

# Perbandingan Kinerja MobileNetV2 dan EfficientNetB0 dalam Prediksi Kematangan Buah Jaboticaba

Mahardika Wildan Fitriani <sup>1</sup>, Wiwi Widayani <sup>\*2</sup>, Heri Sismoro <sup>3</sup>, Nur Aini <sup>4</sup>, Dwi Nurani <sup>5</sup>

<sup>1</sup> Universitas Amikom Yogyakarta; mahardikawildan@students.amikom.ac.id

<sup>\*2</sup> Universitas Amikom Yogyakarta; wiwi.w@amikom.ac.id

<sup>3</sup> Universitas Amikom Yogyakarta; herisismoro@amikom.ac.id

<sup>4</sup> Universitas Amikom Yogyakarta; nuraini@amikom.ac.id

<sup>5</sup> Universitas Amikom Yogyakarta; dwinurani@amikom.ac.id

**Abstrak:** Penentuan kematangan buah yang akurat penting untuk meningkatkan kualitas panen dan efisiensi distribusi, namun praktik manual masih rentan terhadap subjektivitas. Penelitian ini membandingkan kinerja Convolutional Neural Network (CNN), MobileNetV2, dan EfficientNetB0 dalam memprediksi jumlah hari menuju kematangan buah Jaboticaba (*Plinia cauliflora*) berbasis citra RGB. Dataset diperoleh dari pengamatan lapangan dan diuji dalam dua skenario: 656 citra dan dataset hasil augmentasi sebanyak  $\pm 2.500$  citra. Seluruh model dilatih dengan optimizer Adam dan dievaluasi menggunakan Mean Absolute Error (MAE). Pada dataset kecil, MobileNetV2 menunjukkan performa terbaik dengan MAE uji 1,54 hari, lebih baik dibanding CNN (3,41 hari) dan EfficientNetB0 (6,05 hari). Setelah jumlah data diperbesar, kinerja seluruh model meningkat, dengan MobileNetV2 tetap unggul (MAE 0,95 hari), diikuti EfficientNetB0 (1,30 hari) dan CNN (1,86 hari). Peningkatan terbesar terjadi pada EfficientNetB0, yang mengindikasikan bahwa model berkapasitas tinggi membutuhkan data lebih besar untuk belajar secara optimal. Hasil ini menegaskan bahwa ukuran dataset berpengaruh signifikan terhadap akurasi, serta bahwa MobileNetV2 memberikan keseimbangan terbaik antara ketepatan dan stabilitas.

**Keywords:** Jaboticaba, deep learning, MobileNetV2, EfficientNetB0, prediksi kematangan.

DOI: <https://doi.org/10.47134/jacis.v6i1.177>

\*Correspondensi: wiwi.widayani

Email: [wiwi.w@amikom.ac.id](mailto:wiwi.w@amikom.ac.id)

Receive: 2 Maret 2026

Accepted: 31 Maret 2026

Published: 29 April 2026



**Copyright:** © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

**Abstrak:** Accurate fruit ripeness estimation is essential for improving harvest quality and distribution efficiency; however, manual assessment remains prone to subjectivity. This study compares the performance of Convolutional Neural Network (CNN), MobileNetV2, and EfficientNetB0 in predicting the number of days to ripeness of Jaboticaba (*Plinia cauliflora*) using RGB images. The dataset was collected from field observations and evaluated under two scenarios: 656 images and an augmented dataset of approximately 2,500 images. All models were trained using the Adam optimizer and evaluated with Mean Absolute Error (MAE). On the small dataset, MobileNetV2 achieved the best performance with a test MAE of 1.54 days, outperforming CNN (3.41 days) and EfficientNetB0 (6.05 days). After increasing the dataset size, the performance of all models improved, with MobileNetV2 remaining superior (MAE 0.95 days), followed by EfficientNetB0 (1.30 days) and CNN (1.86 days). These findings confirm that dataset size significantly affects accuracy and that MobileNetV2 provides the best balance between accuracy and stability.

**Keywords:** Jaboticaba, deep-learning, MobileNetV2, EfficientNetB0, ripeness regression.

## PENDAHULUAN

Perkembangan teknologi kecerdasan buatan (*Artificial Intelligence/AI*), khususnya pada bidang computer vision dan deep learning, telah mendorong transformasi signifikan dalam sektor pertanian modern. Implementasi teknologi ini memungkinkan proses pemantauan kualitas hasil pertanian dilakukan secara otomatis, objektif, dan konsisten dibandingkan metode manual yang masih bergantung pada observasi visual manusia. Beberapa studi sebelumnya menegaskan bahwa deep learning telah menjadi fondasi penting dalam perkembangan kecerdasan buatan modern karena mampu menghasilkan performa tinggi, skalabilitas luas, dan adaptabilitas pada berbagai domain [1], sebagai teknologi kunci dalam transformasi sector pertanian modern melalui otomatisasi analisis data dan peningkatan efisiensi produksi [2], mampu memberikan akurasi, skalabilitas dan generalisasi yang tinggi [3], cepat serta relevan bagi pertanian cerdas (*smart agiculture*) [4]. Salah satu aspek penting dalam manajemen pascapanen adalah penentuan tingkat kematangan buah yang berpengaruh langsung terhadap waktu panen optimal, kualitas distribusi, serta pengurangan kerugian pascapanen. Oleh karena itu, pengembangan metode prediksi kematangan berbasis citra menjadi topik yang semakin relevan dalam kerangka precision agriculture.

Berbagai penelitian telah menunjukkan bahwa model berbasis *Convolutional Neural Network* (CNN) efektif dalam mengekstraksi fitur visual untuk analisis kematangan buah. Secara umum, riset terdahulu dapat dikelompokkan ke dalam tiga arus utama. Pertama, pendekatan klasifikasi kematangan berbasis CNN konvensional maupun deep CNN yang menunjukkan performa tinggi pada berbagai komoditas buah. Misalnya, CNN mampu mencapai akurasi tinggi pada klasifikasi kematangan tomat [12], serta lebih dari 85% pada klasifikasi kematangan buah kelapa sawit menggunakan kombinasi CNN dan ResNet50 [5]. Pendekatan serupa juga diterapkan pada berbagai buah seperti apel, berry, dan stroberi dengan hasil yang menunjukkan peningkatan akurasi mencapai sekitar 90,24% [6], hasil akurasi 93,75% pada data training, 90,33% pada data testing, dan 100% pada 30 sampel data baru, yang menegaskan bahwa augmentasi transformasi dan ColorJitter memberikan dampak positif terhadap generalisasi model dan ketahanan terhadap variasi kondisi visua [7], deep learning melalui transfer learning efektif untuk klasifikasi tingkat kematangan apel, namun performa model sangat dipengaruhi oleh keseimbangan antara akurasi dan efisiensi komputasi [10], Schmidhuber menegaskan bahwa LSTM menjadi terobosan penting untuk data sekuensial karena mampu mengatasi vanishing gradient problem yang lama menjadi hambatan utama pelatihan jaringan dalam [13].

Kedua, sejumlah penelitian mulai mengeksplorasi arsitektur lightweight dan efisien seperti MobileNet dan EfficientNet untuk mengatasi keterbatasan komputasi dan kebutuhan implementasi pada perangkat edge. Studi sebelumnya melaporkan bahwa MobileNetV2 dapat memberikan kinerja kompetitif dengan kompleksitas model yang lebih rendah pada klasifikasi fase pertumbuhan tomat [11], sementara EfficientNet-B0 menunjukkan peningkatan performa setelah proses fine-tuning pada klasifikasi penyakit buah [9]. Selain itu, beberapa penelitian juga menggabungkan model ringan dengan teknik lanjutan seperti GAN dan attention untuk meningkatkan presisi deteksi kematangan [8]. Temuan-temuan ini mengindikasikan bahwa pemilihan arsitektur yang efisien menjadi faktor penting dalam sistem monitoring berbasis citra.

Ketiga, sebagian penelitian mulai mengarah pada pendekatan regresi untuk mengestimasi tingkat kematangan secara kontinu. Sebagai contoh, CNN yang dikombinasikan dengan data spektroskopi NIR telah digunakan untuk mengestimasi tingkat kematangan apel secara regresi [14]. Namun, sebagian besar studi yang ada masih berfokus pada klasifikasi diskret tingkat kematangan, bukan pada prediksi kontinu seperti jumlah hari menuju kematangan. Selain itu, banyak penelitian memanfaatkan data multimodal (misalnya NIR atau sensor tambahan) atau model yang relatif kompleks, sehingga kurang praktis untuk skenario berbasis citra RGB sederhana.

Meskipun kemajuan telah dicapai, terdapat beberapa celah penelitian (*research gap*) yang masih terbuka. Pertama, studi komparatif yang secara khusus mengevaluasi trade-off antara akurasi dan efisiensi pada tugas regresi kematangan berbasis citra RGB masih terbatas. Kedua, objek buah yang diteliti pada penelitian sebelumnya didominasi oleh komoditas umum seperti tomat, apel, jeruk, dan stroberi, sementara kajian pada buah jaboticaba (anggur Brazil) masih sangat minim. Padahal, jaboticaba memiliki karakteristik perubahan warna yang khas dari hijau hingga ungu gelap yang berpotensi menjadi indikator visual kuat untuk pemodelan kematangan. Ketiga, sebagian penelitian berfokus pada implementasi sistem atau peningkatan arsitektur kompleks, namun belum banyak yang melakukan evaluasi komparatif mendalam antara CNN baseline dan model lightweight modern pada skenario regresi hari menuju kematangan berbasis citra RGB.

Berdasarkan kesenjangan tersebut, penelitian ini bertujuan untuk menganalisis perbandingan kinerja CNN, MobileNetV2, dan EfficientNetB0 dalam memprediksi jumlah hari menuju kematangan buah jaboticaba menggunakan citra RGB. Kebaruan (*novelty*) penelitian ini terletak pada: (i) formulasi tugas sebagai regresi kontinu hari menuju matang alih-alih klasifikasi diskret, (ii) evaluasi komparatif yang menekankan keseimbangan antara performa prediksi dan efisiensi arsitektur pada dataset jaboticaba yang relatif terbatas, serta (iii) penggunaan pipeline berbasis citra visual RGB tanpa ketergantungan pada sensor tambahan. Tujuan penelitian ini adalah mengidentifikasi arsitektur yang paling efektif dan efisien untuk prediksi kematangan jaboticaba, sehingga dapat menjadi referensi dalam pengembangan sistem monitoring kematangan buah berbasis AI yang praktis dan adaptif untuk sektor pertanian presisi.

## METODE

Penelitian ini disusun sebagai eksperimen kuantitatif komparatif untuk mengevaluasi tiga arsitektur deep learning berbasis citra—CNN baseline, MobileNetV2, dan EfficientNetB0—dalam memprediksi jumlah hari menuju kematangan buah Jaboticaba dari citra RGB. Secara formal, dataset didefinisikan sebagai himpunan pasangan input–target

$$\mathcal{D} = (I_i, y_i)_{i=1}^N, \quad (1)$$

dengan  $(I_i \in \mathbb{R}^{H \times W \times 3})$  adalah citra RGB dan  $(y_i \in \mathbb{R})$  adalah label regresi yang menyatakan jumlah hari menuju matang. Fungsi prediktor yang dipelajari dinyatakan melalui persamaan (2)

$$\hat{y}_i = f_{\theta}(I_i), \quad (2)$$

dengan  $(\theta)$  adalah parameter model. Seluruh proses penelitian mengikuti alur pada Gambar 1, mulai dari akuisisi data, pra-pengolahan, pemodelan, pelatihan, hingga evaluasi.

Objek penelitian adalah buah Jaboticaba (*Plinia cauliflora*) yang diamati dari fase awal pembungaan sampai matang penuh. Akuisisi citra dilakukan harian di kebun pribadi dengan pencahayaan alami untuk mempertahankan karakteristik lapangan. Contoh objek pohon dan buah disajikan pada Gambar 2. Pada tahap ini, setiap sampel diberi label numerik berbasis selisih hari terhadap tanggal buah mencapai matang; secara konseptual, bila  $(t_i)$  adalah hari pengambilan citra ke-(i) dan  $(t^*)$  adalah hari buah matang, maka label dapat dinyatakan seperti persamaan (3)

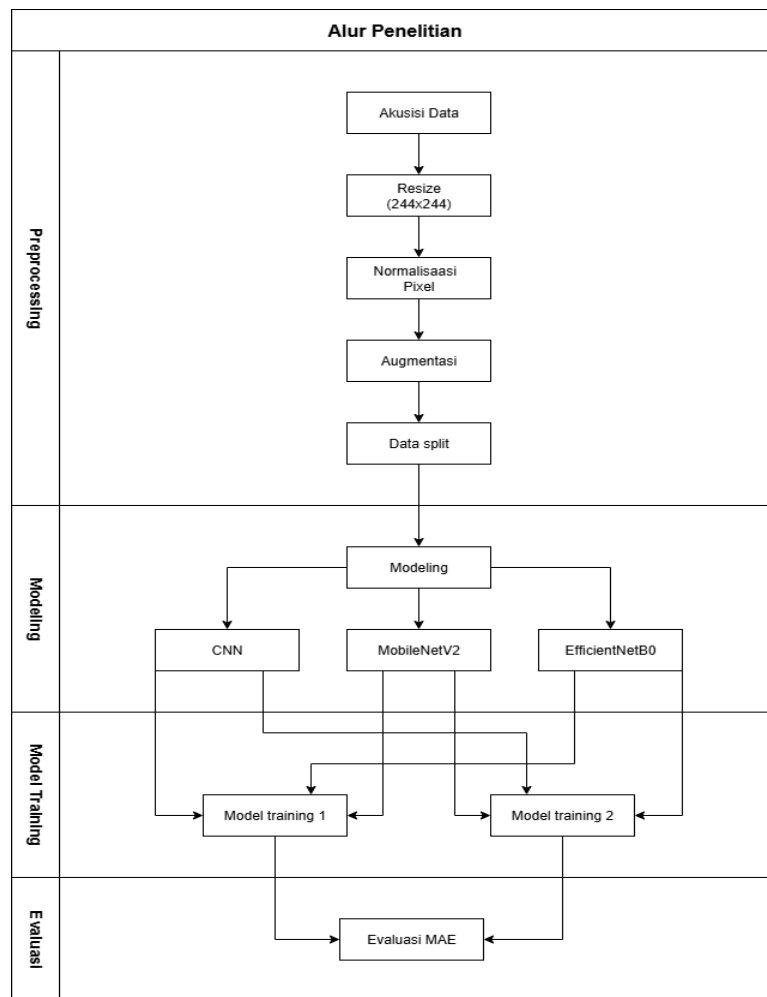
$$y_i = t^* - t_i. \tag{3}$$

Selanjutnya, populasi penelitian mencakup seluruh citra yang diperoleh selama periode observasi, sementara sampel dipilih dengan purposive sampling melalui kurasi kualitas (inklusi-eksklusi) agar input yang digunakan memiliki kejernihan visual dan label yang valid.

Sebelum pelatihan, citra diproses agar seragam dan stabil secara numerik. Pertama, setiap citra diubah ukuran menjadi 224x224 piksel. Secara umum, operasi ini dapat dinyatakan sebagai transformasi pada persamaan (4)

$$I_i^{(r)} = \mathcal{R}(I_i; 224, 224), \tag{4}$$

dengan  $(\mathcal{R}(\cdot))$  adalah operator resizing (misal bilinear interpolation). Kedua, normalisasi piksel dilakukan untuk memetakan intensitas ke rentang  $([0,1])$ ,



Gambar 1. Tahapan penelitian yang dilakukan

$$I_i^{(n)} = \frac{I_i^{(r)}}{255}. \quad (4)$$

Ketiga, untuk mengurangi overfitting pada dataset terbatas, augmentasi diterapkan hanya pada data latih melalui transformasi acak yang mempertahankan makna semantik objek. Secara matematis, satu sampel hasil augmentasi dapat ditulis sebagai

$$\tilde{I}_i = \mathcal{T}(I_i^{(n)}; \omega), \quad (5)$$

dengan  $(\tau(\cdot))$  merepresentasikan komposisi transformasi (*rotasi, translasi, shear, zoom, dan flip*) dan  $(\omega)$  adalah parameter acak transformasi. Dengan demikian, distribusi data latih diperkaya tanpa mengubah target ( $y_i$ ).



**Gambar 2.** Pohon buah Jaboticaba (*Plinia cauliflora*)

Dataset kemudian dibagi menggunakan skema hold-out menjadi tiga subset: latih, validasi, dan uji. Secara formal,

$$\mathcal{D} = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{val}} \cup \mathcal{D}_{\text{test}}, \quad \mathcal{D}_{\text{train}} \cap \mathcal{D}_{\text{val}} = \emptyset, \mathcal{D}_{\text{train}} \cap \mathcal{D}_{\text{test}} = \emptyset, \mathcal{D}_{\text{val}} \cap \mathcal{D}_{\text{test}} = \emptyset, \quad (6)$$

dengan proporsi 80%:10%:10%. Data latih digunakan untuk memperbarui parameter, data validasi untuk pemantauan generalisasi selama pelatihan, dan data uji hanya untuk evaluasi final.

Pada tahap pemodelan, ketiga arsitektur dibandingkan dalam kerangka regresi yang sama, yaitu menghasilkan keluaran skalar ( $\hat{y}$ ). Setiap model terdiri dari backbone ekstraksi fitur dan head regresi. Secara umum, proses inferensi dapat ditulis sebagai persamaan (7)

$$\mathbf{z}_i = g_{\theta_b}(I_i^{(n)}), \quad \hat{y}_i = h_{\theta_h}(\mathbf{z}_i), \quad (7)$$

dengan  $(g_{\theta_b})$  adalah backbone (CNN/MobileNetV2/EfficientNetB0) dan  $(h_{\theta_h})$  adalah head regresi (misalnya Global Average Pooling diikuti *fully connected layer*). Pada CNN baseline, backbone dibangun dari rangkaian operasi konvolusi dan pooling. Untuk sebuah layer konvolusi ke- $(\ell)$ , aktivasi dapat dinyatakan dengan persamaan (8)

$$\mathbf{A}^{(\ell)} = \sigma\left(\mathbf{W}^{(\ell)} * \mathbf{A}^{(\ell-1)} + \mathbf{b}^{(\ell)}\right), \quad (8)$$

dengan (\*) konvolusi, ( $\sigma(\cdot)$ ) adalah aktivasi ReLU, dan ( $\mathbf{A}^{(0)} = I_i^{(n)}$ ). Mekanisme pooling kemudian mereduksi resolusi spasial:

$$\mathbf{P}^{(\ell)} = \text{MaxPool}(\mathbf{A}^{(\ell)}). \quad (9)$$

Versi CNN yang diperkuat menambahkan dropout untuk regularisasi, yang dapat dimodelkan sebagai persamaan (10)

$$\mathbf{u} = \mathbf{m} \odot \mathbf{v}, \quad m_j \sim \text{Bernoulli}(1 - p), \quad (10)$$

dengan ( $p$ ) adalah dropout rate, ( $\odot$ ) adalah perkalian elemen, dan ( $\mathbf{V}$ ) aktivasi layer.

Pada MobileNetV2 dan EfficientNetB0, penelitian memanfaatkan dua skenario pelatihan: (i) feature extractor dengan parameter backbone dibekukan dan hanya head yang dilatih, serta (ii) fine-tuning dengan membuka sebagian layer akhir untuk adaptasi domain. Secara formal, pada skenario freeze:

$$\theta_b \leftarrow \theta_b^{\text{pre}}, \quad \nabla_{\theta_b} \mathcal{L} = 0, \quad (11)$$

Sementara pada fine-tuning:

$$\theta_b = \{\theta_b^{\text{frozen}}, \theta_b^{\text{train}}\}, \quad \nabla_{\theta_b^{\text{train}}} \mathcal{L} \neq 0, \quad (12)$$

sehingga hanya subset parameter tertentu yang diperbarui. Pada EfficientNetB0 “from scratch”, bobot awal tidak berasal dari pretraining, sehingga parameter diinisialisasi acak melalui persamaan (13)

$$\theta_b \sim \text{Init}(\cdot), \quad \text{weights} = \emptyset. \quad (13)$$

Dengan rancangan ini, perbandingan tidak hanya mengukur ketepatan prediksi, tetapi juga menguji dampak transfer learning vs pelatihan penuh pada dataset Jaboticaba.

Pelatihan model dilakukan dengan meminimalkan fungsi kerugian Mean Absolute Error. Untuk *mini batch*  $\mathcal{B} \subset \mathcal{D}_{\text{train}}$ , fungsi objektifnya adalah

$$\mathcal{L}_{\text{MAE}}(\theta) = \frac{1}{|\mathcal{B}|} \sum_{(I_i, y_i) \in \mathcal{B}} |y_i - f_{\theta}(I_i)|. \quad (14)$$

Parameter diperbarui menggunakan Adam. Secara ringkas, untuk gradien  $\mathbf{g}_t = \nabla_{\theta} \mathcal{L}_t$ ,

$$\begin{aligned} \mathbf{m}_t &= \beta_1 \mathbf{m}_{t-1} + (1 - \beta_1) \mathbf{g}_t, & \mathbf{v}_t &= \beta_2 \mathbf{v}_{t-1} + (1 - \beta_2) \mathbf{g}_t^2, \\ \hat{\mathbf{m}}_t &= \frac{\mathbf{m}_t}{1 - \beta_1^t}, & \hat{\mathbf{v}}_t &= \frac{\mathbf{v}_t}{1 - \beta_2^t}, & \theta_{t+1} &= \theta_t - \alpha \frac{\hat{\mathbf{m}}_t}{\sqrt{\hat{\mathbf{v}}_t} + \epsilon}. \end{aligned} \quad (15)$$

Untuk menilai ketahanan model terhadap ukuran data, eksperimen dilakukan pada dua skenario pelatihan (dataset dasar dan dataset diperluas). Selain itu, mekanisme early stopping digunakan pada beberapa konfigurasi untuk menghentikan pelatihan ketika kinerja validasi tidak membaik. Secara konseptual, kriteria ini dapat dinyatakan sebagai penghentian jika

$$\mathcal{L}_{\text{val}}^{(e)} \geq \min_{k \leq e-\rho} \mathcal{L}_{\text{val}}^{(k)} \quad (16)$$

selama ( $\rho$ ) epoch berturut-turut (patience), dan model terbaik dipulihkan berdasarkan nilai minimum validasi.

Evaluasi akhir dilakukan pada data uji menggunakan MAE sebagai metrik utama:

$$\text{MAE}_{\text{test}} = \frac{1}{N_{\text{test}}} \sum_{(I_i, y_i) \in \mathcal{D}_{\text{test}}} |y_i - f_{\theta}(I_i)|. \quad (17)$$

Selain nilai MAE, dinamika generalisasi dianalisis melalui kurva  $\mathcal{L}_{\text{train}}(e)$  dan  $\mathcal{L}_{\text{val}}(e)$  terhadap epoch ( $e$ ) untuk mengidentifikasi indikasi underfitting/overfitting. Pengujian tambahan pada citra baru lapangan dilakukan sebagai verifikasi eksternal terbatas untuk melihat apakah prediksi tetap stabil pada kondisi yang tidak identik dengan data pelatihan.

Seluruh eksperimen diimplementasikan menggunakan Python dengan framework TensorFlow/Keras pada lingkungan Google Colab. Proses pengolahan data dan visualisasi menggunakan pustaka numerik standar, sedangkan penyimpanan dataset dan artefak model dilakukan melalui penyimpanan cloud untuk menjaga reproduisibilitas serta audit trail eksperimen.

## HASIL DAN PEMBAHASAN

Penelitian ini mengevaluasi pengaruh ukuran data dan pilihan arsitektur jaringan saraf konvolusional terhadap ketepatan prediksi jumlah hari menuju kematangan buah Jaboticaba. Eksperimen dilakukan dalam dua kondisi: Percobaan 1 menggunakan dataset dasar sebanyak 656 citra, dan Percobaan 2 menggunakan dataset yang diperluas melalui augmentasi hingga  $\pm 2.500$  citra. Ketiga model—CNN baseline, MobileNetV2, dan EfficientNetB0—dilatih dengan konfigurasi input yang seragam ( $224 \times 224 \times 3$ ) dan target keluaran berupa nilai skalar regresi, lalu dievaluasi menggunakan Mean Absolute Error (MAE) sebagai ukuran kesalahan prediksi rata-rata dalam satuan hari.

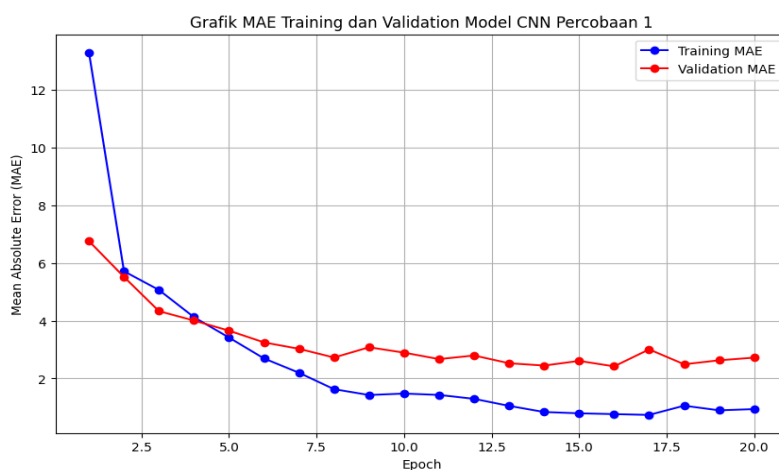
Sebagai pijakan komparasi, Tabel 1 merangkum karakteristik umum ketiga arsitektur. CNN digunakan sebagai baseline yang dilatih dari awal (tanpa pretrained weights), sedangkan MobileNetV2 dan EfficientNetB0 memanfaatkan skema transfer learning dan/atau fine-tuning sesuai rancangan percobaan. Seluruh model dioptimasi dengan Adam dan fungsi kerugian MAE, sehingga perbedaan performa yang muncul terutama merefleksikan perbedaan kapasitas representasi, strategi pelatihan (freeze vs fine-tuning vs from-scratch), serta kecukupan ukuran data terhadap kompleksitas arsitektur.

**Tabel 1.** Karakteristik umum ketiga arsitektur

Komponen	CNN	MobileNetV2	EfficientNetB0
Tipe Model	Dibangun dari awal	Pre-trained ImageNet, fine-tuning	Pre-trained ImageNet, fine-tuning
Jumlah Layer Utama	2 Conv + 2 Pooling + Dense	MobileNetV2 backbone + GAP + Dense	EfficientNetB0 backbone + GAP + Dropout + Dense
Input Shape	$224 \times 224 \times 3$	$224 \times 224 \times 3$	$224 \times 224 \times 3$
Output	1 (regresi jumlah hari menuju matang)	1 (regresi jumlah hari menuju matang)	1 (regresi jumlah hari menuju matang)
Optimizer	Adam	Adam	Adam
Loss	MAE	MAE	MAE
Epoch	20	20	30

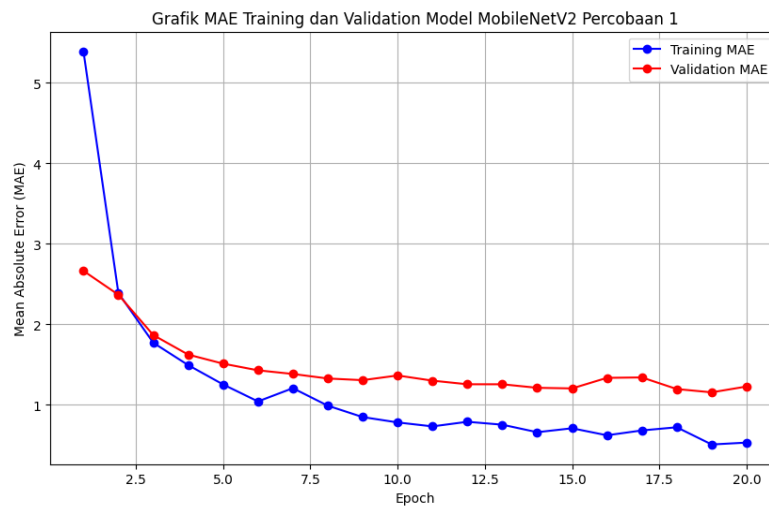
### Percobaan 1 (656 citra)

Percobaan 1 bertujuan menguji kemampuan model mempelajari pola visual kematangan pada kondisi data terbatas. Kondisi ini relevan untuk studi lapangan awal, di mana jumlah citra yang terkurasi umumnya belum besar. CNN baseline menunjukkan MAE pelatihan yang rendah (0,94 hari), tetapi MAE validasi dan uji lebih tinggi (2,72 dan 3,41 hari). Gap yang cukup besar antara pelatihan dan validasi mengindikasikan bahwa model belajar sangat baik pada data latih, namun kemampuan generalisasi pada data yang tidak terlihat masih terbatas. Pola kurva pada Gambar 3 memperlihatkan penurunan MAE yang cepat pada epoch awal lalu melambat hingga konvergen, sementara kurva validasi berfluktuasi ringan dengan jarak yang tetap terhadap kurva pelatihan. Secara interpretatif, hal ini konsisten dengan keterbatasan variasi data dan kapasitas ekstraksi fitur CNN yang relatif sederhana (dua blok konvolusi), sehingga model cenderung menangkap pola dominan pada data latih tetapi kurang robust terhadap variasi pencahayaan, sudut, atau tekstur pada citra baru.



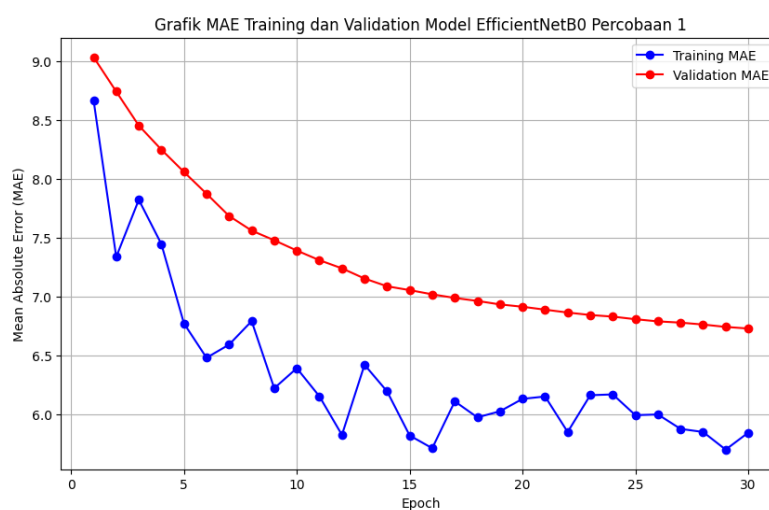
**Gambar 3.** Grafik MAE Training dan Validation Model CNN Percobaan 1

MobileNetV2 (*feature extractor/frozen backbone*) memberikan performa terbaik pada kondisi ini, dengan MAE uji 1,54 hari sejalan dengan penelitian terdahulu yang menegaskan bahwa MobileNetV2 sebagai feature extractor / frozen backbone memberikan performa sangat baik. Menunjukkan state-of-the-art performance untuk mobile/embedded vision tasks pada klasifikasi, deteksi objek, dan segmentasi dengan parameter jauh lebih kecil dibanding model besar seperti VGG atau ResNet, sehingga Sangat cocok digunakan sebagai pretrained frozen feature extractor pada dataset terbatas [15]. Selisih MAE pelatihan dan validasi juga relatif kecil (0,53 vs 1,23 hari), serta kurva pelatihan–validasi pada Gambar 4 tampak stabil dan konvergen tanpa divergensi tajam. Hasil ini menunjukkan bahwa representasi fitur umum dari pretrained ImageNet membantu model menangkap struktur visual dasar (tepi, tekstur, kontur) sehingga pembelajaran pada dataset kecil menjadi lebih efektif. Pembekuan backbone juga membatasi derajat kebebasan parameter yang diperbarui, sehingga menurunkan risiko overfitting pada data yang jumlahnya terbatas.



**Gambar 4.** Grafik MAE Training dan Validation Model MobileNetV2 Percobaan 1

Sebaliknya, EfficientNetB0 (*pretrained, frozen*) menunjukkan MAE uji yang paling tinggi, yaitu 6,05 hari. Kurva pada Gambar 5 memperlihatkan penurunan MAE yang lambat dan bertahan pada tingkat kesalahan yang relatif besar. Dalam konteks dataset kecil, hasil ini mengindikasikan bahwa konfigurasi EfficientNetB0 yang digunakan belum mampu menghasilkan representasi yang cukup sesuai untuk memetakan citra Jaboticaba ke target hari menuju matang. Dua faktor yang paling mungkin adalah: (i) kapasitas dan karakteristik blok EfficientNetB0 membutuhkan variasi data yang lebih kaya agar head regresi dapat memanfaatkan fitur secara optimal, dan/atau (ii) domain gap—fitur pretrained ImageNet tidak serta-merta selaras dengan gradasi warna dan tekstur spesifik Jaboticaba pada kondisi pengambilan data lapangan. Kegagalan model akan muncul untuk sampel uji dengan tingkat kematangan menengah dengan bayangan tidak merata model akan memprediksi kematangan terlalu muda sebaliknya untuk citra dengan paparan cahaya berlebih model akan memprediksi lebih tua 4-5 hari, selain itu Error juga muncul pada sudut pengambilan tertentu ketika sebagian permukaan buah tertutup daun atau objek lain (*occlusion*), sehingga tekstur dan distribusi warna tidak terbaca utuh, dalam kasus ini prediksi dapat meleset hingga lebih dari 5 hari.



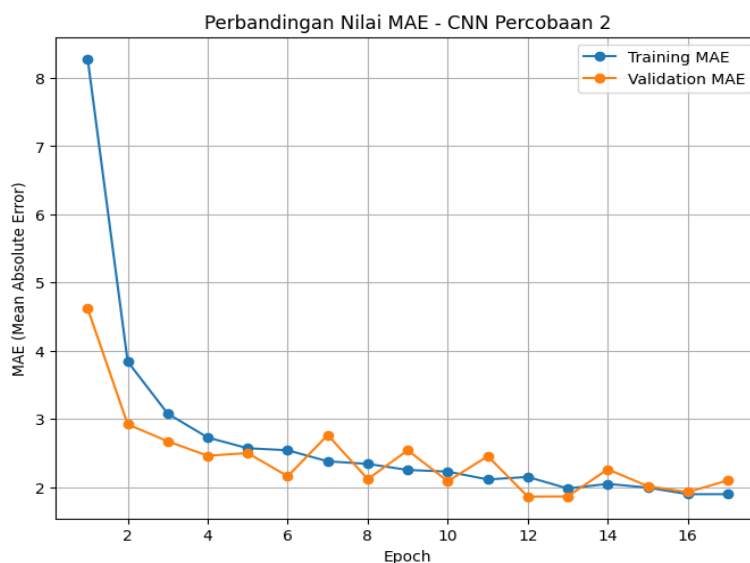
**Gambar 4.** Grafik MAE Training dan Validation Model EfficientNetB0 Percobaan 1

Komparasi Percobaan 1 menegaskan bahwa MobileNetV2 paling efektif pada dataset kecil, diikuti CNN, sedangkan EfficientNetB0 menjadi yang terlemah. Temuan ini menekankan bahwa pada skenario data terbatas, arsitektur ringan dan stabil dengan transfer learning cenderung lebih unggul dibanding model yang lebih kompleks dengan konfigurasi yang belum dioptimalkan untuk domain spesifik.

### Percobaan 2 ( $\pm 2.500$ citra)

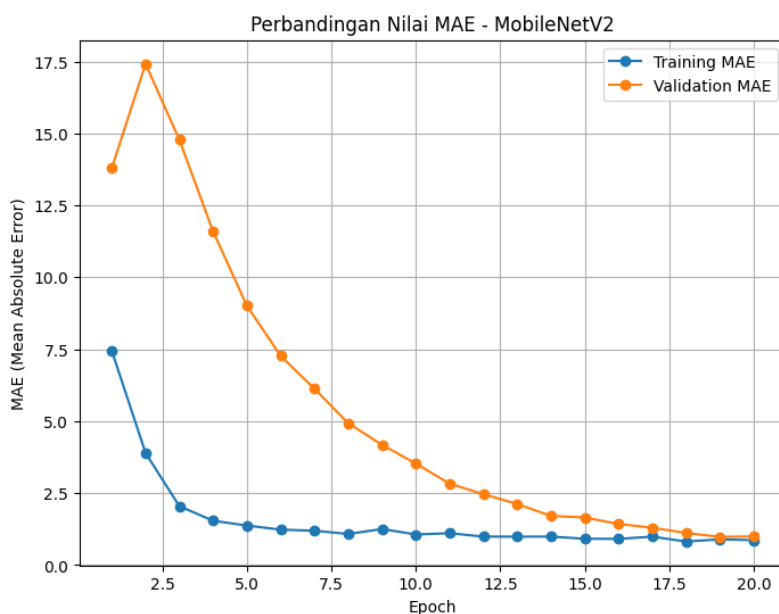
Percobaan 2 dilakukan untuk menguji dampak peningkatan jumlah data terhadap stabilitas pelatihan dan penurunan MAE pada setiap model. Dengan dataset yang diperluas melalui augmentasi, diharapkan distribusi visual menjadi lebih beragam sehingga model dapat mempelajari pola yang lebih general.

Pada kondisi data yang lebih besar, CNN mengalami peningkatan yang paling jelas dalam aspek generalisasi. MAE uji turun dari 3,41 hari menjadi 1,86 hari, dan gap train-val hampir hilang (1,83 vs 1,86 hari) sebagaimana tercermin pada Gambar 5 yang menunjukkan bahwa penambahan jumlah serta keragaman data berhasil mengurangi overfitting dan membantu model mempelajari representasi visual yang lebih stabil terhadap sampel baru namun peningkatan performa rata-rata tidak sepenuhnya menghilangkan error pada kondisi visual kompleks. Penggunaan dropout dan early stopping memperkuat stabilitas ini, dengan dicirikan kurva pelatihan dan validasi menurun bersama dan bergerak berdekatan hingga penghentian otomatis. Secara keseluruhan, hasil ini menunjukkan bahwa CNN baseline dapat menjadi kompetitif ketika diberikan variasi data yang memadai dan regularisasi yang tepat.



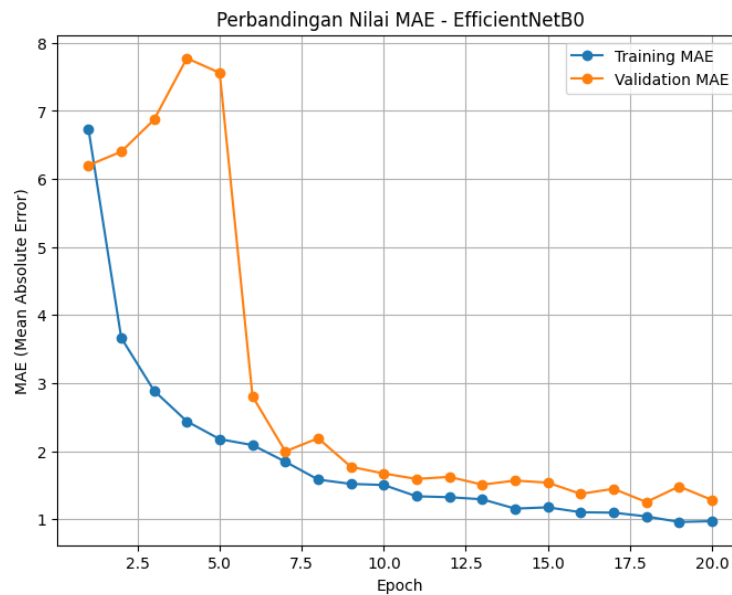
**Gambar 5.** Grafik MAE training dan validation model CNN percobaan 2

MobileNetV2 (dua tahap: frozen + fine-tuning 30 layer terakhir) mempertahankan posisi terbaik dengan MAE uji 0,95 hari (Gambar 6). Gap pelatihan-validation tetap kecil (0,86 vs 0,99 hari), dan kurvanya pada Gambar 6 konvergen secara stabil. Kinerja ini mengindikasikan bahwa fine-tuning pada bagian akhir backbone berhasil mengadaptasi fitur pretrained ke karakteristik spesifik Jaboticaba tanpa mengorbankan generalisasi. Secara praktis, hasil ini memperkuat argumen bahwa MobileNetV2 merupakan pilihan yang sangat efektif ketika dataset sudah cukup untuk mendukung penyesuaian bobot, tetapi tetap membutuhkan efisiensi komputasi.



**Gambar 6.** Grafik MAE training dan validation model *MobileNetV2* percobaan 2

EfficientNetB0 menunjukkan perbaikan drastis dibanding Percobaan 1. MAE uji turun dari 6,05 hari menjadi 1,30 hari (Gambar 7), dan kurva pada Gambar 7 memperlihatkan penurunan MAE yang stabil dengan gap pelatihan–validasi yang moderat. Temuan ini konsisten dengan hipotesis bahwa arsitektur yang lebih kompleks membutuhkan variasi data yang lebih kaya agar dapat memanfaatkan kapasitas representasi secara efektif. Hal ini sejalan dengan tinjauan oleh Naghipour et al. [3] dan Xiao et al. [4] yang menekankan bahwa performa model deep learning dalam tugas deteksi dan klasifikasi buah sangat dipengaruhi oleh kualitas serta keberagaman data, khususnya ketika menggunakan arsitektur yang lebih dalam dan kompleks. Menariknya, pada konfigurasi Percobaan 2 EfficientNetB0 dilatih dari nol (tanpa pretrained), dan hasil yang lebih baik mengindikasikan bahwa pembelajaran fitur spesifik domain dapat mengungguli transfer learning ketika domain gap relatif besar dan data sudah cukup tersedia, hal ini juga dilaporkan oleh Amin et al. [5], yang menunjukkan bahwa model CNN yang dilatih dengan penyesuaian terhadap domain tertentu dapat menghasilkan akurasi yang lebih tinggi dibandingkan pendekatan transfer learning generik pada klasifikasi kesegaran buah. Namun demikian, MAE EfficientNetB0 masih berada di atas MobileNetV2, sehingga secara empiris MobileNetV2 tetap menawarkan kombinasi akurasi dan stabilitas terbaik pada skenario ini.



**Gambar 7.** Grafik MAE training dan validation model *EfficientNetB0* percobaan 2

Komparasi Percobaan 2 menyimpulkan urutan performa MobileNetV2 (0,95 hari) > EfficientNetB0 (1,30 hari) > CNN (1,86 hari). Secara metodologis, hasil ini menunjukkan bahwa peningkatan ukuran data memperkecil kesalahan prediksi di semua arsitektur [16], tetapi derajat peningkatannya berbeda: CNN paling terbantu dalam mengurangi gap generalisasi, EfficientNetB0 paling terbantu dalam menurunkan error absolut dari kondisi awal yang buruk, sementara MobileNetV2 paling konsisten unggul pada kedua kondisi data [17].

### Perbandingan lintas percobaan dan implikasi

Tabel 2 merangkum seluruh MAE untuk kedua percobaan. Secara umum, peningkatan data dari 656 menjadi  $\pm 2.500$  citra menghasilkan penurunan MAE uji pada semua model, yang memperkuat temuan bahwa ukuran dan keragaman data merupakan faktor dominan dalam tugas regresi berbasis citra sebagaimana ditekankan pada studi sebelumnya [3],[4]. Perubahan terbesar terjadi pada EfficientNetB0 (6,05  $\rightarrow$  1,30 hari) dan CNN (3,41  $\rightarrow$  1,86 hari), sedangkan MobileNetV2 tetap unggul dan semakin baik (1,54  $\rightarrow$  0,95 hari). Besarnya penurunan error pada EfficientNetB0 dipengaruhi oleh kapasitas model yang tinggi dan kompleksitas arsitektur yang membuatnya sangat sensitif terhadap keterbatasan data; pada kondisi data kecil model cenderung belum mampu menangkap pola secara optimal (underfitting atau representasi fitur tidak matang), namun ketika jumlah dan variasi data meningkat, model ini mampu memanfaatkan kapasitasnya untuk mempelajari fitur yang lebih diskriminatif [5], [8]. Sebaliknya, error yang relatif lebih kecil dan konsisten pada MobileNetV2 dipengaruhi oleh desain arsitektur yang ringan dan efisien, dengan jumlah parameter yang lebih sedikit sehingga lebih stabil terhadap variasi dataset dan tidak mudah overfitting, sebagaimana juga diamati pada aplikasi CNN ringan untuk deteksi dan klasifikasi buah [6]. Sementara itu, CNN konvensional berada di antara keduanya karena masih bergantung pada kualitas fitur yang dipelajari dari data dan sangat terbantu oleh augmentasi dalam meningkatkan generalisasi [7]. Kondisi dataset besar membuat kurva pelatihan–validasi menjadi lebih rapat (lebih stabil) dan mempercepat konvergensi, terutama pada model yang sebelumnya menunjukkan gap besar.

**Tabel 2.** Ringkasan hasil nilai Mean Absolute Error (MAE) dari seluruh model pada kedua percobaan

Model	MAE Training (P1)	MAE Validation (P1)	MAE Evaluasi (P1)	MAE Training (P2)	MAE Validation (P2)	MAE Evaluasi (P2)
CNN	0,94	2,72	3,41	1,83	1,86	1,86
MobileNetV2	0,53	1,23	1,54	0,86	0,99	0,95
EfficientNetB0	5,85	6,73	6,05	0,97	1,28	1,30

Dari perspektif pemilihan model untuk implementasi, hasil ini memberikan dua implikasi. Pertama, jika ketersediaan data masih terbatas, MobileNetV2 dengan transfer learning (frozen backbone) adalah opsi yang paling aman dan stabil. Kedua, ketika dataset telah cukup besar, fine-tuning MobileNetV2 tetap memberikan performa terbaik, sementara EfficientNetB0 mulai kompetitif, meskipun memerlukan pengaturan pelatihan yang tepat agar kapasitasnya termanfaatkan optimal.

## SIMPULAN

Penelitian ini membandingkan kinerja tiga arsitektur deep learning CNN baseline, MobileNetV2, dan EfficientNetB0 untuk memprediksi jumlah hari menuju kematangan buah Jaboticaba berbasis citra RGB menggunakan metrik Mean Absolute Error (MAE) yang berkontribusi pada analisis komparatif yang secara sistematis mempengaruhi skala data terhadap kinerja model dalam memprediksi kematangan buah dalam pendekatan regresi berbasis citra. Hasil penelitian menunjukkan bahwa ukuran dan keragaman data memiliki peran krusial dalam meningkatkan akurasi dan stabilitas model. Pada kondisi data terbatas, pendekatan transfer learning dengan arsitektur ringan terbukti lebih efektif dalam menjaga generalisasi. Sementara itu, pada kondisi data yang lebih besar, seluruh model mengalami peningkatan performa, dengan model berkapasitas tinggi menunjukkan peningkatan yang paling signifikan, menegaskan pentingnya kecukupan data untuk memanfaatkan potensi representasi fitur yang lebih kompleks.

Pada dataset kecil, MobileNetV2 menunjukkan performa terbaik dengan MAE uji 1,54 hari, lebih unggul dibandingkan CNN (3,41 hari) dan EfficientNetB0 (6,05 hari). Hasil ini menunjukkan bahwa transfer learning dengan backbone dibekukan lebih efektif pada data terbatas karena mampu menekan overfitting. Setelah augmentasi data, seluruh model mengalami peningkatan performa. MobileNetV2 tetap menjadi yang terbaik (MAE 0,95 hari), diikuti EfficientNetB0 (1,30 hari) dan CNN (1,86 hari). Peningkatan terbesar terjadi pada EfficientNetB0, yang mengindikasikan bahwa model berkapasitas tinggi membutuhkan data lebih besar untuk belajar secara optimal.

Secara keseluruhan, jumlah dan keragaman data terbukti berpengaruh langsung terhadap akurasi kematangan berbasis citra. MobileNetV2 memberikan keseimbangan terbaik antara akurasi dan stabilitas generalisasi pada kedua skenario, sehingga direkomendasikan untuk sistem monitoring kematangan Jaboticaba berbasis AI. Penelitian selanjutnya disarankan memperkaya variasi data lapangan, menambahkan metrik regresi seperti RMSE dan koefisien determinasi, serta melakukan validasi eksternal yang lebih luas.

## DAFTAR PUSTAKA

- [1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
- [2] U. Nawaz et al., "AI in Agriculture: A Survey of Deep Learning Techniques for Crops, Fisheries and Livestock," *arXiv:2507.22101*, 2025.
- [3] M. Naghipour, L. Sook Ling, and T. Connie, "A Review of AI Techniques in Fruit Detection and Classification: Analyzing Data, Features and AI Models Used in Agricultural Industry," *IJTech*, vol. 15, no. 3, 2024.
- [4] F. Xiao, X. Zhang, and Y. Li, "Fruit Detection and Recognition Based on Deep Learning for Automatic Harvesting: An Overview and Review," *Agronomy*, vol. 13, no. 6, 2023, doi: 10.3390/agronomy13061579.
- [5] U. Amin et al., "Automatic Fruits Freshness Classification Using CNN and Transfer Learning," *Applied Sciences*, vol. 13, no. 14, 2023, doi: 10.3390/app13148254.
- [6] H. Mundhada et al., "Fruit Detection and Three-Stage Maturity Grading Using CNN," *International Journal of Next-Generation Computing*, vol. 14, no. 1, 2023.
- [7] V. Lusen, I. Purnamasari, and B. N. Sari, "Penerapan CNN untuk Klasifikasi Tingkat Kematangan Berry dengan Augmentasi Transformasi dan ColorJitter Menggunakan TensorFlow," *Jurnal Lentera*, vol. 23, no. 3, pp. 418–427, 2024.
- [8] B. Benmouna et al., "Convolutional Neural Networks for Estimating the Ripening State of Fuji Apples Using Visible and Near-Infrared Spectroscopy," *Food and Bioprocess Technology*, vol. 15, pp. 2226–2236, 2022, doi: 10.1007/s11947-022-02891-5.
- [9] A. R. Rachmad Dianto et al., "Analisis Efektivitas Algoritma MobileNetV3-Large dan EfficientNet-B0 untuk Klasifikasi Citra Penyakit Daun Jeruk," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 3, 2025.
- [10] A. R. R. Ivani et al., "Perbandingan VGG16 dan MobileNetV2 untuk Klasifikasi Tingkat Kematangan Buah Apel," *Jurnal Sistem Komputer Asia*, vol. 3, no. 1, 2025.
- [11] M. Irvai and D. Mahdalena, "Penerapan Teknik Masking dan Augmentasi pada Arsitektur CNN dan MobileNetV2 untuk Klasifikasi Fase Pertumbuhan Tomat," *JTIUST*, vol. 10, no. 1, 2025.
- [12] M. Han and C. Yi, "Deep Convolutional Neural Networks for Palm Fruit Maturity Classification," *arXiv:2502.20223*, 2025.
- [13] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015, doi: 10.1016/j.neunet.2014.09.003.
- [14] M. Tan and Q. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *Proc. ICML*, 2019.
- [15] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4510–4520, 2018.
- [16] Sahil, "Impact of Dataset Size on Deep Learning Model," *impact-of-dataset-size-on-deep-learning-model*, [Online]. Tersedia : <https://www.geeksforgeeks.org/deep-learning/impact-of-dataset-size-on-deep-learning-model/>, [Diakses : 28 April 2026]
- [17] Ali Hasan Md. Linkon, Md. Mahir Labib, Faisal Haque Bappy, Soumik Sarker, Marium-E-Jannat, Md Saiful Islam, "Deep Learning Approach Combining Lightweight CNN Architecture with Transfer Learning: An Automatic Approach for the Detection and Recognition of Bangladeshi Banknotes, " *11th International Conference on Electrical and Computer Engineering (ICECE)*, 2020.

