

IDENTIFIKASI TINGKAT KEMATANGAN BUAH PADA TANAMAN KELAPA SAWIT MENGGUNAKAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK DAN PENDEKATAN DEEP LEARNING

¹William Owen Wijaya✉, ¹Dhanny Rukmana Manday, ¹Agrifa Insani Napitupulu,
¹Mardi Turnip, ²Saroha Manurung

¹Fakultas Sains dan Teknologi, Universitas Prima Indonesia, Medan, Indonesia

²Institut Teknologi Sawit Indonesia (ITSI), Medan, Indonesia

Email: williamowenwijaya191204@gmail.com

ABSTRACT

Palm oil quality is largely determined by the free fatty acid (FFA) content, which is influenced by the ripeness of the fruit. Traditionally, determining the ripeness level of palm oil fruit relies on visual inspection by experts, which is time-consuming and dependent on individual skill. To address this, a system has been developed using the Convolutional Neural Network (CNN) method to automate the ripeness classification process. This study focuses on classifying palm oil fruit into three categories: ripe, unripe, and overripe, using a dataset of 1,380 images with 460 images per class. The dataset was split into 80% training data and 20% validation data. The CNN architecture employed was MobileNetV2, known for its simplicity and low computational complexity. Images were resized to 224 x 224 pixels, and two optimizers—Adam and RMSProp—were compared with learning rates of 0.001 and 0.0001 over 30 epochs. The best results were achieved using the Adam optimizer with a learning rate of 0.001, yielding a training accuracy of 91% and a test accuracy of 87%. This shows promising potential for automated palm oil fruit ripeness detection.

Keyword: Palm Oil, Convolutional Neural Network, MOBILENET, Adam, RMSProp.

ABSTRAK

Kualitas minyak kelapa sawit sebagian besar ditentukan oleh kandungan asam lemak bebas (ALB), yang dipengaruhi oleh kematangan buah. Secara tradisional, penentuan tingkat kematangan buah kelapa sawit bergantung pada inspeksi visual oleh para ahli, yang memakan waktu dan bergantung pada keterampilan individu. Untuk mengatasi hal ini, sebuah sistem telah dikembangkan menggunakan metode Convolutional Neural Network (CNN) untuk mengotomatiskan proses klasifikasi kematangan. Studi ini berfokus pada pengklasifikasian buah kelapa sawit menjadi tiga kategori: matang, belum matang, dan terlalu matang, menggunakan kumpulan data 1.380 gambar dengan 460 gambar per kelas. Kumpulan data dibagi menjadi 80% data pelatihan dan 20% data validasi. Arsitektur CNN yang digunakan adalah MobileNetV2, yang dikenal karena kesederhanaannya dan kompleksitas komputasi yang rendah. Gambar diubah ukurannya menjadi 224 x 224 piksel, dan dua optimizer Adam dan RMSProp dibandingkan dengan tingkat pembelajaran 0,001 dan 0,0001 selama 30 epoch. Hasil terbaik dicapai dengan menggunakan optimizer Adam dengan tingkat pembelajaran 0,001, menghasilkan akurasi pelatihan sebesar 91% dan akurasi pengujian sebesar 87%. Hal ini menunjukkan potensi yang menjanjikan untuk deteksi kematangan buah kelapa sawit secara otomatis.

Kata Kunci: Kelapa Sawit, Convolutional Neural Network, MOBILENET, Adam, RMSProp.

PENDAHULUAN

Minyak kelapa sawit merupakan salah satu produk pertanian yang memiliki nilai ekonomi tinggi dan diaplikasikan secara luas di berbagai sektor, termasuk biofuel, industri makanan, dan produk perawatan tubuh. Produksi minyak kelapa sawit di Indonesia dan Malaysia menyuplai sekitar 85% dari total produksi dunia, menjadikannya komoditas penting dalam perekonomian global (Teng et al., 2020; Ayompe et al., 2021). Meskipun demikian, penentuan tingkat kematangan buah kelapa sawit masih dilakukan

secara manual, yang memerlukan waktu, tenaga, dan konsentrasi tinggi.

Hal ini menjadi tantangan unik dalam industri ini karena ketepatan dalam menentukan kematangan buah sangat mempengaruhi kualitas minyak yang dihasilkan (Gulzar, 2023). Untuk mengatasi tantangan ini, penggunaan teknologi seperti **Convolutional Neural Network (CNN)** menjadi solusi yang menjanjikan. CNN adalah bagian dari Deep Neural Networks yang dirancang khusus untuk pengolahan

citra dan telah terbukti memberikan akurasi tinggi dalam klasifikasi objek (Triyogi et al., 2023).

Meskipun CNN menawarkan banyak keuntungan, terdapat beberapa tantangan dalam implementasinya, seperti keterbatasan data pelatihan dan kompleksitas model (Pulungan et al., 2021; Ashari et al., 2022). Oleh karena itu, penelitian ini bertujuan untuk mengeksplorasi dan menganalisis kesenjangan dalam literatur terkait penggunaan CNN untuk mendeteksi kematangan buah kelapa sawit.

Tujuan umum dari penelitian ini adalah untuk mengembangkan sistem otomatis yang dapat mendeteksi tingkat kematangan buah kelapa sawit dengan akurasi tinggi menggunakan CNN. **Tujuan khusus** mencakup analisis perbandingan antara metode CNN dengan metode lain dalam klasifikasi citra serta pengembangan model berbasis **MobileNet** untuk meningkatkan efisiensi pelatihan dan akurasi klasifikasi (Maulana, 2023; Al-Mashhadani & Park, 2023). Kontribusi dari penelitian ini diharapkan dapat memberikan dampak signifikan terhadap efisiensi produksi minyak kelapa sawit dan mengurangi ketergantungan pada metode manual.

Industri minyak kelapa sawit saat ini didominasi oleh negara-negara di Asia Tenggara, terutama Malaysia dan Indonesia, yang memiliki kondisi iklim ideal untuk produksi kelapa sawit. Sejak diperkenalkan pada abad ke-14, minyak kelapa sawit telah menjadi komoditas penting, terutama sebagai bahan utama biodiesel. Malaysia dan Indonesia menyuplai sekitar 85% dari total produksi minyak kelapa sawit dunia (Teng et al., 2020).

Pengolahan citra adalah teknik yang digunakan untuk memanipulasi gambar atau citra guna menghasilkan data yang diinginkan. Ini merupakan subdisiplin dalam bidang Kecerdasan Buatan yang memanfaatkan citra digital untuk menyelesaikan berbagai masalah. Metode pengolahan citra melibatkan perhitungan matematis pada level piksel atau geometris (Ratna, 2020).

CNN adalah arsitektur jaringan saraf yang dirancang khusus untuk analisis citra. CNN memiliki kemampuan untuk menangkap informasi pada berbagai skala melalui penggunaan lapisan konvolusi dan pooling. Hal ini memungkinkan CNN untuk melakukan ekstraksi fitur secara otomatis tanpa memerlukan pemrograman manual (Gulzar, 2023; Triyogi et al., 2023). Dalam konteks deteksi kematangan buah kelapa sawit, CNN dapat digunakan untuk mengklasifikasikan citra berdasarkan tingkat kematangan dengan akurasi tinggi (Saragih & Emanuel, 2021).

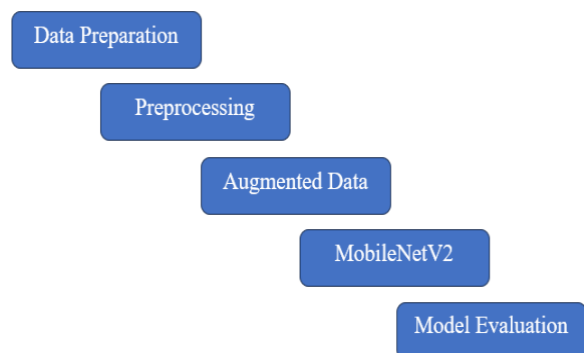
MobileNet adalah arsitektur CNN yang dirancang untuk perangkat dengan sumber daya

terbatas. Dengan menggunakan teknik pembelajaran transfer, MobileNet dapat mencapai kinerja tinggi sambil tetap efisien dalam penggunaan sumber daya. Dalam penelitian ini, MobileNetV2 digunakan sebagai model dasar dengan modifikasi tambahan untuk meningkatkan akurasi klasifikasi dan mengurangi tingkat kesalahan (Maulana, 2023; Al-Mashhadani & Park, 2023).

Meskipun banyak penelitian telah dilakukan mengenai penggunaan CNN dalam klasifikasi kematangan buah kelapa sawit, masih terdapat kesenjangan dalam analisis metodologi dan hasil yang diperoleh. Sebagian besar penelitian belum membahas secara mendalam tentang tantangan unik dalam penerapan CNN serta perbandingan dengan metode lain seperti Faster R-CNN atau teknik augmentasi data (Pulungan et al., 2021; Ashari et al., 2022). Penelitian ini bertujuan untuk mengisi kesenjangan tersebut dengan memberikan analisis komprehensif mengenai efektivitas metode CNN dalam mendeteksi kematangan buah kelapa sawit. Dengan demikian, penelitian ini memberikan kontribusi ganda yaitu pada pengembangan teknologi deteksi kematangan buah kelapa sawit tetapi juga memberikan wawasan baru mengenai penerapan CNN dalam konteks pertanian modern.

METODE PENELITIAN

Dalam penelitian yang kami lakukan ini, kami bertujuan untuk melakukan klasifikasi tingkat kematangan buah kelapa sawit menjadi tiga kelas, yaitu: matang, mentah, dan terlampau matang menggunakan data citra berupa gambar. Kami menggunakan salah satu arsitektur algoritma deep learning, yaitu Convolutional Neural Networks (CNN) dengan model MobileNetV2 (Achmad Naila Muna Ramadhani et al., 2023). Tujuan akhir dari penelitian ini adalah untuk mengukur tingkat akurasi model dalam mendeteksi tingkat kematangan buah sawit secara otomatis melalui gambar.



Gambar 1. Tahapan kerja

Persiapan Data

Pada bagian ini, kami menggunakan data citra buah kelapa sawit yang diambil dari website Kaggle dengan alamat <https://www.kaggle.com/datasets/ramadanizikri112/ripeness-of-oil-palm-fruit>. Data ini dibagi menjadi tiga kelas utama, yaitu: matang, mentah, dan terlampau matang, yang akan digunakan untuk melatih model dalam klasifikasi tingkat kematangan buah sawit.

Label Name	Data	Images
Ripe	460	
Unripe	460	
Over Ripe	460	
Total	1380	

Gambar 2. Persiapan Data

Berdasarkan Gambar 2, terdapat total 1.380 data citra yang terdiri dari tiga kelas berbeda, masing-masing dengan 460 data citra. Kelas-kelas tersebut adalah: matang, mentah, dan terlampau matang. Data citra ini kemudian dibagi menjadi dua bagian, yaitu data training dan data validasi. Data training digunakan untuk melatih model deep learning, sementara data validasi berfungsi untuk memvalidasi dan menguji performa model. Persentase pembagian data antara training dan validasi adalah 80% untuk training dan 20% untuk validasi.

Preprocessing

Setelah melakukan data preparation, langkah selanjutnya adalah memasuki tahap preprocessing. Preprocessing merupakan bagian dari tahap awal dalam pengolahan data citra yang bertujuan untuk menyiapkan data agar dapat diproses oleh model secara efektif. Pada penelitian ini, dilakukan dua proses utama dalam preprocessing, yaitu rescale dan resize:

1. Rescale: Proses ini digunakan untuk mengubah nilai piksel gambar yang semula berada dalam

rentang 0 hingga 255 pada skala RGB menjadi rentang 0 hingga 1, atau -1 hingga 1. Hal ini dilakukan untuk mempermudah model dalam memproses data citra, serta menghindari nilai besar yang dapat menghambat proses pembelajaran model (Mazdavilaya et al., 2024).

2. Resize: Gambar diubah ukurannya menjadi 224 x 224 piksel, sesuai dengan ukuran input yang dibutuhkan oleh model. Ukuran ini dipilih untuk memastikan bahwa model menerima data yang konsisten dan terformat dengan baik, sehingga mendukung peningkatan kinerja model dalam klasifikasi (Kurniawan et al., 2023).

Augmented Data

Tahap selanjutnya adalah augmentasi data, yang bertujuan untuk meningkatkan jumlah dan variasi data pelatihan menggunakan metode ImageDataGenerator. Augmentasi data ini dilakukan untuk membantu model mempelajari lebih banyak variasi citra dari berbagai sudut pandang, sehingga meningkatkan generalisasi dan kinerja model. Beberapa teknik augmentasi yang digunakan antara lain:

1. Rotasi: Mengubah sudut pandang citra dengan cara memutarnya, sehingga model dapat belajar dari berbagai orientasi gambar.
2. Flip horizontal: Membalik citra secara horizontal, sehingga model dapat memahami gambar dari sisi yang berbeda.
3. Zoom: Memperbesar atau memperkecil citra untuk memberikan variasi skala pada data dan memungkinkan model memahami detail dari berbagai skala.
4. Shift range: Menambahkan pergeseran acak pada gambar, baik secara horizontal maupun vertikal, yang membantu memperluas variasi data dan meningkatkan kemampuan generalisasi model.

Dengan augmentasi ini, model deep learning dapat dilatih pada dataset yang lebih beragam tanpa perlu menambah jumlah data asli, yang pada akhirnya memperkuat generalisasi dan akurasi model dalam klasifikasi citra (Arnandito & Sasongko, 2024).

MobileNetV2

MobileNetV2, yang dirilis pada tahun 2018, merupakan pengembangan lanjutan dari MobileNet yang dirilis pada tahun 2017. Model ini merupakan bagian dari Convolutional Neural Network (CNN) yang dikembangkan oleh Google untuk aplikasi mobile dan perangkat dengan sumber daya terbatas. MobileNetV2 berfokus pada efisiensi sambil tetap menjaga tingkat akurasi yang tinggi. Berikut adalah arsitektur dari MobileNetV2.

Tabel 1. Arsitektur MobileNetV2

Input	Operator	<i>t</i>	<i>c</i>	<i>n</i>	<i>s</i>
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	-

Sumber: [www. https://towardsdatascience.com](https://towardsdatascience.com)

Berdasarkan arsitektur pada Tabel 2 di atas, kita dapat melihat bahwa MobileNetV2 terdiri dari lapisan awal yang berupa konvolusi, yang diikuti oleh 17 inverted residual blocks. Model ini kemudian diakhiri dengan global average pooling dan fully connected layer. Seluruh tahapan di atas merupakan komponen utama dalam arsitektur MobileNetV2, yang dirancang untuk menghasilkan model yang efisien dalam hal komputasi dan ukuran (Li & Liu, 2021).

MobileNetV2 dapat dijalankan dengan memasukkan tiga input channel, yaitu lebar, tinggi, dan warna. Selain itu, model ini juga menggunakan weights dari ImageNet, yang berarti bahwa model dijalankan dengan menggunakan bobot yang telah dilatih sebelumnya oleh dataset ImageNet. Selanjutnya, `include_top=False` digunakan untuk menghilangkan bagian atas model, yaitu fully connected layer, dan diakhiri dengan `trainable=False`, yang berfungsi untuk membekukan parameter pada lapisan awal MobileNetV2 (Arnandito & Sasongko, 2024).

Dalam penelitian ini, kami akan menggunakan loss function yang berasal dari Categorical Crossentropy. Pemilihan ini didasarkan pada sifat dataset yang digunakan, yang merupakan multi-class classification dengan tiga kelas tingkat kematangan. Label target dalam dataset ini dinyatakan dalam bentuk one-hot encoding. Tujuan dari penggunaan Categorical Crossentropy adalah untuk meminimalkan loss selama proses pelatihan, sehingga model dapat belajar dengan lebih efektif dalam membedakan antara ketiga kelas tersebut.

Untuk hyperparameter yang digunakan dalam penelitian ini, kami memilih Adam optimizer dan RMSProp optimizer. Kami mengatur learning rate pada masing-masing optimizer, yaitu 0,001 dan 0,0001, yang akan berpengaruh pada kecepatan model deep learning dalam beradaptasi dengan data.

Selanjutnya, model akan dilatih dengan jumlah epoch sebanyak 30 kali, di mana setiap nilai dalam

epoch menunjukkan jumlah data training yang dibagi dengan jumlah batch, yang disebut sebagai iteration. Untuk memantau perkembangan model selama pelatihan, kami menggunakan Callback berupa EarlyStopping. Fungsi EarlyStopping ini berguna untuk menghentikan pelatihan model setelah jumlah patience yang ditentukan, jika tidak ada peningkatan pada beberapa epoch secara berturut-turut.

Model Evaluation

Pada tahap ini evaluasi merupakan hal terpenting untuk mengukur kinerja model, apakah model berhasil melakukan klasifikasi tingkat kematangan buah sawit berdasarkan data citra yang tersedia. Metode yang paling umum digunakan dalam mengukur sebuah metrik adalah Confusion matrix yang biasanya memiliki peran untuk melakukan perhitungan performa metrik, pada model klasifikasi seperti penggunaan recall, precision, f1-score dan tentunya akurasi (Parinduri et al., 2023).

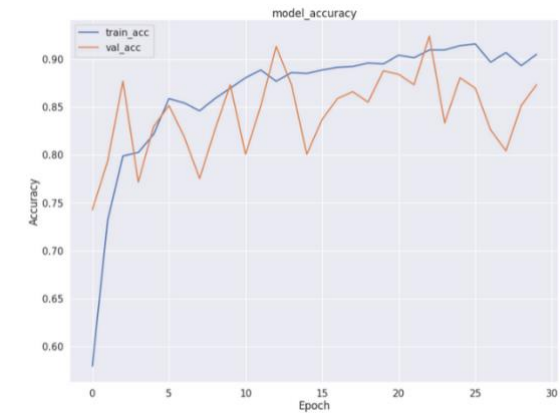
HASIL DAN PEMBAHASAN

Hasil

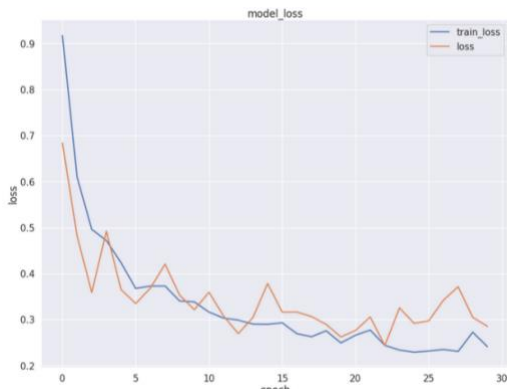
Proses training dilakukan menggunakan arsitektur MobileNetV2 yang sebelumnya telah dilatih. Model ini digunakan untuk mengklasifikasikan tingkat kematangan buah sawit dengan menggunakan learning rate 0,001, dan dua optimizer yang dibandingkan, yaitu Adam dan RMSProp. Tujuan dari eksperimen ini adalah untuk meningkatkan akurasi model, dengan hasil yang ditampilkan pada tabel dibawah.

Tabel 2. Hasil Pengujian dengan Learning Rate 0,001

Model	Learning Rate 0,001			
	Acc	Loss	Val_Acc	Val_Loss
Adam	0.9100	0.2360	0.8225	0.3655
RmsProp	0.9101	0.2302	0.8732	0.2855



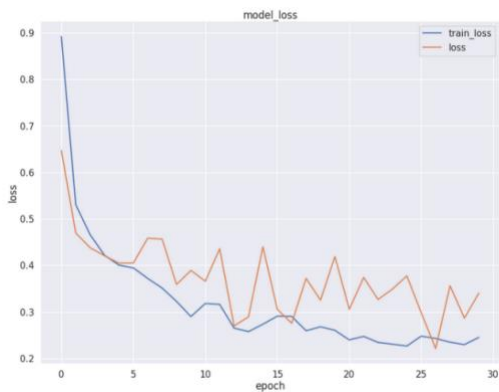
Gambar 3. Grafik Accuracy RmsProp 0,001



Gambar 4. Grafik Loss RmsProp 0,001



Gambar 5. Grafik Accuracy Adam 0,001



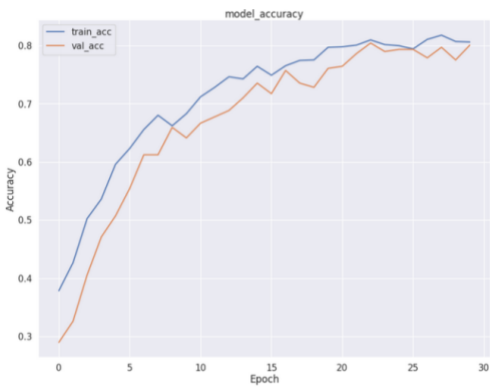
Gambar 6. Grafik Loss Adam 0,001

Dapat dilihat dari tabel 2 yang menunjukkan hasil perbandingan klasifikasi tingkat kematangan buah sawit antara 2 optimizer, yaitu Adam dan RMSProp, dengan learning rate sebesar 0,001. berdasarkan hasil ini, penggunaan optimizer RMSProp menghasilkan akurasi pelatihan sebesar 91% dan akurasi validasi 87%. Selain itu, nilai loss yang dihasilkan oleh model sebesar 23% dan validasi loss sebesar 28%. sebaliknya, optimizer Adam dengan learning rate yang sama memberikan nilai akurasi pelatihan 91% dan akurasi validasi 82%, dengan loss pelatihan 23% dan validasi loss sebesar 36%. Pada percobaan pertama ini dapat disimpulkan bahwa optimizer RMSProp memiliki

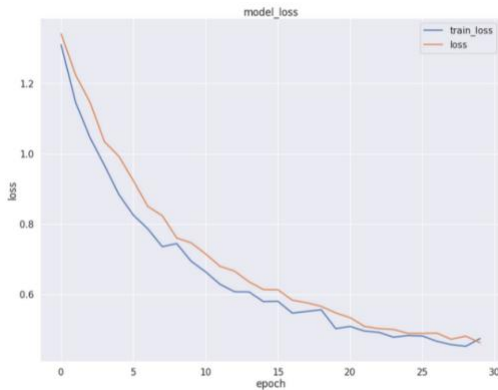
performa yang lebih baik, loss yang dihasilkan juga lebih rendah terhadap data pelatihan maupun validasi. Namun ini akan dibandingkan kembali dengan hasil dari pada percobaan kedua yang menggunakan learning rate 0,0001.

Tabel 3. Hasil Pengujian dengan Learning Rate 0,0001

Model	Learning Rate 0,0001			
	Acc	Loss	Val_Acc	Val_Loss
Adam	0.8051	0.5007	0.8007	0.4621
RmsProp	0.8206	0.4478	0.7826	0.4807



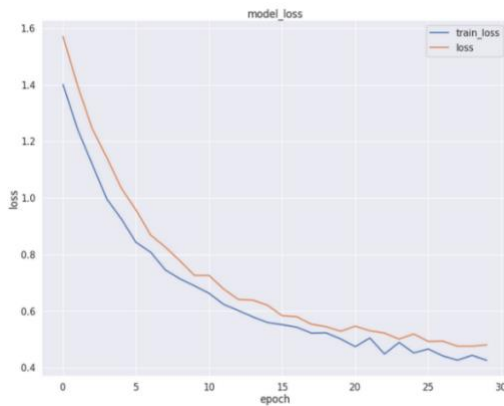
Gambar 7. Grafik Accuracy Adam 0,0001



Gambar 8. Grafik Loss Adam 0,0001



Gambar 9. Grafik Accuracy RmsProp 0,0001



Gambar 10. Grafik Loss RmsProp 0,0001

Pada percobaan kedua, kami menggunakan learning rate sebesar 0,0001 dengan dua optimizer berbeda, yaitu Adam dan RMSProp. Hasil yang diperoleh menunjukkan bahwa optimizer Adam menghasilkan akurasi pelatihan sebesar 80%, akurasi validasi 80%, dengan loss pelatihan sebesar 50% dan validasi loss 46%. Sedangkan, pada optimizer RMSProp, akurasi pelatihan yang dicapai, yaitu 82%, tetapi akurasi validasinya sedikit lebih rendah, sebesar 78%, dengan loss pelatihan 44% dan validasi loss 48%. Dari hasil ini, meskipun akurasi pelatihan pada optimizer adam lebih rendah, optimizer Adam menunjukkan performa yang lebih unggul dalam hal validasi akurasi dan validasi loss. Hal ini menunjukkan bahwa Adam lebih baik dalam meningkatkan performa model klasifikasi tingkat kematangan sawit menggunakan arsitektur MobileNetV2 dibandingkan dengan RMSProp.

Pembahasan

Selanjutnya, kami akan membandingkan performa optimizer terbaik dari dua percobaan sebelumnya, yang menggunakan learning rate 0,001 dan 0,0001, dengan bantuan confusion matrix. Penggunaan confusion matrix sangat membantu dalam memberikan gambaran yang lebih jelas dan komprehensif mengenai kinerja model (Dewi et al., 2024). Confusion matrix bekerja dengan cara menampilkan hasil dari jumlah prediksi yang benar dan salah yang dihasilkan oleh model, melalui empat komponen utama:

True Positive (TP) = jumlah prediksi yang benar untuk kelas positive.

False Positive (FP) = Jumlah prediksi yang salah untuk kelas positive.

False Negative (FN) = jumlah prediksi yang salah untuk kelas negative.

True Negative (TN) = jumlah prediksi yang benar untuk kelas negative.

Dari keempat komponen utama dihasilkan performa matriks yang terdiri dari Accuracy, Precision, Recall, F1-Score. Dimana performa metrics tersebut digunakan untuk mengukur kinerja model.

Accuracy

Akurasi adalah rasio jumlah prediksi yang benar terhadap total jumlah prediksi. Yang biasanya digunakan untuk mengukur seberapa baik suatu model dalam melakukan klasifikasi secara keseluruhan.

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

Precision

Precision menggambarkan seberapa banyak prediksi positif yang benar benar positif. Biasanya digunakan mengukur akurasi prediksi positif, yaitu proporsi dari prediksi positif yang benar.

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

Recall

Recall menggambarkan proses pengukuran dari banyak total kasus positif yang benar benar terdeteksi oleh model.

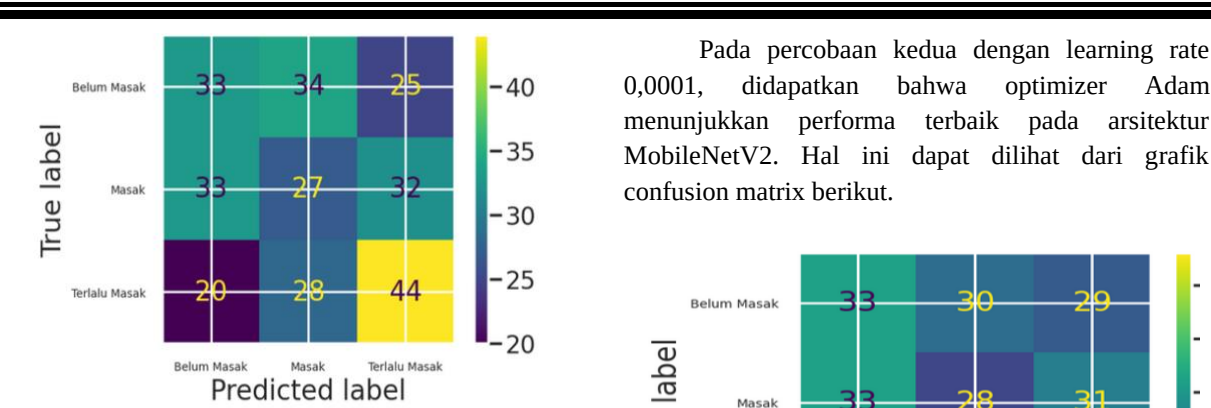
$$Recall = \frac{TP}{TP + FN} \quad (3)$$

F1-Score

F1-Score adalah nilai rata rata dari hasil presisi dan recall. yang biasanya digunakan bila ingin menyeimbangkan nilai dari presisi dan recall untuk mendapatkan nilai terbaik dari kedua matriks tersebut.

$$F1\ Score = 2 \times \frac{Recall \times Precision}{Recall + Precision}$$

Pada percobaan pertama dengan penggunaan learning rate 0,001, diperoleh bahwa performa optimizer RMSProp lebih unggul dibandingkan Adam dalam arsitektur MobileNetV2. Hal ini dapat dilihat dari grafik confusion matrix berikut.



Gambar 11. Grafik Confusion Matrix Learning Rate 0,001 RMSProp

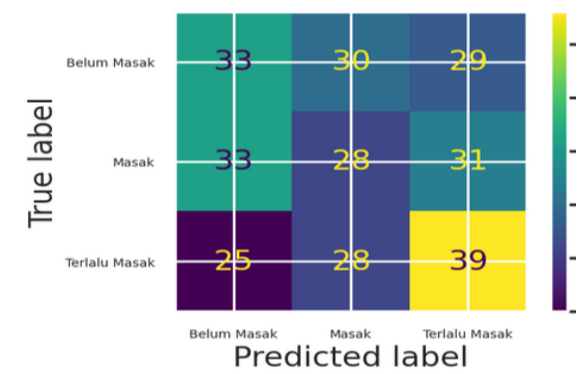
Tabel 4. Hasil Nilai Validasi Dengan Penggunaan Confusion Matrix

	Precision	Recall	F1-score	Support
Belum Masak	0.38	0.36	0.37	92
Masak	0.30	0.29	0.30	92
Terlalu Masak	0.44	0.48	0.46	92
Accuracy Macro avg	0.37	0.38	0.38	276
Weighted avg	0.37	0.38	0.38	276

Berdasarkan grafik confusion matrix di atas, hasil pengujian dengan 20% data validasi menunjukkan bahwa model berhasil mengklasifikasikan buah sawit dengan beberapa hasil yang bervariasi. Model mampu mengklasifikasikan buah masak dengan benar sebanyak 27 kali, namun salah sebanyak 65 kali. Untuk keadaan buah yang belum masak, model berhasil mengklasifikasikannya dengan benar sebanyak 33 kali dan salah sebanyak 59 kali. Sementara itu, buah yang terlalu masak diklasifikasikan dengan benar sebanyak 44 kali dan salah sebanyak 48 kali.

Secara keseluruhan, model ini menghasilkan akurasi sebesar 38%. Dari sisi precision, nilai yang diperoleh adalah 38% untuk buah belum masak, 30% untuk buah masak, dan 44% untuk buah terlalu masak. Dari segi recall, hasil yang diperoleh adalah 36% untuk buah belum masak, 29% untuk buah masak, dan 44% untuk buah terlalu masak. Adapun nilai F1-Score yang dihasilkan adalah 37% untuk buah belum masak, 30% untuk buah masak, dan 46% untuk buah terlalu masak. Hasil ini menunjukkan bahwa model memiliki kinerja yang lebih baik dalam mengklasifikasikan buah yang terlalu masak dibandingkan dengan dua kategori lainnya.

Pada percobaan kedua dengan learning rate 0,0001, didapatkan bahwa optimizer Adam menunjukkan performa terbaik pada arsitektur MobileNetV2. Hal ini dapat dilihat dari grafik confusion matrix berikut.



Gambar 12. Grafik Confusion Matrix Learning Rate 0,0001 Adam

Tabel 5. Hasil Nilai Validasi Dengan Penggunaan Confusion Matrix

	Precision	Recall	F1-score	Support
Belum Masak	0.36	0.36	0.36	92
Masak	0.33	0.30	0.31	92
Terlalu Masak	0.39	0.42	0.41	92
Accuracy Macro avg	0.36	0.36	0.36	276
Weighted avg	0.36	0.36	0.36	276

Berdasarkan grafik confusion matrix, hasil klasifikasi tingkat kematangan buah sawit menunjukkan bahwa kelas belum masak diklasifikasikan dengan benar sebanyak 33 kali, sementara 59 kali salah diprediksi. Untuk kelas masak, 28 kali diklasifikasikan dengan benar dan 64 kali salah diprediksi. Sedangkan untuk kelas terlalu masak, 39 kali diklasifikasikan dengan benar, dengan 53 kali salah diprediksi. Dari confusion matrix ini, diperoleh nilai akurasi sebesar 36%. Untuk precision, nilai yang dihasilkan adalah 36% untuk kelas belum masak, 33% untuk kelas masak, dan 39% untuk kelas terlalu masak. Recall yang dihasilkan adalah 36% untuk belum masak, 30% untuk masak, dan 42% untuk terlalu masak. Sedangkan F1-Score yang dihasilkan adalah 36% untuk belum masak, 31% untuk masak, dan 41% untuk terlalu masak.

KESIMPULAN

Dalam penelitian ini, telah dilakukan klasifikasi tingkat kematangan buah sawit menggunakan algoritma Convolutional Neural Network (CNN) dengan arsitektur MobileNetV2. Data yang digunakan dibagi menjadi 80% untuk pelatihan dan 20% untuk validasi. Hasil eksperimen membuktikan bahwa penggunaan hyperparameter dengan learning rate 0,001 dan optimizer RMSProp menghasilkan performa terbaik dibandingkan penggunaan optimizer Adam dengan learning rate 0,0001. Pada tahap pelatihan, model berhasil mencapai akurasi sebesar 91% dan validasi akurasi 87%. Namun, dari hasil evaluasi melalui confusion matrix, model mengalami overfitting dan underfitting, di mana nilai accuracy yang di dapat hanya 38%. Model juga cenderung lebih baik dalam mengklasifikasikan kelas "terlalu masak" dibandingkan kelas masak dan belum masak. Meskipun hasil pelatihan menunjukkan akurasi yang tinggi, kesenjangan antara akurasi pelatihan dan validasi mengindikasikan perlunya perbaikan lebih lanjut. Untuk penelitian mendatang, disarankan untuk menggunakan dataset yang lebih besar dan lebih bersih, serta mempertimbangkan penggunaan variasi hyperparameter seperti learning rate, ukuran batch, dan dropout. Hal ini diharapkan dapat mengurangi masalah overfitting dan menghasilkan model yang lebih generalis untuk klasifikasi tingkat kematangan buah sawit. Selain itu, peningkatan jumlah epoch juga dapat membantu model dalam mencapai performa yang lebih baik dan stabil. Dengan langkah-langkah ini, diharapkan kinerja model dapat ditingkatkan untuk penggunaan praktis di lapangan.

DISEMINASI

Artikel ini telah diseminasikan pada Seminar Nasional Teknologi Informasi dan Komunikasi (SEMNASTIK) APTIKOM Tahun 2024 yang diselenggarakan oleh Universitas Methodist Indonesia pada tanggal 24-26 Oktober 2024.

DAFTAR PUSTAKA

- Achmad Naila Muna Ramadhani, Galuh Wilujeng Saraswati, Rama Tri Agung, & Heru Agus Santoso. (2023). Performance Comparison of Convolutional Neural Network and MobileNetV2 for Chili Diseases Classification. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 7(4), 940–946. <https://doi.org/10.29207/resti.v7i4.5028>
- Al-Mashhadani, Z., & Park, J.-H. (2023). Autonomous agricultural monitoring robot for efficient smart farming. *2023 23rd International Conference on Control, Automation and Systems (ICCAS)*, 640–645.
- Arnandito, S., & Sasongko, T. B. (2024). Comparison of EfficientNetB7 and MobileNetV2 in Herbal Plant Species Classification Using Convolutional Neural Networks. *Journal of Applied Informatics and Computing*, 8(1), 176–185. <https://doi.org/10.30871/jaic.v8i1.7927>
- Ashari, S., Yanris, G. J., & Purnama, I. (2022). Oil Palm Fruit Ripeness Detection using Deep Learning. *Sinkron: Jurnal Dan Penelitian Teknik Informatika*, 6(2), 649–656.
- Ayompe, L. M., Schaafsma, M., & Egho, B. N. (2021). Towards sustainable palm oil production: The positive and negative impacts on ecosystem services and human wellbeing. *Journal of Cleaner Production*, 278, 123914.
- Dewi, S., Ramadhani, F., & Djasmayena, S. (2024). Klasifikasi Jenis Jerawat Berdasarkan Gambar Menggunakan Algoritma CNN (Convolutional Neural Network). *Hello World Jurnal Ilmu Komputer*, 3(2), 68–73. <https://doi.org/10.56211/helloworld.v3i2.518>
- Gulzar, Y. (2023). Fruit image classification model based on MobileNetV2 with deep transfer learning technique. *Sustainability*, 15(3), 1906.
- Kurniawan, R., Martadinata, A. T., & Cahyo, S. D. (2023). Klasifikasi Tingkat Kematangan Buah Sawit Berbasis Deep Learning dengan Menggunakan Arsitektur Yolov5. *Journal of Information System Research (JOSH)*, 5(1), 302–309. <https://doi.org/10.47065/josh.v5i1.4408>
- Li, W., & Liu, K. (2021). Confidence-aware object detection based on mobilenetv2 for autonomous driving. *Sensors*, 21(7). <https://doi.org/10.3390/s21072380>
- Maulana, T. (2023). *Peningkatan Kualitas Pada Produksi Minyak Kelapa Sawit (Cpo) Pt. Blankahan Oil Mill Dengan Menggunakan Metode Seven Tools*. Universitas Medan Area.
- Mazdavilaya, T. K., Yanto, F., Budianita, E., Sanjaya, S., & Syafria, F. (2024). Implementasi VGG 16 dan Augmentasi Zoom Untuk Klasifikasi Kematangan Sawit. 4(6), 2881–2891. <https://doi.org/10.30865/klik.v4i6.1940>
- Parinduri, S. K., Rosnelly, R., Purnama, A., Sihotang, A., & Adelina, M. C. (2023). Perbandingan Algoritma Support Vector Machine Dan K-Nearest Neighbors Pada Kematangan Buah Sawit. *Device*, 13(2), 147–155. <https://doi.org/10.32699/device.v13i2.5400>
- Pulungan, A. B., Nafis, Z., Anwar, M., Hastuti, H., Hamdani, H., & Myori, D. E. M. (2021). Object detection with a webcam using the Python programming language. *Journal of Applied Engineering and Technological Science (JAETS)*, 2(2), 103–111.
- Ratna, S. (2020). Pengolahan citra digital dan histogram dengan phyton dan text editor phycharm. *Technologia: Jurnal Ilmiah*, 11(3), 181–186.

- Saragih, R. E., & Emanuel, A. W. R. (2021). Banana ripeness classification based on deep learning using convolutional neural network. *2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT)*, 85–89.
- Teng, S., Khong, K. W., & Ha, N. C. (2020). Palm oil and its environmental impacts: A big data analytics study. *Journal of Cleaner Production*, 274, 122901.
- Triyogi, R., Magdalena, R., & Hidayat, B. (2023). Mendeteksi Kematangan Buah Kelapa Sawit Menggunakan Convolutional Neural Network Deep Learning. *Jurnal Nasional SAINS Dan TEKNIK; Vol 1 No 1 (1): December 2023* DO - 10.25124/Jnst.V1i1.6732.