggi dengan efisiensi komputasi yang baik melalui metode *compound scaling* yang menyeimbangkan kedalaman, lebar, dan resolusi jaringan [11]. ResNet50 memiliki keunggulan pada penggunaan residual connection yang membantu proses pelatihan jaringan mendalam agar terhindar dari masalah vanishing gradient dan penurunan performa [12]. Sementara itu, DenseNet121 memanfaatkan *dense connectivity* yang memungkinkan setiap lapisan menerima informasi dari seluruh lapisan sebelumnya sehingga meningkatkan penggunaan kembali fitur dan efisiensi parameter [10].

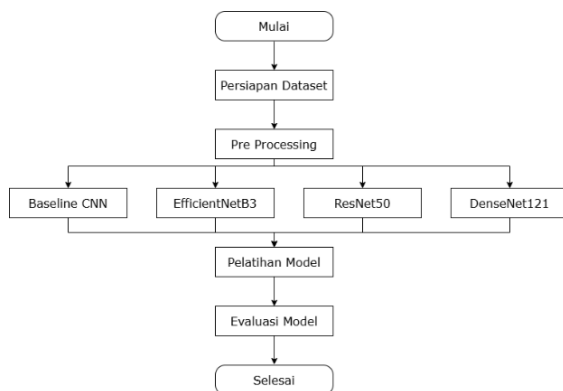
Meskipun berbagai penelitian telah menerapkan CNN dan transfer learning pada klasifikasi citra pertanian, penelitian yang secara khusus mengevaluasi dan membandingkan beberapa arsitektur transfer learning modern pada klasifikasi kematangan buah kelapa sawit menggunakan dataset terbatas masih relatif terbatas. Selain itu, sebagian besar penelitian

sebelumnya lebih berfokus pada performa akurasi tanpa membahas kemampuan generalisasi dan kompleksitas model secara komprehensif. Oleh karena itu, kontribusi utama penelitian ini terletak pada evaluasi komprehensif beberapa arsitektur transfer learning modern dalam klasifikasi kematangan buah kelapa sawit dengan mempertimbangkan performa klasifikasi, kemampuan generalisasi, serta efisiensi komputasi model.

Penelitian ini bertujuan untuk mengevaluasi dan membandingkan performa beberapa arsitektur *deep learning* dalam klasifikasi kematangan buah kelapa sawit, yaitu *Baseline CNN*, *EfficientNetB3*, *ResNet50*, dan *DenseNet121*. Evaluasi dilakukan berdasarkan beberapa metrik pengujian seperti *accuracy*, *precision*, *recall*, dan *F1-score* untuk mengetahui kemampuan masing-masing model dalam mengklasifikasikan tingkat kematangan buah kelapa sawit. Selain itu, penelitian ini juga bertujuan untuk menganalisis kemampuan generalisasi dan efektivitas setiap arsitektur dalam menangani dataset citra buah kelapa sawit sehingga dapat diperoleh model yang paling optimal untuk diterapkan pada sistem klasifikasi otomatis berbasis *deep learning*.

## 2. Metodologi Penelitian

Penelitian ini dimulai dengan mempersiapkan dataset buah kelapa sawit dengan 3 kelas : matang, mentah dan busuk lalu dilanjutkan ke tahap pre processing, lalu perancangan model *baseline CNN* dan 3 model *transfer learning* : *EfficientNetB3*, *ResNet50* dan *DenseNet121*. Setelah itu dilakukannya proses pelatihan dan evaluasi model. Untuk diagram alur penelitian dapat dilihat pada gambar 1.



Gambar 1. Diagram Alur Penelitian

### 2.1. Persiapan Dataset

Dataset yang akan digunakan pada penelitian ini adalah foto-foto buah tandan kelapa sawit dengan 3 kelas : matang, mentah dan busuk. Dataset diambil langsung dari kebun kelapa sawit Balai Pelatihan Jambi dengan menggunakan kamera handphone. Total dataset buah kelapa sawit berjumlah 302 gambar, yang terdiri dari 100 gambar buah kelapa sawit dengan kondisi matang, yang dapat dilihat contoh nya pada gambar 2.



Gambar 2. Buah Kelapa Sawit Matang

103 gambar buah kelapa sawit dengan kondisi mentah, yang dapat dilihat contohnya pada gambar 3.



Gambar 3. Buah Kelapa Sawit Mentah

Dan 99 gambar buah kelapa sawit dengan kondisi busuk, yang dapat dilihat contoh nya pada gambar 4.



Gambar 4. Buah Kelapa Sawit Busuk

## 2.2. Pre Processing

Pada tahap preprocessing, dataset asli yang terdiri dari 302 gambar terlebih dahulu dibagi menggunakan metode stratified split menjadi data training, validation, dan testing dengan rasio sekitar 70:20:10. Pembagian dataset dilakukan sebelum proses augmentasi untuk menghindari data leakage serta memastikan bahwa citra hasil augmentasi dari gambar yang sama tidak muncul pada data validation maupun testing.

Setelah proses pembagian dataset selesai, akan dilakukan proses augmentasi untuk meningkatkan jumlah dan variasi data guna mengurangi risiko *overfitting* pada model *deep learning* [13]. Augmentasi hanya diterapkan pada data training menggunakan ImageDataGenerator dari Keras. Teknik augmentasi yang digunakan meliputi rotation hingga  $\pm 10^\circ$ , horizontal dan vertical shifting sebesar 10%, brightness adjustment sebesar 80%–120%, zoom hingga 10%, serta horizontal flipping. Pendekatan ini bertujuan untuk meningkatkan variasi data training dan membantu model mempelajari pola visual yang lebih beragam sehingga dapat mengurangi risiko *overfitting*. Data validation dan testing tetap menggunakan citra asli tanpa augmentasi agar evaluasi model dapat merepresentasikan performa model pada data yang benar-benar baru dan belum pernah dilihat sebelumnya.

Tahap selanjutnya adalah resizing seluruh gambar menjadi  $224 \times 224$  piksel dan normalisasi nilai piksel dari rentang [0,255] menjadi [0,1] agar sesuai dengan input model CNN dan membantu stabilitas proses pelatihan.

## 2.3. Baseline CNN

Model *baseline* CNN ini dibangun sebagai tolak ukur untuk menilai keberhasilan model model yang menggunakan *transfer learning*. Struktur dan konfigurasi lapisan pada model *baseline* CNN dapat dilihat pada tabel 1.

Tabel 1. Struktur Model Baseline CNN

| Lapisan         | Konfigurasi Lapisan     |
|-----------------|-------------------------|
| Konvolusi       | 16 filter, 3x3 dan ReLu |
| Max-pooling     | 2x2                     |
| Konvolusi       | 32 filter, 3x3 dan ReLU |
| Max-pooling     | 2x2                     |
| Konvolusi       | 16 filter, 3x3 dan ReLu |
| Max-pooling     | 2x2                     |
| Flatten         | 14.400 Neuron           |
| Fully connected | 128 Neuron              |
| Output layer    | Softmax 3 kelas         |

Model yang digunakan merupakan arsitektur CNN konvensional yang terdiri dari tiga blok *convolution* dan *max-pooling*, kemudian dilanjutkan dengan lapisan *flatten*, *dense*, dan *output*. Setiap lapisan *convolution* menggunakan kernel berukuran  $3 \times 3$  dengan fungsi aktivasi ReLU, sedangkan *max-pooling* berukuran  $2 \times 2$  digunakan untuk mengurangi dimensi spasial fitur. Jumlah filter pada masing-masing lapisan *convolution* adalah 16, 32, dan 16, sehingga model dapat mengekstraksi fitur visual dari tingkat sederhana hingga lebih kompleks secara bertahap.

Lapisan *flatten* berfungsi mengubah *feature map* dua dimensi menjadi vektor satu dimensi sebelum diproses oleh lapisan *fully connected*. Selanjutnya, lapisan *dense* dengan 128 *neuron* digunakan untuk mengintegrasikan fitur hasil ekstraksi. Pada bagian akhir, lapisan *output* menggunakan fungsi aktivasi softmax dengan tiga *neuron* untuk menghasilkan probabilitas prediksi pada kelas matang, mentah, dan busuk.

## 2.4. EfficientNetB3

EfficientNetB3 merupakan arsitektur CNN modern yang menerapkan metode *compound scaling*, yaitu pendekatan yang menyesuaikan kedalaman, lebar, dan resolusi input jaringan secara seimbang untuk meningkatkan performa model dengan kompleksitas yang tetap efisien [14]. Model ini menggunakan bobot *pretrained* dari ImageNet sehingga dapat memanfaatkan *transfer learning* dalam proses pelatihan. Struktur dan konfigurasi pada model EfficientNetB3 setelah dimodifikasi dapat dilihat pada tabel 2.

Tabel 2. Struktur Model EfficientNetB3

| Lapisan         | Konfigurasi Lapisan                                   |
|-----------------|-------------------------------------------------------|
| Input layer     | $224 \times 224 \times 3$                             |
| EfficientNetB3  | Pretrained ImageNet, tanpa top layer, pooling = 'avg' |
| Fully connected | 128 neuron, ReLU, L2 Regularization (0.0001)          |
| Fully connected | 64 neuron, ReLU, L2 Regularization (0.0001)           |
| Dropout         | 0.5                                                   |
| Output layer    | Softmax 3 kelas                                       |

Pada implementasinya, ukuran input  $224 \times 224 \times 3$  digunakan untuk menyesuaikan dimensi citra dengan kebutuhan arsitektur EfficientNetB3 pada proses *transfer learning*. Meskipun EfficientNetB3 secara default mendukung resolusi input yang lebih tinggi, penggunaan ukuran  $224 \times 224$  dipilih untuk menjaga konsistensi dimensi input antar model pembandingan serta mengurangi kompleksitas komputasi dan waktu pelatihan.

tanpa mengurangi kemampuan model dalam mengekstraksi fitur penting dari citra buah kelapa sawit. Lalu untuk bagian atas jaringan dihapus dan diganti dengan global *average pooling* untuk mengubah fitur hasil ekstraksi menjadi vektor satu dimensi. Proses *fine-tuning* dilakukan pada sepuluh lapisan terakhir *backbone* EfficientNetB3, kecuali lapisan *BatchNormalization* yang tetap dibekukan untuk menjaga stabilitas distribusi data. Setelah itu, ditambahkan dua *fully connected layer* masing-masing 128 dan 64 *neuron* dengan aktivasi ReLU dan regularisasi L2 untuk mengurangi *overfitting*. *BatchNormalization* dan *Dropout* sebesar 0.5 juga diterapkan guna meningkatkan stabilitas pembelajaran dan mencegah *overfitting*.

Lapisan *output* menggunakan fungsi aktivasi *softmax* dengan tiga *neuron* untuk mengklasifikasikan citra ke dalam kelas matang, mentah, dan busuk. Model dikompilasi menggunakan optimizer Adam dan fungsi *loss categorical\_crossentropy* dengan evaluasi berdasarkan akurasi. Selain itu, *callback ReduceLROnPlateau* digunakan untuk menurunkan *learning rate* secara otomatis ketika performa validasi tidak mengalami peningkatan agar proses optimasi menjadi lebih stabil.

## 2.5. ResNet50

ResNet50 dipilih sebagai salah satu model pembandingan karena dikenal memiliki performa yang baik dalam tugas klasifikasi citra serta mampu mengatasi masalah *vanishing gradient* pada jaringan yang dalam [12]. Arsitektur ini menggunakan konsep *skip connection*, yaitu mekanisme yang memungkinkan informasi diteruskan langsung ke lapisan yang lebih dalam tanpa melewati seluruh proses transformasi, sehingga proses pelatihan menjadi lebih stabil dan akurat. Struktur dan konfigurasi lapisan ResNet50 setelah modifikasi ditunjukkan pada Tabel 3.

Tabel 3. Struktur Model ResNet50

| Lapisan         | Konfigurasi Lapisan                                   |
|-----------------|-------------------------------------------------------|
| Input layer     | 224x224x3                                             |
| ResNet50        | Pretrained ImageNet, tanpa top layer, pooling = 'avg' |
| Fully connected | 128 neuron, ReLU, L2 Regularization (0.0001)          |
| Fully connected | 64 neuron, ReLU, L2 Regularization (0.0001)           |
| Dropout         | 0.5                                                   |
| Output layer    | Softmax 3 kelas                                       |

Model ResNet50 menggunakan ukuran input 224x224x3 digunakan karena merupakan ukuran input standar pada arsitektur ResNet50 yang telah

dilatih menggunakan dataset ImageNet. Penggunaan ukuran ini memungkinkan pretrained weights dari ImageNet dimanfaatkan secara optimal selama proses *transfer learning* serta menjaga kompatibilitas struktur jaringan dengan model asli. Lalu untuk model ini juga menggunakan pretrained weights dari ImageNet dengan menghapus *fully connected layer* bawaan dan menerapkan *global average pooling*. Beberapa lapisan akhir kemudian di-fine-tune agar model dapat beradaptasi dengan dataset penelitian tanpa kehilangan representasi fitur umum hasil pelatihan awal. Setelah *backbone* ResNet50, ditambahkan dua *fully connected layer* masing-masing 128 dan 64 neuron dengan fungsi aktivasi ReLU dan regularisasi L2. Selain itu, *BatchNormalization* dan *Dropout* sebesar 0.5 digunakan untuk meningkatkan stabilitas pelatihan dan mengurangi *overfitting*. Pada bagian akhir, lapisan *output* menggunakan tiga *neuron* dengan fungsi aktivasi *softmax* untuk mengklasifikasikan citra ke dalam kelas matang, mentah, dan busuk.

## 2.6. DenseNet121

DenseNet121 digunakan sebagai model pembandingan karena memiliki kemampuan meningkatkan aliran informasi dan gradien antar lapisan melalui *dense connections*. Pada arsitektur ini, setiap lapisan menerima input dari seluruh lapisan sebelumnya sehingga memungkinkan *feature reuse* dan efisiensi parameter yang lebih baik dibandingkan CNN konvensional [15]. Struktur dan konfigurasi lapisan DenseNet121 setelah modifikasi ditunjukkan pada Tabel 4.

Tabel 4. Struktur Model DenseNet121

| Lapisan         | Konfigurasi Lapisan                                   |
|-----------------|-------------------------------------------------------|
| Input layer     | 224x224x3                                             |
| DenseNet121     | Pretrained ImageNet, tanpa top layer, pooling = 'avg' |
| Fully connected | 128 neuron, ReLU, L2 Regularization (0.0001)          |
| Fully connected | 64 neuron, ReLU, L2 Regularization (0.0001)           |
| Dropout         | 0.5                                                   |
| Output layer    | Softmax 3 kelas                                       |

Model DenseNet121 menggunakan ukuran input 224x224x3 digunakan untuk menyesuaikan spesifikasi standar DenseNet121 yang telah dilatih sebelumnya pada dataset ImageNet. Penggunaan resolusi ini membantu model mempertahankan performa ekstraksi fitur yang optimal sekaligus meningkatkan efisiensi komputasi selama proses pelatihan dan evaluasi. Untuk model ini juga menggunakan *pretrained weights* dari ImageNet

dengan menghilangkan *top layer* dan menerapkan *global average pooling* untuk agregasi fitur. Beberapa lapisan akhir kemudian di-*fine-tune* agar model lebih mampu mengenali karakteristik citra buah kelapa sawit. Setelah *backbone*, ditambahkan dua *fully connected* layer masing-masing 128 dan 64 *neuron* dengan aktivasi ReLU dan regularisasi L2, serta *BatchNormalization* dan *Dropout* sebesar 0.5 untuk mengurangi *overfitting*. Pada bagian output, digunakan tiga *neuron* dengan fungsi aktivasi *softmax* untuk mengklasifikasikan citra ke dalam kelas matang, mentah, dan busuk.

## 2.7. Pelatihan Model

Proses pelatihan pada model baseline CNN dan model - model transfer learning dilakukan menggunakan pendekatan yang sama agar hasil evaluasi dapat dibandingkan secara adil. Kedua model menggunakan fungsi loss categorical crossentropy yang umum digunakan pada klasifikasi multi-kelas [16]. Fungsi ini mengukur perbedaan antara probabilitas prediksi model dan label sebenarnya sehingga model dapat memperbaiki performanya selama proses pelatihan. Evaluasi model dilakukan menggunakan metrik accuracy untuk mengetahui persentase prediksi yang benar.

Selama pelatihan, beberapa *callback* diterapkan untuk meningkatkan performa model dan mencegah *overfitting*. *TensorBoard* digunakan untuk memantau proses pelatihan secara *real-time*, seperti perubahan *loss* dan *accuracy* pada setiap epoch. Selain itu, *callback EarlyStopping* diterapkan untuk menghentikan pelatihan apabila *validation loss* tidak mengalami peningkatan selama 5 epoch berturut-turut. Pendekatan ini membantu mengurangi *overfitting* sekaligus menghemat waktu dan sumber daya komputasi. Seluruh proses pelatihan dilakukan dengan maksimum 25 epoch.

## 2.8. Evaluasi Model

Setelah proses pelatihan selesai, tahap berikutnya adalah evaluasi model untuk mengukur kemampuan model dalam mengklasifikasikan data baru yang belum pernah digunakan selama pelatihan. Evaluasi dilakukan menggunakan data uji yang telah dipisahkan sebelumnya dari data training dan validation. Pada tahap ini, model menerima input berupa gambar buah kelapa sawit dengan tiga kelas, yaitu matang, mentah, dan busuk, kemudian menghasilkan prediksi berdasarkan fitur yang telah dipelajari selama proses pelatihan. Hasil prediksi tersebut

selanjutnya dibandingkan dengan label asli pada data uji untuk mengetahui tingkat kinerja model dalam melakukan klasifikasi.

Beberapa metrik evaluasi seperti *accuracy*, *precision*, *recall* dan *F1-score* akan digunakan untuk mengukur kinerja model. *Accuracy* memberikan gambaran tentang seberapa baik model mengklasifikasikan sampel secara keseluruhan, sementara *precision*, *recall*, dan *F1-score* memberikan wawasan tentang kinerja model dalam mengklasifikasikan sampel dari kelas tertentu [17]. Perhitungan *accuracy*, *precision*, *recall*, dan *F1-score* dapat dilihat pada Persamaan (1)–(4).

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

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

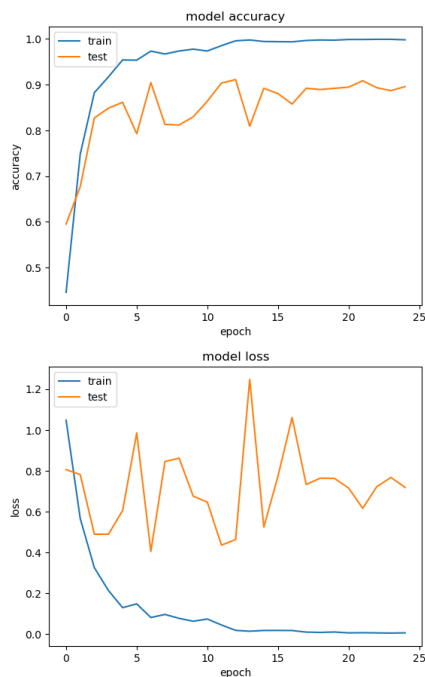
$$Recall = \frac{TP}{TP+FN} \quad (3)$$

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

## 3. Hasil dan Pembahasan

### 3.1. Baseline CNN

Model *baseline* CNN dilatih menggunakan arsitektur CNN konvensional yang terdiri dari tiga convolution layer dan *max-pooling* layer [18]. CNN. Proses pelatihan dilakukan menggunakan *optimizer* Adam dan fungsi loss categorical crossentropy dengan maksimum 25 epoch. Berdasarkan hasil pelatihan, model baseline CNN menunjukkan peningkatan performa secara bertahap pada setiap epoch. Grafik *training accuracy* dan *validation accuracy* menunjukkan bahwa model mampu mempelajari pola visual dari citra buah kelapa sawit dengan cukup baik. Selain itu, nilai *training loss* dan *validation loss* mengalami penurunan selama proses pelatihan berlangsung. Hasil dari akurasi dan *loss* selama pelatihan dan validasi model Baseline CNN dapat dilihat pada gambar 5 dibawah ini.



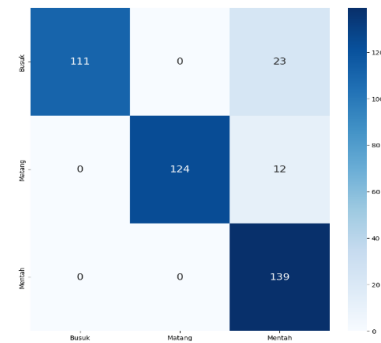
Gambar 5. Hasil Akurasi dan *Loss* Pelatihan dan Validasi Baseline CNN

Berdasarkan grafik pelatihan, baseline CNN menunjukkan proses pembelajaran yang cukup stabil meskipun masih terdapat selisih antara *training accuracy* dan *validation accuracy* yang mengindikasikan adanya *overfitting* ringan. Hal ini disebabkan karena model dibangun dari awal menggunakan dataset yang relatif terbatas dibandingkan model *transfer learning* yang memanfaatkan *pretrained weights*. Hasil evaluasi model baseline CNN pada data *testing* dapat dilihat pada Tabel 3.

Tabel 5. Hasil Keseluruhan Baseline CNN

| Metrik Evaluasi  | Nilai  |
|------------------|--------|
| <i>Accuracy</i>  | 91.44% |
| <i>Recall</i>    | 91.4%  |
| <i>Precision</i> | 93.16% |
| <i>F1-Score</i>  | 91.59% |

Selain menggunakan metrik evaluasi, analisis performa model juga dilakukan menggunakan *confusion matrix* yang ditunjukkan pada Gambar 6.



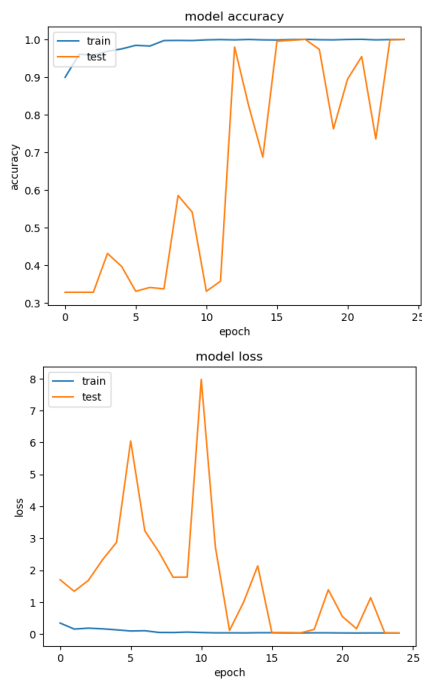
Gambar 6. Hasil *Confusion Matrix* Baseline CNN

Berdasarkan *confusion matrix*, sebagian besar citra berhasil diklasifikasikan dengan benar. Namun demikian, masih terdapat beberapa kesalahan klasifikasi terutama pada kelas yang memiliki karakteristik visual yang mirip, seperti warna buah yang mendekati tingkat kematangan tertentu.

Berdasarkan *confusion matrix*, sebagian besar kesalahan klasifikasi terjadi antara kelas mentah dan matang. Hal ini diduga karena kedua kelas memiliki karakteristik warna yang relatif mirip pada kondisi pencahayaan tertentu. Selain itu, variasi intensitas cahaya saat proses pengambilan gambar juga dapat memengaruhi kemampuan model dalam membedakan karakteristik visual antar kelas. Kesalahan klasifikasi tersebut lebih banyak ditemukan pada model Baseline CNN dibandingkan model *transfer learning*, yang menunjukkan bahwa fitur yang dipelajari oleh model *pretrained* lebih representatif untuk membedakan tingkat kematangan buah kelapa sawit.

### 3.2. EfficientNetB3

EfficientNetB3 digunakan sebagai model *transfer learning* dengan memanfaatkan *pretrained weights* dari ImageNet. Model ini menerapkan metode *compound scaling* yang menyeimbangkan kedalaman, lebar, dan resolusi jaringan secara efisien [19]. Hasil pelatihan menunjukkan bahwa EfficientNetB3 mampu mencapai konvergensi lebih cepat dibandingkan baseline CNN. Grafik *training accuracy* dan *validation accuracy* menunjukkan peningkatan performa yang stabil dengan nilai *validation accuracy* yang tinggi. Selain itu, nilai *loss* mengalami penurunan secara konsisten selama proses pelatihan. Hasil dari akurasi dan *loss* selama pelatihan dan validasi model EfficientNetB3 dapat dilihat pada gambar 7 dibawah ini.

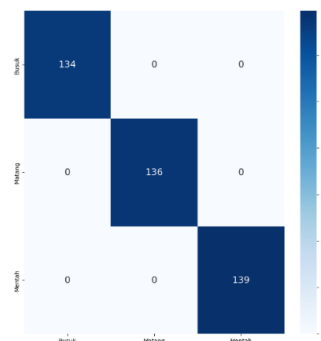


Gambar 7. Hasil Akurasi dan Loss Pelatihan dan Validasi EfficientNetB3

Berdasarkan grafik pelatihan, EfficientNetB3 mengindikasikan kemampuan generalisasi yang baik pada dataset penelitian dengan selisih *training* dan *validation accuracy* yang relatif kecil. Hal ini menunjukkan bahwa model mampu mempelajari pola data tanpa mengalami *overfitting* yang signifikan. Hasil evaluasi EfficientNetB3 dapat dilihat pada Tabel 6.

Tabel 6. Hasil Keseluruhan EfficientNetB3

| Metrik Evaluasi | Nilai |
|-----------------|-------|
| Accuracy        | 100%  |
| Recall          | 100%  |
| Precision       | 100%  |
| F1-Score        | 100%  |

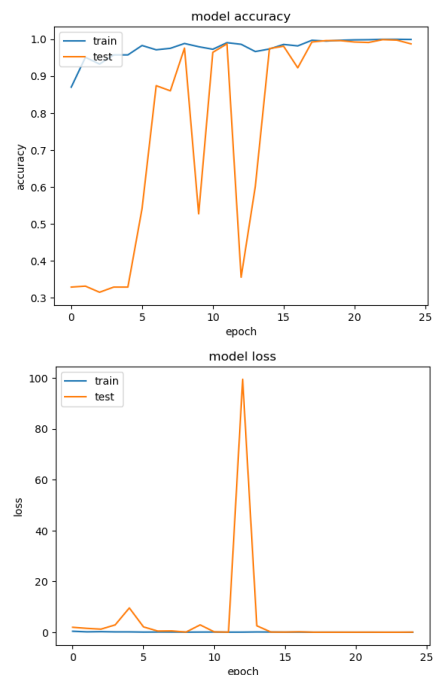


Gambar 8. Hasil Confussion Matrix EfficientNetB3

Berdasarkan *confusion matrix*, EfficientNetB3 mampu mengklasifikasikan sebagian besar citra dengan benar pada ketiga kelas. Tingkat kesalahan klasifikasi relatif rendah dibandingkan *baseline CNN*. Hal ini menunjukkan bahwa EfficientNetB3 memiliki kemampuan ekstraksi fitur yang lebih baik.

### 3.3. ResNet50

ResNet50 digunakan sebagai model pembanding karena memiliki *residual connection* yang mampu membantu proses pelatihan jaringan mendalam menjadi lebih stabil [20]. Hasil pelatihan menunjukkan bahwa ResNet50 mampu menghasilkan performa klasifikasi yang baik dengan peningkatan *accuracy* yang stabil pada setiap epoch. Grafik pelatihan menunjukkan bahwa model mampu mempertahankan *validation accuracy* yang tinggi selama proses pelatihan berlangsung.



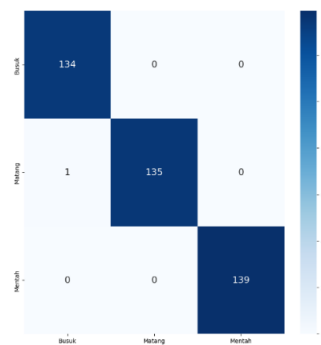
Gambar 9. Hasil Akurasi dan Loss Pelatihan dan Validasi ResNet50

Nilai *training loss* dan *validation loss* mengalami penurunan yang konsisten sehingga menunjukkan bahwa model mampu mempelajari fitur visual secara efektif. Penggunaan *residual connection* membantu model mengatasi permasalahan *vanishing gradient* pada jaringan yang dalam. Hasil evaluasi model ResNet50 dapat dilihat pada Tabel 7.

Tabel 7. Hasil Keseluruhan ResNet50

| Metrik Evaluasi | Nilai  |
|-----------------|--------|
| Accuracy        | 99.75% |

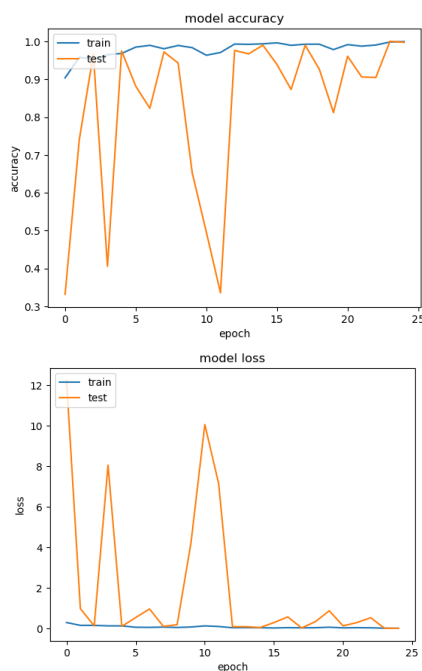
|           |        |
|-----------|--------|
| Recall    | 99.75% |
| Precision | 99.76% |
| F1-Score  | 99.75% |

Gambar 10. Hasil *Confusion Matrix* ResNet50

Berdasarkan *confusion matrix*, ResNet50 mampu melakukan klasifikasi dengan baik pada seluruh kelas. Namun demikian, masih terdapat beberapa kesalahan prediksi pada citra dengan kondisi pencahayaan dan tekstur yang mirip.

### 3.4 DenseNet121

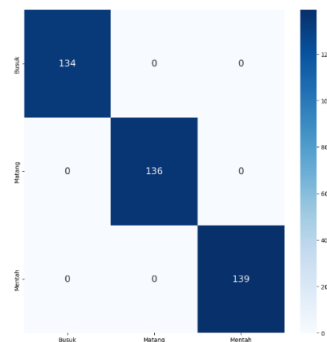
DenseNet121 digunakan sebagai model *transfer learning* karena memiliki mekanisme *dense connectivity* yang memungkinkan setiap lapisan menerima informasi dari seluruh lapisan sebelumnya [21]. Hasil pelatihan menunjukkan bahwa DenseNet121 mampu menghasilkan performa klasifikasi yang sangat baik dengan peningkatan *accuracy* yang stabil selama proses pelatihan. Nilai *training loss* dan *validation loss* juga mengalami penurunan secara konsisten.

Gambar 11. Hasil Akurasi dan *Loss* Pelatihan dan Validasi DenseNet121

DenseNet121 mengindikasikan kemampuan generalisasi yang baik pada dataset penelitian karena selisih antara *training accuracy* dan *validation accuracy* relatif kecil. *Dense connectivity* membantu model mempertahankan aliran informasi dan gradien secara optimal selama proses pelatihan. Hasil evaluasi DenseNet121 dapat dilihat pada Tabel 8.

Tabel 8. Hasil Keseluruhan DenseNet121

| Metrik Evaluasi | Nilai |
|-----------------|-------|
| Accuracy        | 100%  |
| Recall          | 100%  |
| Precision       | 100%  |
| F1-Score        | 100%  |

Gambar 12. Hasil *Confusion Matrix* DenseNet121

Berdasarkan *confusion matrix*, DenseNet121 mampu mengklasifikasikan sebagian besar citra dengan benar pada ketiga kelas. Tingkat kesalahan klasifikasi relatif rendah sehingga menunjukkan kemampuan model dalam mengenali karakteristik visual buah kelapa sawit secara efektif.

### 3.4 Perbandingan Hasil Model

Setelah seluruh model selesai dilatih dan diuji, dilakukan perbandingan performa berdasarkan metrik evaluasi yang digunakan. Hasil perbandingan performa seluruh model dapat dilihat pada Tabel 9.

Tabel 9. Perbandingan Performa Setiap Model

| Model          | Accuracy | Precision | Recall | F1-Score |
|----------------|----------|-----------|--------|----------|
| Baseline CNN   | 91,44%   | 93,16%    | 91,44% | 91,59%   |
| EfficientNetB3 | 100%     | 100%      | 100%   | 100%     |
| ResNet50       | 99,75%   | 99,76%    | 99,75% | 99,75%   |
| DenseNet121    | 100%     | 100%      | 100%   | 100%     |

Selain membandingkan performa klasifikasi, penelitian ini juga menganalisis karakteristik arsitektur dari setiap model yang digunakan. Perbandingan tersebut meliputi jumlah layer serta mekanisme utama yang digunakan oleh masing-masing arsitektur dalam proses ekstraksi fitur. Karakteristik arsitektur model dapat dilihat pada Tabel 10.

Tabel 10. Perbandingan Arsitektur Model

| Model          | Jumlah Layer                           | Arsitektur                                             |
|----------------|----------------------------------------|--------------------------------------------------------|
| Baseline CNN   | ±18 layer                              | Beberapa blok konvolusi berurutan + pooling            |
| EfficientNetB3 | ±30-40 block (dengan MBConv)           | MBConv + compound scaling (depth, width, resolution)   |
| ResNet50       | 50 layer (dibagi dalam residual block) | Convolutional block + identity block (skip connection) |
| DenseNet121    | 121 layer                              | Dense block + transition layer                         |

Selain karakteristik arsitektur, evaluasi juga dilakukan terhadap kompleksitas komputasi masing-masing model. Analisis ini bertujuan untuk mengetahui kebutuhan sumber daya yang diperlukan oleh setiap model melalui perbandingan jumlah parameter, ukuran model, dan waktu pelatihan. Hasil perbandingan kompleksitas komputasi model ditunjukkan pada Tabel 11.

Tabel 11. Perbandingan Kompleksitas Komputasi Model

| Model          | Parameter (Juta) | Ukuran Model (mb) | Waktu Training |
|----------------|------------------|-------------------|----------------|
| Baseline CNN   | 1.86             | 7.4               | 28-30 detik    |
| EfficientNetB3 | 10.9             | 43                | 38-45 detik    |
| ResNet50       | 23.8             | 95                | 58-63 detik    |
| DenseNet121    | 8.1              | 33                | 55-60 detik    |

Berdasarkan Tabel 11, terdapat perbedaan kompleksitas komputasi yang cukup signifikan antara model Baseline CNN dan model transfer learning. Baseline CNN memiliki jumlah parameter dan ukuran model yang paling kecil sehingga membutuhkan waktu pelatihan yang relatif lebih singkat. Sebaliknya, ResNet50 memiliki jumlah parameter terbesar sehingga memerlukan sumber daya komputasi yang lebih tinggi. Sementara itu, DenseNet121 dan EfficientNetB3 mampu menghasilkan performa klasifikasi yang sangat tinggi dengan kompleksitas yang masih relatif efisien dibandingkan ResNet50.

Hasil penelitian menunjukkan bahwa penggunaan *transfer learning* memberikan peningkatan performa dibandingkan *baseline CNN* dalam klasifikasi kematangan buah kelapa sawit. Perbedaan performa antar model dipengaruhi oleh

karakteristik arsitektur masing-masing model dalam melakukan ekstraksi fitur dan proses pembelajaran.

*Baseline CNN* mampu mempelajari pola dasar dari citra buah kelapa sawit melalui kombinasi *convolution layer* dan *max-pooling layer*. Namun karena model dilatih dari awal menggunakan dataset yang relatif terbatas, kemampuan generalisasi model masih lebih rendah dibandingkan model *transfer learning*. Hal ini menyebabkan *baseline CNN* masih mengalami beberapa kesalahan klasifikasi pada citra dengan karakteristik visual yang mirip.

EfficientNetB3 menunjukkan performa yang sangat baik karena mampu menjaga keseimbangan antara kompleksitas model dan efisiensi komputasi melalui metode *compound scaling*. Pendekatan ini memungkinkan model melakukan ekstraksi fitur secara mendalam tanpa meningkatkan jumlah parameter secara berlebihan.

ResNet50 juga menghasilkan performa yang tinggi karena penggunaan *residual connection* membantu mengatasi masalah *vanishing gradient* pada jaringan yang dalam. Dengan adanya *skip connection*, informasi dapat diteruskan secara langsung ke lapisan berikutnya sehingga proses pelatihan menjadi lebih stabil.

DenseNet121 menunjukkan performa yang kompetitif karena mekanisme *dense connectivity* memungkinkan setiap lapisan memperoleh informasi dari seluruh lapisan sebelumnya. Pendekatan ini meningkatkan *feature reuse* dan membantu model mempertahankan aliran *gradien* selama proses pelatihan.

Meskipun beberapa model transfer learning menghasilkan accuracy yang sangat tinggi, hasil tersebut perlu diinterpretasikan secara hati-hati mengingat ukuran dataset yang masih relatif terbatas. Dataset kecil berpotensi menyebabkan model mempelajari pola-pola spesifik pada data pelatihan sehingga meningkatkan risiko *overfitting* dan menurunkan kemampuan generalisasi model pada data baru. Hasil accuracy sebesar 100% yang diperoleh EfficientNetB3 dan DenseNet121 menunjukkan kemampuan model dalam mengenali pola pada dataset penelitian, namun hasil tersebut tidak dapat langsung digeneralisasikan pada populasi data yang lebih luas tanpa pengujian pada dataset yang lebih beragam.

Untuk membantu mengurangi risiko tersebut, penelitian ini menerapkan beberapa strategi regularisasi seperti augmentasi data pada training set, dropout, L2 regularization, dan early stopping selama proses pelatihan. Selain itu, proses pembagian dataset dilakukan sebelum augmentasi untuk menghindari data leakage dan memastikan independensi data testing.

Meskipun model transfer learning menghasilkan performa klasifikasi yang sangat tinggi, penelitian ini masih memiliki beberapa keterbatasan. Salah satunya adalah jumlah dataset yang relatif terbatas sehingga kemampuan generalisasi model belum dapat dievaluasi secara menyeluruh pada kondisi lingkungan yang lebih beragam. Selain itu, penelitian ini belum menerapkan pendekatan evaluasi seperti k-fold cross validation maupun external validation dataset sehingga evaluasi model masih bergantung pada satu skenario pembagian dataset. Oleh karena itu, penelitian selanjutnya disarankan menggunakan dataset yang lebih besar dan lebih beragam serta menerapkan metode evaluasi yang lebih robust untuk memperoleh hasil yang lebih generalizable.

Meskipun model transfer learning menghasilkan performa klasifikasi yang sangat tinggi, hasil penelitian ini masih memiliki beberapa keterbatasan. Salah satunya adalah jumlah dataset yang relatif terbatas sehingga kemampuan generalisasi model belum dapat dievaluasi secara menyeluruh pada kondisi lingkungan yang lebih beragam. Selain itu, penelitian ini belum menerapkan pendekatan evaluasi seperti k-fold cross validation maupun external validation dataset sehingga evaluasi model masih bergantung pada satu skenario pembagian dataset. Oleh karena itu, penelitian selanjutnya disarankan untuk menggunakan dataset yang lebih besar dan lebih beragam serta menerapkan metode evaluasi yang lebih robust untuk memperoleh hasil yang lebih generalizable.

#### 4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa pendekatan transfer learning mampu memberikan performa yang lebih baik dibandingkan baseline CNN dalam klasifikasi kematangan buah kelapa sawit. Model Baseline CNN memperoleh accuracy sebesar 91.44%, sedangkan model transfer learning menunjukkan performa yang lebih tinggi, yaitu EfficientNetB3 dan DenseNet121 dengan accuracy sebesar 100%, serta ResNet50 sebesar 99.75%.

Hasil penelitian menunjukkan bahwa penggunaan pretrained model dari ImageNet mampu meningkatkan kemampuan ekstraksi fitur dan generalisasi model pada dataset citra buah kelapa sawit. EfficientNetB3 menunjukkan performa yang sangat baik dengan efisiensi komputasi melalui metode compound scaling, ResNet50 mampu menghasilkan proses pelatihan yang stabil melalui residual connection, sedangkan DenseNet121 menunjukkan kemampuan feature reuse yang efektif melalui dense connectivity.

Meskipun model transfer learning menunjukkan performa yang sangat tinggi, hasil penelitian ini perlu diinterpretasikan secara hati-hati karena jumlah dataset asli yang digunakan relatif terbatas, yaitu 302 citra. Augmentasi data memang mampu meningkatkan variasi data pelatihan, namun tidak sepenuhnya dapat menggantikan keberagaman data nyata yang diperoleh dari kondisi lapangan yang berbeda. Oleh karena itu, penelitian selanjutnya disarankan menggunakan dataset yang lebih besar, lebih beragam, serta berasal dari berbagai lokasi perkebunan untuk meningkatkan kemampuan generalisasi model.

#### Daftar Rujukan

- [1] A. Syaifuddin, L. N. A. Mualifah, L. Hidayat, and A. M. Abadi, "Detection of palm fruit maturity level in the grading process through image recognition and fuzzy inference system to improve quality and productivity of crude palm oil (CPO)," *J. Phys. Conf. Ser.*, vol. 1581, no. 1, 2020, doi: 10.1088/1742-6596/1581/1/012003.
- [2] A. Septiarini, H. R. Hatta, H. Hamdani, A. Oktavia, A. A. Kasim, and S. Suyanto, "Maturity grading of oil palm fresh fruit bunches based on a machine learning approach," *2020 5th Int. Conf. Informatics Comput. ICIC 2020*, 2020, doi: 10.1109/ICIC50835.2020.9288603.
- [3] A. Septiarini, A. Sunyoto, H. Hamdani, A. A. Kasim, F. Utaminigrum, and H. R. Hatta, "Machine vision for the maturity classification of oil palm fresh fruit bunches based on color and texture features," *Sci. Hortic. (Amsterdam)*, vol. 286, no. April, p. 110245, 2021, doi: 10.1016/j.scienta.2021.110245.
- [4] N. A. Aliteh *et al.*, "Fruit battery method for oil palm fruit ripeness sensor and comparison with computer vision method," *Sensors (Switzerland)*, vol. 20, no. 3, pp. 1–14, 2020, doi: 10.3390/s20030637.
- [5] S. Espinoza, C. Aguilera, L. Rojas, and P. G. Campos, "Analysis of Fruit Images with Deep Learning: A Systematic Literature Review and Future Directions," *IEEE Access*, no. October 2023, pp. 1–1, 2023, doi: 10.1109/access.2023.3345789.
- [6] S. Srivastava, A. V. Divekar, C. Anilkumar, I. Naik, V. Kulkarni, and V. Pattabiraman, "Comparative analysis of deep learning image detection algorithms," *J. Big Data*, vol. 8, no. 1, 2021, doi: 10.1186/s40537-021-00434-w.
- [7] J. Naranjo-Torres, M. Mora, R. Hernández-García, R. J.

- Barrientos, C. Fredes, and A. Valenzuela, "A review of convolutional neural network applied to fruit image processing," *Appl. Sci.*, vol. 10, no. 10, 2020, doi: 10.3390/app10103443. [20]
- [8] J. Gupta, S. Pathak, and G. Kumar, "Deep Learning (CNN) and Transfer Learning: A Review," *J. Phys. Conf. Ser.*, 2022, doi: 10.1088/1742-6596/2273/1/012029. [21]
- [9] M. Hussain, J. J. Bird, and D. R. Faria, "A Study on CNN Transfer Learning for Image Classification," pp. 191–202, 2019, doi: 10.1007/978-3-319-97982-3.
- [10] H. A. Santoso, B. F. Safsalsa, and N. Febrianto, "Comparative analysis of convolutional neural network and DenseNet121 transfer learning in agriculture focusing on crop leaf disease identification," no. May, 2026, doi: 10.1108/ACI-03-2024-0132.
- [11] H. Alhichri, A. S. Alswayed, Y. Bazi, N. Ammour, and N. A. Alajlan, "Classification of Remote Sensing Images Using EfficientNet-B3 CNN Model with Attention," *IEEE Access*, vol. 9, pp. 14078–14094, 2021, doi: 10.1109/ACCESS.2021.3051085.
- [12] Y. Yahya Sugandi, I. Soesanti, and H. A. Nugroho, "A Systematic Literature Review of Convolutional Neural Network Architecture for Lung Disease Detection," *2023 6th Int. Conf. Inf. Commun. Technol. ICOIACT 2023*, pp. 230–235, 2023, doi: 10.1109/ICOIACT59844.2023.10455864.
- [13] T. Eelbode, P. Sinonquel, F. Maes, and R. Bisschops, "Pitfalls in training and validation of deep learning systems," *Best Pract. Res. Clin. Gastroenterol.*, vol. 52–53, no. xxxx, p. 101712, 2021, doi: 10.1016/j.bpg.2020.101712.
- [14] F. Adnan, M. J. Awan, A. Mahmoud, H. Nobanee, A. Yasin, and A. M. Zain, "EfficientNetB3-Adaptive Augmented Deep Learning (AADL) for Multi-Class Plant Disease Classification," *IEEE Access*, vol. 11, no. August, pp. 85426–85440, 2023, doi: 10.1109/ACCESS.2023.3303131.
- [15] I. Dawar, A. Srivastava, M. Singal, N. Dhyani, and S. Rastogi, *A systematic literature review on municipal solid waste management using machine learning and deep learning*, vol. 58, no. 6, 2025.
- [16] N. Gowdra, R. Sinha, S. MacDonell, and W. Yan, "Maximum Categorical Cross Entropy (MCCE): A Noise - Robust Alternative Loss Function To Mitigate Racial Bias in Convolutional Neural Networks (CNN) By Reducing Overfitting," no. 2006, pp. 1–11, 2021.
- [17] M. Grandini, E. Bagli, and G. Visani, "Metrics for Multi-Class Classification: an Overview," pp. 1–17, 2020, [Online]. Available: <http://arxiv.org/abs/2008.05756>.
- [18] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, *A review of convolutional neural networks in computer vision*, vol. 57, no. 4. Springer Netherlands, 2024.
- [19] C. F. G. Dos Santos and J. P. Papa, "Avoiding Overfitting: A Survey on Regularization Methods for Convolutional Neural Networks," *ACM Comput. Surv.*, vol. 54, no. 10 s, pp. 1–27, 2022, doi: 10.1145/3510413.
- T. Wang, Z. Dou, C. Bao, and Z. Shi, "Diffusion Mechanism in Residual Neural Network: Theory and Applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 2, pp. 667–680, 2024, doi: 10.1109/TPAMI.2023.3272341.
- S. Dey, R. Roychoudhury, S. Malakar, and R. Sarkar, "An optimized fuzzy ensemble of convolutional neural networks for detecting tuberculosis from Chest X-ray images," *Appl. Soft Comput.*, vol. 114, p. 108094, 2022, doi: 10.1016/j.asoc.2021.108094.