

SISTEM PREDIKSI TINGKAT KEMATANGAN BUAH TOMAT MENGUNAKAN ARSITEKTUR VGG16

Rayhan Dzikri Fauzi¹, Muhammad Fachrie²

1,2) Informatika, Fakultas Sains & Teknologi, Universitas Teknologi Yogyakarta, Indonesia

Article Info

Article history:

Received: 03 Oktober 2024

Revised: 13 Oktober 2024

Accepted: 23 Oktober 2024

ABSTRACT

Abstrak

Penelitian ini bertujuan untuk mengembangkan sistem prediksi tingkat kematangan buah tomat menggunakan metode Convolutional Neural Networks (CNN). Penentuan kematangan tomat yang dilakukan secara manual sering kali memerlukan banyak waktu, tenaga, dan rentan terhadap kesalahan karena subjektivitas penilaian manusia. Dalam konteks industri pengolahan pangan dan pertanian, penilaian manual ini dapat menyebabkan ketidakakuratan dan kerugian ekonomi. Oleh karena itu, penelitian ini berfokus pada penerapan teknologi kecerdasan buatan (AI), khususnya CNN, untuk mengotomatisasi dan meningkatkan akurasi penilaian kematangan tomat. Dataset yang digunakan dalam penelitian ini berupa gambar tomat dalam kategori matang, setengah matang, dan mentah yang diperoleh dari platform Kaggle. Dua arsitektur CNN, yaitu Simple CNN dan VGG16, diterapkan untuk melakukan klasifikasi. Setelah proses pre-processing dan augmentasi data, kedua model dilatih dan diuji. Hasil penelitian menunjukkan bahwa model VGG16 memiliki performa yang lebih baik dibandingkan Simple CNN, dengan akurasi 97,17% pada data latih dan 95,56% pada data validasi. Kesimpulannya, model VGG16 lebih unggul dalam memprediksi tingkat kematangan tomat dan dapat diimplementasikan untuk meningkatkan efisiensi dan akurasi dalam penilaian kematangan buah, yang berpotensi memberikan manfaat besar bagi industri pertanian dan pengolahan makana.

Kata Kunci: *Augmentasi Data, Convolutional Neural Networks, Kematangan Buah, Simple CNN, VGG16*

Abstract

This study aims to develop a tomato ripeness prediction system using Convolutional Neural Networks (CNN). Manual assessment of tomato ripeness often requires significant time and effort and is prone to errors due to human subjectivity. In the context of food processing and agriculture, this manual approach can lead to inaccuracies and economic losses. Therefore, this research focuses on applying artificial intelligence (AI) technology, particularly CNN, to automate and enhance the accuracy of tomato ripeness assessment. The dataset used in this study consists of tomato images in three categories: ripe, half-ripe, and unripe, obtained from the Kaggle platform. Two CNN architectures, Simple CNN and VGG16, were implemented for classification. After data pre-processing and augmentation, both models were trained and tested. The results show that the VGG16 model outperforms Simple CNN, achieving 97.17% accuracy on training data and 95.56% on validation data. In conclusion, the VGG16 model is superior in predicting tomato ripeness and can be implemented to improve the efficiency and accuracy of ripeness assessment, potentially providing significant benefits to the agricultural and food processing industries.

Keywords: *Convolutional Neural Networks, Data Augmentation, Fruit Ripeness, Simple CNN, VGG16*

Djtechno: Jurnal Teknologi Informasi oleh Universitas
Dharmawangsa Artikel ini bersifat open access yang

didistribusikan di bawah syarat dan ketentuan dengan Lisensi Internasional Creative Commons Attribution NonCommercial ShareAlike 4.0 ([CC-BY-NC-SA](https://creativecommons.org/licenses/by-nc-sa/4.0/)).



Corresponding Author:
E-mail : dzikrirehan8@gmail.com

1. PENDAHULUAN

Tomat merupakan salah satu komoditas hortikultura yang sangat penting di Indonesia karena memiliki nilai ekonomi yang tinggi dan digunakan dalam berbagai produk makanan (Sigitta et al., 2023). Salah satu aspek utama yang menentukan kualitas tomat adalah tingkat kematangannya, yang mempengaruhi rasa, tekstur, dan kandungan nutrisinya (Ishak et al., 2022). Salah satu alasan tomat tetap menjadi komoditas panen terbesar di Indonesia adalah iklim yang cocok untuk budidaya tomat (Palupiningsih et al., 2023). Saat ini, penentuan tingkat kematangan tomat umumnya masih dilakukan secara manual oleh petani atau pekerja di industri pengolahan tomat. Pendekatan manual ini tidak hanya memakan waktu dan tenaga, tetapi juga rentan terhadap kesalahan akibat subjektivitas penilaian manusia (Helsaputra et al., 2021). Dalam skala besar, penilaian manual ini bisa menyebabkan ketidakakuratan yang signifikan dan mengakibatkan kerugian ekonomi.

Seiring dengan kemajuan teknologi, penerapan kecerdasan buatan (AI) menjadi solusi yang potensial untuk mengatasi masalah ini. Salah satu metode yang paling menonjol dalam pemrosesan citra untuk klasifikasi adalah *Convolutional Neural Network (CNN)*. CNN mampu memproses data visual dengan sangat efektif dan telah berhasil diterapkan dalam berbagai penelitian untuk mengidentifikasi tingkat kematangan buah, termasuk tomat (Soekarta et al., 2023). CNN dipresentasikan dalam bentuk tiga dimensi atau didesain untuk mengolah data tiga dimensi yaitu dimensi dari masukan gambar (*height and weight*) dan *depth* (kedalaman) (Saxena, 2022). Teknologi ini memberikan peluang untuk mengotomatisasi proses penentuan tingkat

kematangan tomat, sehingga mempercepat proses dan meningkatkan akurasi hasil penilaian.

Citra digital merupakan gambar yang direpresentasikan dalam bentuk matriks dua dimensi yang terdiri dari piksel-piksel dengan nilai intensitas tertentu. Pemrosesan citra digital melibatkan teknik untuk menganalisis dan mengklasifikasikan objek berdasarkan pola visual seperti warna, tekstur, dan bentuk (Fathurrahman et al., 2019). Dalam konteks prediksi kematangan buah, citra digital dari tomat dianalisis untuk mengidentifikasi karakteristik visual yang berkaitan dengan kematangan.

Deep learning merupakan salah satu cabang kecerdasan buatan (AI) yang menggunakan jaringan saraf tiruan dengan lapisan tersembunyi (*hidden layers*) untuk memproses data secara lebih dalam dan kompleks. Metode deep learning, khususnya CNN, telah terbukti efektif dalam mendeteksi penyakit tanaman secara otomatis dan akurat (Athallah Tsany Rakha Dzaky, 2021). Teknologi deep learning digunakan untuk mengenali pola-pola yang ada dalam data yang sangat besar, terutama data yang tidak terstruktur seperti gambar, teks, atau suara (Biringkanae & Raza Bunahri, 2023). Salah satu keuntungan utama dari *deep learning* adalah kemampuannya untuk secara otomatis mengekstraksi fitur-fitur dari data tanpa memerlukan campur tangan manusia secara eksplisit dalam menentukan fitur-fitur tersebut (Helsaputra et al., 2021). *Deep learning* menggunakan beberapa lapisan tersembunyi yang masing-masing mampu mempelajari pola-pola yang lebih kompleks dengan tingkat abstraksi yang lebih tinggi. Pada setiap lapisan, informasi diproses dan diklasifikasikan, dengan hasil akhir yang dihasilkan oleh lapisan output. CNN merupakan suatu algoritma atau metode yang digunakan pada *deep learning* yang sangat banyak digunakan dalam pengolahan gambar (Pouyanfar et al., 2018).

Convolutional Neural Network (CNN) adalah jenis jaringan saraf tiruan yang secara khusus dirancang untuk mengolah data berbentuk citra. CNN memiliki kemampuan untuk secara otomatis mengekstraksi fitur-fitur penting dari gambar, seperti tepi, pola, dan tekstur, yang sangat berguna dalam klasifikasi gambar (Ghosh et al., 2019). CNN memiliki kemampuan untuk menerima input berupa citra dan

mempelajari aspek-aspek di dalamnya, melakukan ekstraksi fitur secara otomatis, dan kemudian melakukan klasifikasi berdasarkan fitur-fitur yang diperoleh (Dwi Suci Anggraeni et al., 2022). CNN terdiri dari beberapa lapisan, termasuk lapisan konvolusi yang bertugas melakukan operasi konvolusi pada input gambar untuk menghasilkan peta fitur (*feature map*) yang merepresentasikan informasi penting dari citra tersebut. CNN telah banyak digunakan dalam berbagai aplikasi pemrosesan citra, termasuk klasifikasi penyakit tanaman dan prediksi kematangan buah (Putri Ananda et al., 2023).

Meskipun penentuan tingkat kematangan tomat merupakan faktor penting yang memengaruhi kualitas dan nilai ekonomis produk, metode penilaian manual yang saat ini banyak digunakan oleh petani dan industri pengolahan tomat memiliki beberapa kelemahan. Metode ini memerlukan banyak waktu dan tenaga serta rentan terhadap ketidakakuratan karena subjektivitas manusia. Selain itu, dalam skala besar, metode manual dapat menyebabkan ketidakakuratan yang signifikan dan kerugian ekonomi.

Dengan perkembangan teknologi kecerdasan buatan, khususnya deep learning dan CNN, terdapat peluang untuk mengotomatiskan proses penilaian kematangan tomat secara lebih efisien dan akurat. Namun, sejauh mana teknologi ini mampu mengatasi berbagai tantangan yang ada, seperti variasi kondisi pencahayaan, bentuk, serta tekstur tomat yang mempengaruhi hasil pemrosesan citra, belum sepenuhnya dipahami dan dioptimalkan dalam konteks spesifik di Indonesia.

2. METODE PENELITIAN

Cara melakukan klasifikasi dengan menggunakan model *Convolutional Neural Network* (CNN) melibatkan beberapa tahapan seperti yang digambarkan pada Gambar 1. Pertama, data dikumpulkan dari berbagai sumber, Setelah data terkumpul, proses *pre-processing* dilakukan untuk membersihkan dan mempersiapkan data tersebut, misalnya dengan normalisasi dan Augmentasi Data. Setelah *pre-processing* selesai, model CNN diterapkan pada data tersebut. Proses *training model* kemudian dilakukan

untuk melatih model CNN dengan data yang telah diproses. Akhirnya, model yang telah dilatih dievaluasi untuk menilai kinerja dan akurasinya dalam melakukan tugas klasifikasi yang diinginkan.

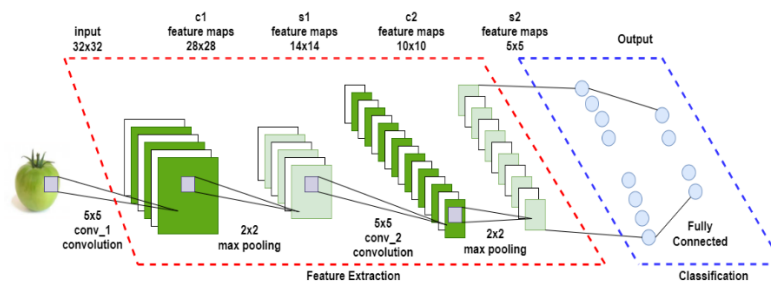


Gambar 1. Alur Penelitian

Alur penelitian berdasarkan pada Gambar 1. Lebih rinci dijelaskan sebagai berikut :

- 1) Pengumpulan Data: Data untuk penelitian ini berupa dataset gambar buah tomat dalam berbagai kondisi, seperti matang, mentah, setengah matang, dan busuk yang diperoleh dari platform Kaggle. Data dikumpulkan melalui dua metode utama, yaitu pengambilan langsung gambar buah tomat dan pengumpulan dataset dari Kaggle. Data diperoleh dengan melakukan pengambilan gambar langsung terhadap buah tomat. Proses ini dilakukan untuk mendapatkan gambar yang beragam dan representatif, mencakup contoh gambar dari setiap tingkat kematangan buah tomat. Pengambilan gambar dilakukan dengan memperhatikan faktor-faktor seperti pencahayaan, sudut pengambilan, dan resolusi gambar untuk memastikan kualitas data yang baik.
- 2) Pre-Processing Data: Membagi data mentah menjadi 2 yaitu data uji dan data validasi. Kedua data tersebut dibagi menjadi 70% untuk data uji dan 30% untuk data validasi. Fungsi pembagian data ini bertujuan untuk melatih model yang akan dibuat serta menghitung akurasi dari pelatihan model tersebut. Lalu ada Proses augmentasi data, proses ini akan melibatkan empat jenis pemrosesan, termasuk RandomFlip, RandomRotation, RandomZoom, dan Rescaling. Tujuan dari augmentasi data ini adalah untuk mengurangi risiko *overfitting* dengan melakukan penyesuaian seminimal mungkin pada data asli.
- 3) Model CNN: Metode dengan menggunakan *Convolutional Neural Network* (CNN) merupakan salah satu metode dari machine learning yang merupakan pengembangan dari *Multi Layer Perceptron* (MLP) yang mana dirancang untuk mengolah atau membuat data dari dua dimensi. CNN juga merupakan salah

satu jenis metode dari *Deep Neural Network* yang dikarenakan didalamnya memiliki tingkat jaringan dan mempunyai banyak penerapan yang dilakukan di dalam citra (Vicky Purnama Putra et al., 2023). Model yang digunakan sebagai perbandingan yaitu menggunakan dua model Simple CNN dan VGG16. Keluaran dari tahapan pembentukan model adalah model dengan ekstensi .h5 yang digunakan selanjutnya untuk proses pelatihan. Secara umum CNN terdiri dari dua lapisan umum yaitu lapisan feature learning dan lapisan classification seperti yang ditunjukkan pada gambar 2 dibawah (Albawi et al., 2017).

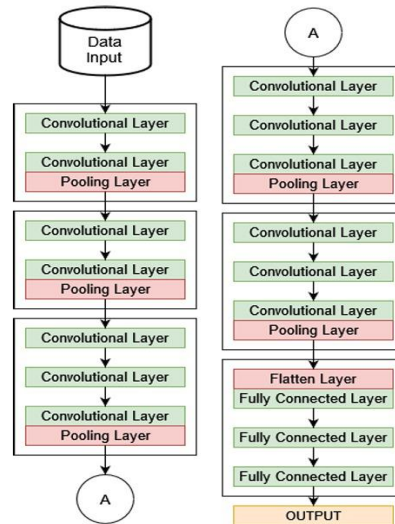


Gambar 2. Arsitektur CNN

Proses yang terjadi dalam VGG16 pada dasarnya tidak berbeda jauh dengan uraian yang telah dijelaskan sebelumnya oleh peneliti. Namun, ada sedikit perbedaan dalam cara menghitung jumlah lapisan serta menamai setiap lapisan tersebut. Pada Gambar 2, peneliti memberikan penjelasan global mengenai lapisan-lapisan mana yang akan dilewati oleh data gambar hingga menghasilkan nilai pada lapisan output.

Sementara pada Gambar 3, terlihat bahwa beberapa nama lapisan tidak tercantum. Hal ini bukan berarti lapisan-lapisan tersebut tidak ada, namun proses-proses yang sebelumnya telah dijelaskan akan tergabung pada satu lapisan. Sehingga satu lapisan dapat memuat lebih dari satu proses yang berlangsung. Dengan demikian, VGG16 memiliki jumlah lapisan sebanyak 16, di mana terdiri atas 13 lapisan konvolusi dan 3 lapisan Fully Connected yang dapat dilihat pula pada Gambar 3 secara visualnya. Lapisan Rectified Linear Unit

(ReLU) pada Gambar 3 nampak tidak ada, sebab lapisan ini sudah sekaligus dijalankan prosesnya menjadi satu pada setiap lapisan konvolusi sebelum dilanjutkan ke tahap berikutnya (Weny Indah Kusumawati & Adisaputra Zidha Noorizki, 2023).



Gambar 3. Arsitektur VGG16

- 4) *Training Model*: *Training model* dilatih dengan fungsi kerugian *Categorical Crossentropy* dan *optimizer AdamW*. Proses pelatihan menggunakan 10, 20 hingga 50 *epoch* untuk menghindari *overfitting*.
- 5) *Evaluation Model*: Kinerja model dievaluasi berdasarkan akurasi pada data latih dan validasi. Pengujian dilakukan menggunakan model Simple CNN dan VGG16, dengan VGG16 menunjukkan performa yang lebih baik.

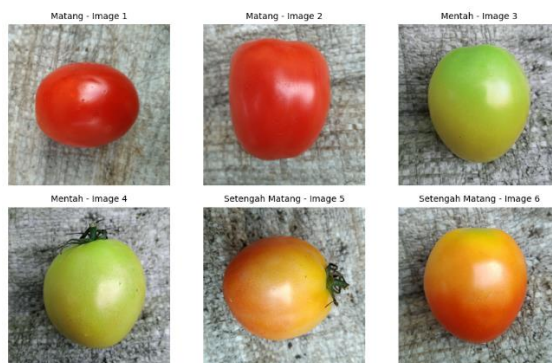
3. HASIL DAN PEMBAHASAN

Studi ini menggunakan 481 dataset citra tomat. Dataset yang dikumpulkan di bagi menjadi tiga class: matang, mentah, dan setengah matang. Pembagian data ditunjukkan dalam Tabel 1. Tabel *class* buah tomat.

Tabel 1. Tabel *class* buah tomat

No	Class	Data
1.	Matang	160
2.	Mentah	161

3.	Setengah Matang	160
Total		481



Gambar 4. Sampel dataset buah tomat

Membagi data mentah menjadi 2 yaitu data uji dan data validasi. Kedua data tersebut dibagi menjadi 70% untuk data uji dan 30% untuk data validasi. Fungsi pembagian data ini bertujuan untuk melatih model yang akan dibuat serta menghitung akurasi dari pelatihan model tersebut. Proses augmentasi data akan melibatkan empat jenis pemrosesan, termasuk *RandomFlip*, *RandomRotation*, *RandomZoom*, dan *Rescaling*. Tujuan dari augmentasi data ini adalah untuk mengurangi risiko *overfitting* dengan melakukan penyesuaian seminimal mungkin pada data asli.

Pada Studi ini menggunakan dua model CNN yaitu simple CNN dan VGG16 dengan dilakukan dua kali percobaan dengan membedakan besaran *epoch* yang digunakan. Berdasarkan hasil pada percobaan pertama, model Simple CNN yang dilatih selama 10 *epoch* menunjukkan performa yang cukup baik pada data validasi dengan akurasi mencapai 93,33%. Namun, terdapat indikasi *overfitting* yang ditandai dengan perbedaan signifikan antara akurasi pada data latih (64,15%) dan data validasi. Nilai loss yang cukup tinggi, baik pada data latih (0,9919) maupun data validasi (0,9097), juga mengindikasikan potensi peningkatan performa model. Untuk mengatasi *overfitting* dan meningkatkan performa, beberapa langkah yang dapat dipertimbangkan antara lain menambah jumlah *epoch*, menggunakan teknik regularisasi, melakukan data augmentasi, memperkaya arsitektur model, dan

melakukan *fine-tuning* terhadap *hyperparameter*. Dengan penyempurnaan lebih lanjut, model *Simple CNN* ini berpotensi untuk menghasilkan performa yang lebih baik dan konsisten pada kedua set data.

Model VGG16 yang dilatih selama 10 *epoch* menunjukkan performa yang sangat baik, dengan akurasi mencapai 92,45% pada data latih dan 97,78% pada data validasi. Nilai *loss* yang rendah, baik pada data latih (0,4733) maupun data validasi (0,0634), mengindikasikan model telah mempelajari pola data dengan sangat baik dan memiliki generalisasi yang luar biasa. Tidak terdapat perbedaan signifikan antara akurasi pada kedua set data, menunjukkan model tidak mengalami *overfitting* yang parah. Dengan performa yang konsisten dan akurasi validasi yang sangat tinggi, model VGG16 ini siap untuk diimplementasikan dalam produksi atau aplikasi nyata. Beberapa peningkatan dapat dilakukan seperti menambah *epoch*, melakukan *fine-tuning hyperparameter*, atau menggunakan data augmentasi, namun secara keseluruhan model ini telah mencapai hasil yang sangat menggembirakan. Tabel 2 Tabel Hasil Akurasi percobaan 1.

Tabel 2. Tabel Hasil Akurasi percobaan 1

Arsitektur	Epoch	Loss	Accuracy	Val_Loss	Val_Accuracy
Simple CNN	10	0,9919	64,15%	0,9097	93,33%
VGG16	10	0.4733	92,45%	0.0634	97,78%

Pada percobaan kedua dengan model *Simple CNN* yang dilatih selama 20 *epoch*, hasilnya menunjukkan peningkatan performa yang signifikan dibandingkan dengan percobaan sebelumnya yang hanya menggunakan 10 *epoch*. Nilai *loss* pada data latih turun menjadi 0,3904, sementara akurasinya meningkat menjadi 87,74%. Ini mengindikasikan bahwa dengan iterasi pelatihan yang lebih banyak, model berhasil mempelajari pola data latih dengan lebih baik. Performa model pada data validasi juga meningkat secara drastis. Nilai *val_loss* turun menjadi 0,3225, sementara *val_accuracy* meningkat hingga mencapai 97,78%. Angka akurasi validasi yang sangat tinggi ini menunjukkan bahwa model memiliki generalisasi yang luar biasa baik pada data yang

belum pernah dilihat sebelumnya, serta menandakan bahwa model tidak mengalami *overfitting* yang signifikan.

Pada percobaan kedua dengan model VGG16 yang dilatih selama 20 *epoch*, hasilnya menunjukkan performa yang sangat baik dan konsisten pada kedua set data. Nilai loss pada data latih turun menjadi 0,1096, sementara akurasi meningkat hingga mencapai 97,17%. Angka akurasi yang sangat tinggi ini mengindikasikan bahwa model telah berhasil mempelajari pola data latih dengan sangat baik. Meskipun demikian, terdapat sedikit penurunan performa pada data validasi dibandingkan dengan percobaan sebelumnya. Nilai *val_loss* sedikit meningkat menjadi 0,3515, sementara *val_accuracy* turun menjadi 95,56%. Meskipun terjadi penurunan, angka akurasi validasi sebesar 95,56% masih sangat tinggi dan mengindikasikan bahwa model memiliki generalisasi yang sangat baik pada data yang belum pernah dilihat sebelumnya. Tabel 3. Tabel Hasil Akurasi percobaan 2.

Tabel 3. Tabel Hasil Akurasi percobaan 2

Arsitektur	Epoch	Loss	Accuracy	Val_Loss	Val_Accuracy
Simple CNN	20	0,3904	87,74%	0,3225	97,78%
VGG16	20	0,1096	97,17%	0,3515	95,56%

Pada percobaan ketiga menggunakan model *Simple* CNN yang dilatih selama 50 *epoch*, performa model menunjukkan hasil yang baik dengan nilai loss pada data latih sebesar 0,0772 dan akurasi 97,33%, serta *val_loss* sebesar 0,1720 dan *val_accuracy* 95,83%, yang mengindikasikan bahwa model dapat mempelajari pola data latih dengan cukup baik dan mampu melakukan generalisasi yang baik pada data baru. Sedangkan pada percobaan kedua menggunakan model VGG16 yang juga dilatih selama 50 *epoch*, hasilnya menunjukkan performa luar biasa pada data latih dengan loss 0,0110 dan akurasi mencapai 100%, namun dengan sedikit penurunan pada data validasi, dengan *val_loss* 0,1299 dan *val_accuracy* 98,61%. Meskipun terdapat sedikit penurunan performa validasi dibandingkan data latih, kedua model menunjukkan

kemampuan yang baik dalam menggeneralisasi data yang belum pernah dilihat sebelumnya, dengan VGG16 menunjukkan hasil yang lebih unggul. Tabel 4. Tabel Hasil Akurasi percobaan 3.

Tabel 4. Tabel Hasil Akurasi percobaan 3

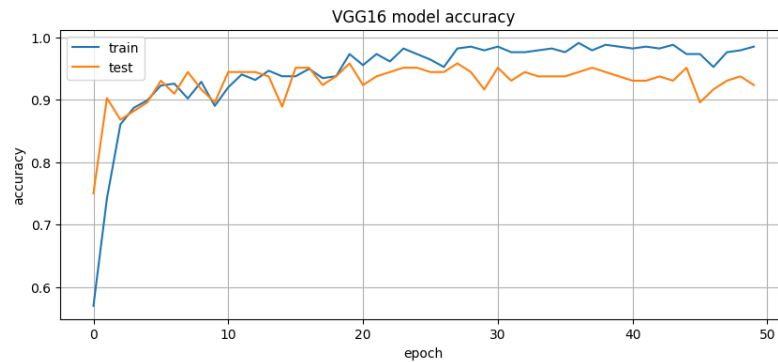
Arsitektur	Epoch	Loss	Accuracy	Val_Loss	Val_Accuracy
Simple	50	0,0110	100,00%	0,1299	98,61%
CNN					
VGG16	50	0,0772	97,33%	0,1720	95,85%

Percobaan Simple CNN menunjukkan performa lebih baik secara keseluruhan, dengan akurasi lebih tinggi dan loss lebih rendah pada data validasi. Namun, ada indikasi kemungkinan *overfitting* pada Percobaan Simple CNN karena akurasi pada data training yang mencapai 100%. Di sisi lain, Percobaan VGG16, meskipun memiliki akurasi dan loss yang sedikit lebih rendah, mungkin lebih ter-generalized dan mampu menangani data baru dengan lebih baik dalam beberapa kasus, sehingga memiliki keunggulan dalam menghadapi variasi data yang belum pernah dilihat sebelumnya.

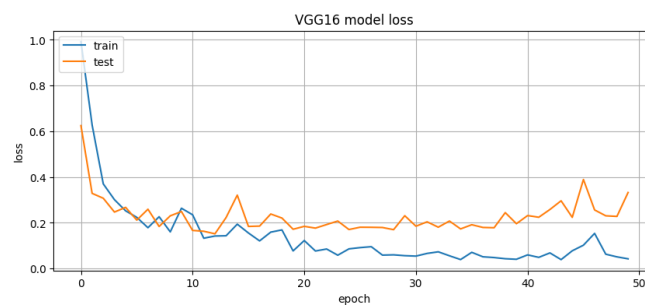
Berdasarkan hasil dari dua percobaan dengan dua model yang berbeda, yaitu Simple CNN dan VGG16, dapat disimpulkan bahwa model VGG16 pada percobaan kedua dengan 20 *epoch* menghasilkan performa terbaik. Pada percobaan ini, model VGG16 mencapai akurasi 97,17% pada data latih dan 95,56% pada data validasi, dengan nilai loss yang rendah, yaitu 0,1096 pada data latih dan 0,3515 pada data validasi. Tabel 4. Hasil Terbaik.

Tabel 5. Hasil Terbaik

Arsitektur	Epoch	Loss	Accuracy	Val_Loss	Val_Accuracy
VGG16	50	0,0772	97,33%	0,1720	95,85%

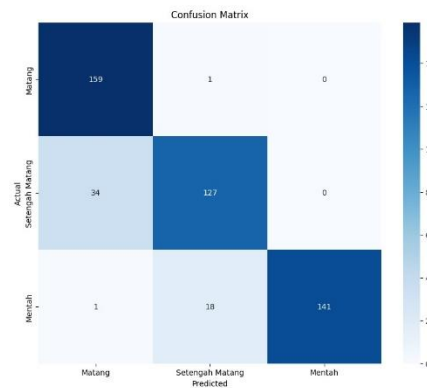


Gambar 5. Grafik perbandingan *accuracy* data train dan test



Gambar 6. Grafik perbandingan *loss* data train dan test

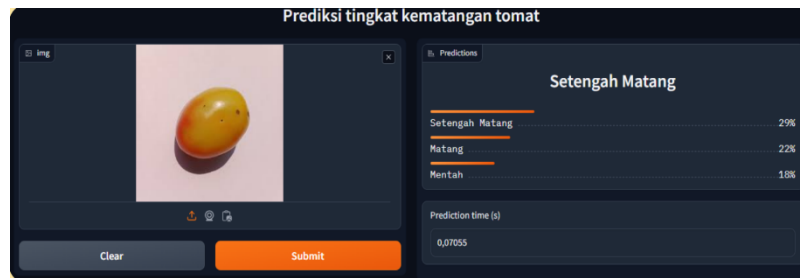
Berdasarkan confusion matrix yang ditampilkan, model menunjukkan performa yang baik dalam memprediksi tingkat kematangan buah tomat. Gambar 11. Confusion matrix.



Gambar 7. Confusion matrix

Pada tiga kelas: Matang, Setengah Matang, dan Mentah. Pada kelas "Matang," model berhasil memprediksi 159 sampel dengan benar, hanya membuat 1 kesalahan dengan salah memprediksi sebagai "Setengah Matang." Di kelas "Setengah Matang,"

terdapat beberapa kesalahan dengan 34 sampel diprediksi sebagai "Matang" dan 18 sebagai "Mentah," namun 127 sampel diprediksi dengan benar. Untuk kelas "Mentah," model berhasil memprediksi 141 sampel dengan benar, hanya membuat 1 kesalahan dengan memprediksi sebagai "Matang" dan 18 sebagai "Setengah Matang." Meskipun ada beberapa kesalahan klasifikasi, model ini secara keseluruhan cukup akurat dalam mengidentifikasi tingkat kematangan tomat.



Gambar 8. Console Hasil Pengujian

4. SIMPULAN

Berdasarkan penelitian ini, model VGG16 terbukti memiliki performa yang lebih baik dalam memprediksi tingkat kematangan buah tomat dibandingkan model Simple CNN. Meskipun Simple CNN menghasilkan hasil yang baik, terutama setelah penambahan *epoch*, model ini masih menunjukkan indikasi *overfitting* yang signifikan. Sebaliknya, model VGG16 menunjukkan generalisasi yang luar biasa, dengan akurasi yang tinggi pada data latih dan validasi. Dengan demikian, VGG16 terbukti lebih andal dan efisien dalam memprediksi tingkat kematangan tomat, menjadikannya pilihan yang lebih optimal untuk diterapkan dalam industri pertanian atau pengolahan tomat. Implementasi lebih lanjut dari model ini dapat membantu meningkatkan akurasi dan konsistensi penilaian kematangan buah, sehingga mampu memberikan solusi yang efisien dan tepat bagi para petani maupun industri terkait.

PUSTAKA

- Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. *2017 International Conference on Engineering and Technology (ICET)*, 1–6. <https://doi.org/10.1109/ICEngTechnol.2017.8308186>
- Athallah Tsany Rakha Dzaky. (2021). Deteksi Penyakit Tanaman Cabai Menggunakan Metode Convolutional Neural Network. *E-Proceeding of Engineering*, 8(2), 3039–3055.

- Biringkanae, P., & Raza Bunahri, R. (2023). Literature Review Penggunaan Teknologi Kecerdasan Buatan dalam Penerbangan: Analisis Perkembangan Teknologi, Potensi Keamanan, dan Tantangan. *JURNAL ILMU MANAJEMEN TERAPAN (JIMT)*, 4(5), 745–752.
- Dwi Suci Anggraeni, Arif Widayana, Putri Dwi Rahayu, & Chaerur Rozikin. (2022). METODE ALGORITMA CONVOLUTIONAL NEURAL NETWORK PADA KLASIFIKASI PENYAKIT TANAMAN CABAI . *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, 7, 73–78.
- Fathurrahman, I., Muliawan Nur, A., & Fathurrahman, F. (2019). Identifikasi Kematangan Buah Mentimun Berbasis Citra Digital Menggunakan Jaringan Syaraf Tiruan Backpropagation. *Infotek : Jurnal Informatika Dan Teknologi*, 2(1), 27–33. <https://doi.org/10.29408/jit.v2i1.976>
- Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De Debashis. (2019). *Fundamental Concepts of Convolutional Neural Network* (Vol. 172).
- Helsaputra, A., Luhur Prasasti, A., & Rendian Septiawan, R. (2021). IMPLEMENTASI DEEP LEARNING UNTUK PREDIKSI TINGKAT KEMATANGAN DAN BOBOT BUAH PEPAYA. *Library Telkom University*, 8(6), 11993–11998.
- Ishak, I., Amal, I., Muhammad, M., & Kaswar, A. B. (2022). Sistem Pendeteksi Kematangan Buah Tomat Berbasis Pengolahan Citra Digital Menggunakan Metode Jaringan Syaraf Tiruan. *Jurnal Media TIK*, 5(1), 65–69.
- Palupiningsih, P., Sujiwanto, A. R., & Prawirodirjo, R. R. B. P. (2023). Analisis Perbandingan Performa Model Klasifikasi Kesehatan Daun Tomat menggunakan arsitektur VGG, MobileNet, dan Inception V3. *Jurnal Ilmu Komputer Dan Agri-Informatika*, 10(1), 98–110. <https://doi.org/10.29244/jika.10.1.98-110>
- Pouyanfar, S., Tao, Y., Mohan, A., Tian, H., Kaseb, A. S., Gauen, K., Dailey, R., Aghajanzadeh, S., Lu, Y.-H., Chen, S.-C., & Shyu, M.-L. (2018). Dynamic Sampling in Convolutional Neural Networks for Imbalanced Data Classification. *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 112–117. <https://doi.org/10.1109/MIPR.2018.00027>
- Putri Ananda, T., Viola Widyasari, S., Ihsan Muttaqin, M., & Stefanie, A. (2023). IDENTIFIKASI TINGKAT KEMATANGAN BUAH PEPAYA MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN). *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(3), 2094–2097. <https://doi.org/10.36040/jati.v7i3.7137>
- Saxena, A. (2022). An Introduction to Convolutional Neural Networks. *International Journal for Research in Applied Science and Engineering Technology*, 10(12), 943–947. <https://doi.org/10.22214/ijraset.2022.47789>
- Sigitta, R. C., Saputra, R. H., & Fathulloh, F. (2023). Deteksi Penyakit Tomat melalui Citra Daun menggunakan Metode Convolutional Neural Network. *AVITEC*, 5(1), 43. <https://doi.org/10.28989/avitec.v5i1.1404>
- Soekarta, R., Nurdjan, N., & Syah, A. (2023). Klasifikasi Penyakit Tanaman Tomat Menggunakan Metode Convolutional Neural Network (CNN). *Insect (Informatics and Security): Jurnal Teknik Informatika*, 8(2), 143–151. <https://doi.org/10.33506/insect.v8i2.2356>
- Vicky Purnama Putra, J., Ayu, F., & Julianto, B. (2023). Implementasi Pendeteksi Penyakit pada Daun Alpukat Menggunakan Metode CNN. *STAINS (SEMINAR NASIONAL TEKNOLOGI & SAINS)*, 2(1), 155–162.
- Weny Indah Kusumawati, & Adisaputra Zidha Noorizki. (2023). Perbandingan Performa Algoritma VGG16 Dan VGG19 Melalui Metode CNN Untuk Klasifikasi Varietas Beras. *Journal of Computer, Electronic, and Telecommunication*, 4(2). <https://doi.org/10.52435/complete.v4i2.387>