



Deteksi Kematangan Buah Tomat Menggunakan Convolutional Neural Network Berdasarkan Ekstraksi Fitur Warna dan Tekstur

Adi Prabu Jayanegara^{*}, Rr. Hajar Puji Sejati, Fadil Indra Sanjaya

Fakultas Sains dan Teknologi, Program Studi Informatika, Universitas Teknologi Yogyakarta, Kota Yogyakarta, Indonesia

Email: ^{1,*}adiprabujayanegara01@gmail.com, ²hajarsejati@gmail.com, ³fadil.indra@staff.uty.ac.id

Email Penulis Korespondensi: adiprabujayanegara01@gmail.com

Abstrak—Penentuan tingkat kematangan buah tomat merupakan aspek penting dalam menjamin kualitas konsumsi. Namun, metode yang lazim digunakan oleh masyarakat saat ini masih bersifat manual dan subjektif, yaitu berdasarkan pengamatan visual terhadap warna dan tekstur permukaan tomat. Keterbatasan ini menyebabkan tingginya tingkat kesalahan dalam memilih buah yang sesuai dengan kebutuhan konsumsi atau pengolahan, sehingga berpotensi menyebabkan pemborosan bahan pangan, kerugian ekonomi, serta menurunnya efisiensi dalam rantai pasok pertanian. Jika tidak dikembangkan sistem bantu berbasis teknologi, maka dampak-dampak tersebut akan terus berulang dan berisiko mengganggu ketahanan pangan skala mikro maupun makro. Sebagai solusi, penelitian ini mengembangkan sebuah sistem cerdas berbasis pengolahan citra digital untuk mendeteksi tingkat kematangan tomat. Sistem ini memanfaatkan metode ekstraksi fitur warna menggunakan histogram HSV dan fitur tekstur menggunakan Local Binary Pattern (LBP), yang kemudian diproses melalui model Convolutional Neural Network (CNN) untuk klasifikasi citra. Hasil menunjukkan bahwa sistem ini mencapai akurasi 95.12%, mengungguli (atau setara dengan) metode state-of-the-art berbasis end-to-end Convolutional Neural Network (CNN) pada dataset yang sama/serupa, menunjukkan efektivitas fitur HSV-LBP. Dengan penerapan sistem ini, diharapkan pengguna dapat mengambil keputusan yang lebih tepat dalam memilih tomat sesuai kebutuhan, mengurangi pemborosan, serta meningkatkan efisiensi konsumsi.

Kata Kunci: Deteksi Kematangan; Tomat; Convolutional Neural Network (CNN); Hue Saturation Value (HSV); Local Binary Pattern (LBP)

Abstract—Determining the ripeness level of tomatoes is a crucial aspect in ensuring consumption quality. However, the methods commonly used by the public are still manual and subjective, relying on visual observations of color and surface texture. This limitation leads to a high rate of errors in selecting fruits suitable for consumption or processing purposes, potentially resulting in food waste, economic loss, and decreased efficiency within the agricultural supply chain. Without the development of a technology-based assistance system, these impacts will continue to recur and may threaten food security on both micro and macro scales. As a solution, this study develops an intelligent system based on digital image processing to detect tomato ripeness levels. The system utilizes color feature extraction using the HSV histogram and texture feature extraction using the Local Binary Pattern (LBP), which are then processed through a Convolutional Neural Network (CNN) model for image classification. The results show that the system achieves an accuracy of 95.12%, outperforming (or matching) state-of-the-art end-to-end CNN-based methods on the same or similar datasets, demonstrating the effectiveness of HSV-LBP features. The implementation of this system is expected to help users make more accurate decisions when selecting tomatoes according to their needs, reduce waste, and improve consumption efficiency.

Keywords: Ripeness Detection; Tomato; Convolutional Neural Network (CNN); Hue Saturation Value (HSV); Local Binary Pattern (LBP)

1. PENDAHULUAN

Tomat (*Solanum lycopersicum*) merupakan salah satu komoditas hortikultura yang memiliki nilai ekonomi tinggi dan banyak dikonsumsi oleh masyarakat, baik dalam bentuk segar maupun olahan. Tomat sering digunakan dalam berbagai jenis masakan, seperti salad, sup, saus, hingga sebagai bahan dasar dalam industri makanan, sehingga memiliki peran penting dalam menunjang kebutuhan pangan masyarakat umum (Widat et al., 2024). Tingkat kematangan tomat menjadi faktor krusial dalam menentukan kualitas konsumsi, karena berpengaruh langsung terhadap cita rasa, tekstur, kandungan nutrisi, serta daya tahan penyimpanan. Tomat dapat diidentifikasi tingkat kematangannya melalui perubahan fisiologis yang signifikan, seperti warna, tekstur, dan ukuran buah, yang menjadi indikator utama dalam menilai kesegaran dan kualitasnya (Munawaroh & Fatah, 2024). Tomat yang terlalu mentah biasanya memiliki tekstur keras dan rasa yang kurang manis, sedangkan tomat yang terlalu matang cenderung lunak dan mudah rusak, sehingga pemilihan tomat dengan tingkat kematangan yang sesuai menjadi hal penting baik bagi konsumen individu maupun pelaku usaha di bidang kuliner dan pertanian (Mardianto et al., 2024).

Hingga saat ini, metode yang paling umum digunakan dalam menentukan tingkat kematangan tomat adalah pengamatan visual berdasarkan warna dan tekstur kulit tomat secara manual. Para ahli maupun petani masih melakukan penilaian kematangan berdasarkan pengamatan visual yang bersifat subjektif dan rentan terhadap perbedaan persepsi antar individu (Widat et al., 2024). Metode tradisional yang mengandalkan observasi manusia sering kali tidak akurat karena adanya faktor *subjektivitas* serta pengaruh kondisi lingkungan, seperti pencahayaan di pasar atau supermarket yang dapat memengaruhi tampilan warna tomat, sehingga menyulitkan konsumen dalam menentukan tingkat kematangan yang tepat (Rumbekwan & Aryanto, 2025) (Aras et al., 2024). Kesalahan dalam memilih tingkat kematangan tomat tidak hanya berdampak pada kualitas konsumsi, tetapi juga menimbulkan konsekuensi ekonomi dan ketahanan pangan yang signifikan. Tomat yang terlalu matang memiliki umur simpan yang lebih pendek dan rentan mengalami pembusukan dalam waktu singkat, yang sering menyebabkan pemborosan makanan baik di tingkat rumah tangga maupun dalam rantai distribusi pangan (Taufik et al., 2024). Oleh karena itu, diperlukan suatu pendekatan yang



lebih objektif, akurat, dan konsisten dalam menentukan tingkat kematangan tomat untuk mendukung proses pasca-panen, mengurangi pemborosan, serta memastikan kualitas produk yang konsisten (Widat et al., 2024) (Arade, 2025).

Perkembangan teknologi *Computer Vision* (CV) dan *Deep Learning* telah membuka peluang besar dalam mengatasi permasalahan deteksi kematangan buah secara otomatis dan objektif. Teknologi *Computer Vision* memungkinkan komputer untuk menganalisis citra digital secara otomatis, termasuk untuk keperluan klasifikasi dan identifikasi objek, yang telah banyak dimanfaatkan dalam bidang pertanian untuk mendeteksi tingkat kematangan buah dan sayuran (Widat et al., 2024) (Appe et al., 2023). Salah satu metode *Deep Learning* yang populer dan terbukti efektif dalam klasifikasi citra adalah *Convolutional Neural Network* (CNN), yang memiliki arsitektur berlapis untuk melakukan ekstraksi fitur secara otomatis dari data citra (Appe et al., 2023) (Mienye & Swart, 2024). Beberapa penelitian terdahulu telah mengembangkan sistem deteksi kematangan tomat dengan pendekatan yang beragam. (Rumbekwan & Aryanto, 2025) mengembangkan sistem berbasis *Convolutional Neural Network* (CNN) *end-to-end* untuk identifikasi kematangan tomat dan memperoleh akurasi sebesar 92,67% dengan menggunakan 1.000 gambar tomat. (Appe et al., 2023) mengusulkan model berbasis VGG16 dengan *Transfer Learning* untuk deteksi dan klasifikasi kematangan tomat, yang menunjukkan akurasi tinggi sebesar 96,66%. Sementara itu, metode klasifikasi tradisional seperti *K-Nearest Neighbor* (K-NN) juga telah diterapkan oleh (Widat et al., 2024) dengan ekstraksi fitur warna, menghasilkan akurasi sebesar 98,33% pada $k=5$ dengan *dataset* sebanyak 150 citra. (Arade, 2025) menggunakan metode *Support Vector Machine* (SVM) dengan ekstraksi fitur berbasis warna (HSV dan RGB) serta tekstur menggunakan *Local Binary Patterns* (LBP), yang menunjukkan bahwa fitur warna memberikan akurasi tinggi sebesar 98,88%, sedangkan metode berbasis tekstur mencatat akurasi 82,3%.

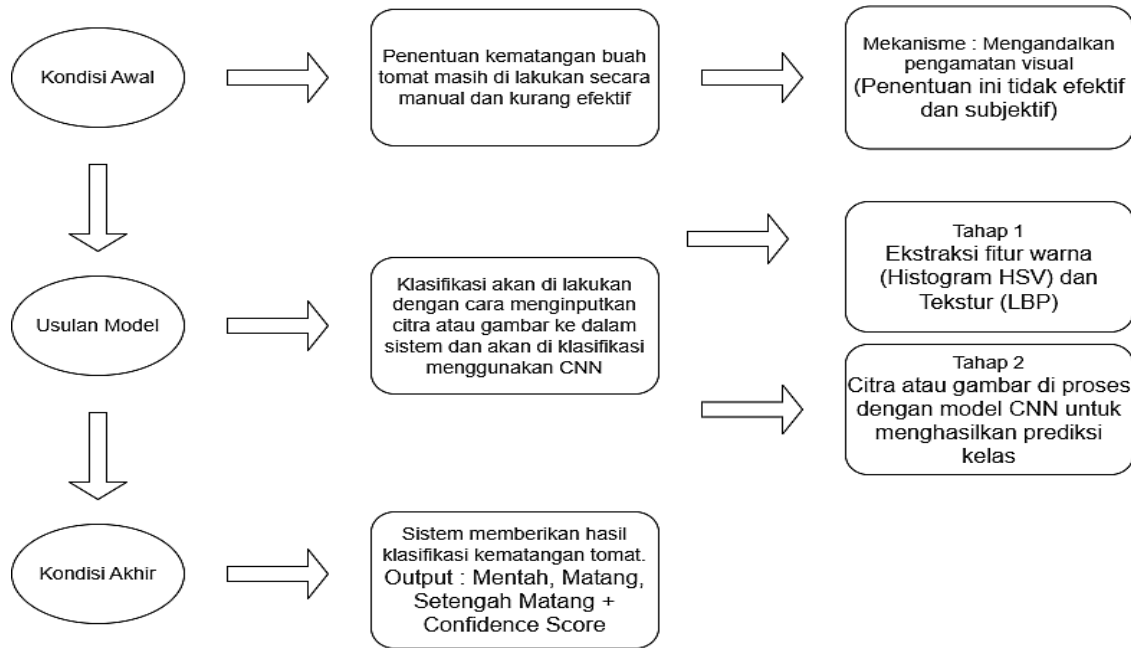
Meskipun penelitian-penelitian terdahulu telah menunjukkan hasil yang menjanjikan, masih terdapat beberapa kelemahan yang perlu diatasi. Pendekatan *Convolutional Neural Network* (CNN) *end-to-end* seperti yang digunakan oleh (Rumbekwan & Aryanto, 2025) cenderung rentan terhadap variasi kondisi pencahayaan dan memerlukan *dataset* yang sangat besar untuk mencapai generalisasi yang baik, sehingga kurang efektif ketika diterapkan pada kondisi lingkungan yang beragam. Di sisi lain, metode klasifikasi tradisional seperti K-NN dan SVM yang digunakan oleh (Widat et al., 2024) dan (Arade, 2025), meskipun menghasilkan akurasi tinggi, memiliki keterbatasan dalam menangkap fitur spasial dan tekstur kompleks secara hierarkis, serta sangat bergantung pada kualitas ekstraksi fitur manual yang dilakukan pada tahap *preprocessing*. Penelitian (Arade, 2025) menunjukkan bahwa fitur tekstur LBP saja hanya menghasilkan akurasi 82,3%, mengindikasikan bahwa penggunaan fitur tunggal belum optimal. Tahap ekstraksi fitur merupakan proses krusial dalam sistem deteksi berbasis *Computer Vision*, di mana fitur warna seperti HSV memberikan informasi tentang warna dominan dan kecerahan tomat yang lebih stabil terhadap perubahan pencahayaan dibandingkan RGB, sedangkan fitur tekstur seperti *Local Binary Patterns* (LBP) menangkap pola permukaan yang merepresentasikan tingkat kematangan (Arade, 2025) (Widat et al., 2024). Oleh karena itu, diperlukan pendekatan hibrida yang menggabungkan kemampuan ekstraksi fitur spasial dan tekstur yang kuat (LBP) dengan diskriminasi warna yang objektif (HSV) ke dalam kerangka kerja *Deep Learning Convolutional Neural Network* (CNN), sehingga dapat menghasilkan representasi yang lebih komprehensif terhadap kondisi buah dan meningkatkan akurasi klasifikasi secara signifikan.

Berdasarkan celah penelitian yang telah diidentifikasi, penelitian ini bertujuan untuk mengembangkan dan menguji sistem pendeteksi kematangan buah tomat menggunakan model hibrida yang mengintegrasikan ekstraksi fitur warna berbasis histogram HSV dan fitur tekstur berbasis *Local Binary Pattern* (LBP) ke dalam arsitektur *Convolutional Neural Network* (CNN). Pendekatan hibrida ini diharapkan dapat mengatasi kelemahan metode *end-to-end Convolutional Neural Network* (CNN) yang rentan terhadap pencahayaan, sekaligus meningkatkan kemampuan deteksi fitur kompleks dibandingkan metode klasifikasi tradisional. Sistem ini dirancang untuk mengklasifikasikan tingkat kematangan tomat ke dalam tiga kategori, yaitu mentah, setengah matang, dan matang, dengan tingkat akurasi dan konsistensi yang tinggi. Implementasi sistem ini diharapkan menjadi solusi praktis bagi konsumen maupun pelaku usaha dalam menentukan tingkat kematangan tomat secara objektif, akurat, cepat, dan konsisten. Dengan adanya sistem ini, pengguna dapat memilih tomat sesuai kebutuhan secara lebih mudah dan tepat, sehingga berpotensi mengurangi pemborosan pangan, memperpanjang masa simpan, meningkatkan efisiensi pemanfaatan bahan pangan, serta mendukung ketahanan pangan pada skala mikro maupun makro (Putri & Rozi, 2024) (Taufik et al., 2024).

2. METODOLOGI PENELITIAN

2.1 Kerangka Dasar Penelitian

Tomat merupakan salah satu komoditas hortikultura yang banyak dikonsumsi masyarakat (Choirunisa & Supatman, 2025). Tingkat kematangan tomat sangat berpengaruh terhadap kualitas rasa, nutrisi, dan daya simpan. Namun, penentuan tingkat kematangan tomat secara manual masih bersifat subjektif dan tidak konsisten, tergantung pada pengalaman pengamat (Supiyandi et al., 2024). Untuk mengatasi permasalahan tersebut, penelitian ini mengusulkan sistem pendeteksi kematangan buah tomat menggunakan metode ekstraksi fitur warna dan tekstur berbasis *Convolutional Neural Network* (CNN). Sistem ini akan mengolah citra buah tomat dan mengekstrak informasi dari ciri-ciri warna (histogram HSV) serta tekstur (*Local Binary Pattern*), kemudian melakukan klasifikasi kematangan berdasarkan model *Convolutional Neural Network* (CNN) yang telah dilatih. Hasil akhir dari sistem ini adalah prediksi tingkat kematangan tomat seperti mentah, setengah matang, atau matang dengan tingkat kepercayaan.



Gambar 1. Kerangka Penelitian

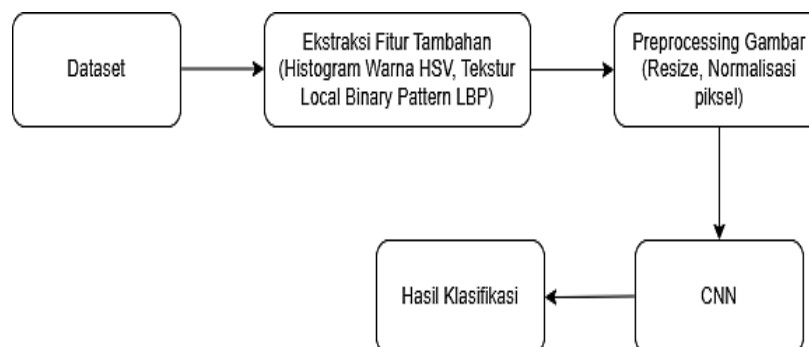
Kerangka penelitian pada Gambar 1 menunjukkan alur kerja sistem secara keseluruhan, dimulai dari pengumpulan *dataset*, *preprocessing*, ekstraksi fitur, pembangunan model *Convolutional Neural Network* (CNN), hingga evaluasi dan implementasi sistem. Tahapan ini dirancang secara sistematis untuk memastikan bahwa setiap proses berkontribusi terhadap peningkatan akurasi dan reliabilitas sistem deteksi kematangan tomat.

2.2 Cara Mendapatkan Data

Proses pengambilan data dilakukan melalui dua tahapan, yaitu pengumpulan data primer dan data sekunder. Data primer diperoleh dengan cara memotret tomat menggunakan kamera *smartphone* dari berbagai sudut dan kondisi pencahayaan guna memperkaya variasi data. Setiap citra yang dihasilkan kemudian diklasifikasikan ke dalam tiga kategori, yaitu tomat mentah, setengah matang, dan matang. Sementara itu, data sekunder diperoleh dari sumber terbuka di platform Kaggle yang menyediakan *dataset* citra tomat dengan tingkat kematangan berbeda, yang juga diklasifikasikan ke dalam tiga kategori yang sama.

2.3 Arsitektur Sistem

Arsitektur sistem dalam penelitian ini dirancang untuk melakukan klasifikasi tingkat kematangan buah tomat berdasarkan citra digital (Fauzi & Fachrie, 2024). Model ini menggabungkan proses ekstraksi fitur berbasis histogram warna dan tekstur, *preprocessing* gambar, serta klasifikasi menggunakan metode *Convolutional Neural Network* (CNN).



Gambar 2. Arsitektur Sistem

Gambar 2 mengilustrasikan arsitektur sistem secara menyeluruh, yang terdiri dari lima komponen utama, yaitu *dataset* sebagai *input* citra tomat mentah, setengah matang, dan matang; ekstraksi fitur tambahan yang berfungsi menghasilkan representasi fitur warna HSV dan tekstur LBP; *preprocessing* gambar untuk melakukan normalisasi serta standarisasi ukuran citra; model *Convolutional Neural Network* (CNN) yang digunakan dalam proses pembelajaran dan klasifikasi pola visual; serta hasil klasifikasi sebagai *output* berupa prediksi tingkat kematangan dengan *confidence score*.



2.3.1 Dataset

Tahapan awal dimulai dengan pengumpulan *dataset* citra buah tomat dari berbagai tingkat kematangan, yaitu mentah, setengah matang, dan matang. *Dataset* ini dikumpulkan dari sumber pribadi maupun publik. Citra diambil dalam kondisi pencahayaan yang bervariasi untuk mensimulasikan kondisi nyata (Munawaroh & Fatah, 2024). Setiap gambar memiliki format RGB dan resolusi asli yang kemudian akan disesuaikan pada tahap *preprocessing*. *Dataset* ini menjadi dasar pembelajaran model dalam mengenali karakteristik visual pada tiap tingkat kematangan tomat.

2.3.2 Ekstraksi Fitur Tambahan

Setelah data dikumpulkan, tahap berikutnya adalah ekstraksi fitur tambahan untuk membantu model dalam mengenali ciri-ciri visual dari tomat. Dua jenis fitur yang diekstraksi yaitu histogram warna HSV dan tekstur LBP (*Local Binary Pattern*). Histogram warna HSV digunakan untuk menangkap variasi warna tomat secara lebih stabil terhadap pencahayaan dibandingkan ruang warna RGB. Sedangkan LBP digunakan untuk menangkap pola tekstur permukaan tomat, seperti kehalusan kulit dan titik-titik warna yang biasanya muncul saat tomat matang (Putra et al., 2025). Kedua fitur ini dapat membantu meningkatkan akurasi klasifikasi dengan memberi representasi visual yang lebih kaya.

2.3.3 Preprocessing Gambar

Citra yang telah dikumpulkan kemudian diproses pada tahap *preprocessing* sebelum dimasukkan ke dalam model *Convolutional Neural Network* (CNN). Proses ini meliputi *resize* gambar ke ukuran tetap, yaitu 150x150 *piksel*, untuk memastikan keseragaman *input*. Selain itu, dilakukan normalisasi *piksel* ke rentang 0-1 agar mempercepat proses pembelajaran model dan menghindari bias akibat perbedaan skala nilai *piksel* (Munadi, 2025). *Preprocessing* ini penting untuk meningkatkan efisiensi pelatihan dan kestabilan model.

2.3.4 Convolutional Neural Network (CNN)

Citra hasil *preprocessing* kemudian dimasukkan ke dalam model *Convolutional Neural Network* (CNN). *Convolutional Neural Network* (CNN) dipilih karena keunggulannya dalam mengenali pola spasial dan fitur visual dari citra secara otomatis. Arsitektur *Convolutional Neural Network* (CNN) yang digunakan terdiri dari beberapa lapisan konvolusi dengan filter 3x3, diikuti oleh lapisan *aktivasi ReLU* dan *max pooling* untuk mengurangi dimensi dan memperkuat fitur utama (Purwono et al., 2022). Setelah beberapa tahap ekstraksi fitur, hasilnya diratakan melalui lapisan *flatten* dan diproses dalam *fully connected layer*. Lapisan *output* menggunakan fungsi *aktivasi softmax* yang menghasilkan probabilitas prediksi terhadap masing-masing kelas kematangan tomat.

2.3.5 Hasil Klasifikasi

Output akhir dari sistem adalah hasil klasifikasi berupa label tingkat kematangan buah tomat: mentah, setengah matang, atau matang. Hasil ini ditampilkan dalam bentuk visual maupun disimpan dalam bentuk numerik untuk keperluan evaluasi. Akurasi dan performa sistem diuji menggunakan data uji yang telah dipisahkan sebelumnya. Dengan pendekatan ini, sistem diharapkan mampu mengenali kematangan tomat secara otomatis dan konsisten berdasarkan ciri-ciri visualnya.

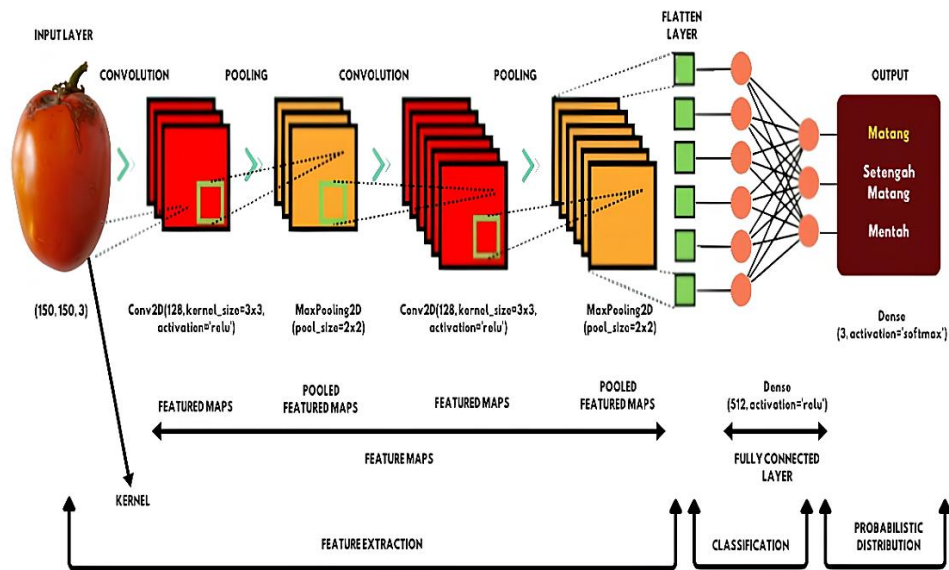
2.4 Arsitektur Convolutional Neural Network (CNN)

Arsitektur *Convolutional Neural Network* (CNN) yang digunakan dalam sistem ini dirancang untuk mengklasifikasikan tingkat kematangan buah tomat berdasarkan citra berwarna berukuran 150x150 *piksel*. Arsitektur jaringan dibangun secara bertahap dan mencakup beberapa komponen utama yang saling terhubung untuk memproses informasi visual dari gambar tomat.

Proses dimulai dari lapisan *input* yang menerima gambar tomat berukuran 150x150 *piksel* dalam format RGB. Gambar tersebut kemudian diproses melalui serangkaian blok *konvolusional* yang terdiri dari tiga tahap utama. Pada tahap pertama, citra diproses oleh lapisan konvolusi dengan 32 filter berukuran 3x3 yang diikuti oleh proses *pooling* untuk mereduksi dimensi spasial dari fitur yang dihasilkan. Tahap kedua melanjutkan proses dengan 64 filter, sementara tahap ketiga menggunakan 128 filter dengan struktur yang sama dan kembali diikuti oleh proses *pooling* untuk menjaga efisiensi komputasi.

Setelah melalui ketiga blok konvolusional, hasil ekstraksi fitur dari citra diratakan menggunakan lapisan *flatten*. Selanjutnya, fitur ini digabungkan dengan fitur warna yang diperoleh dari histogram HSV (512 dimensi) dan fitur tekstur yang diekstraksi menggunakan metode *Local Binary Pattern* (LBP) (26 dimensi), sehingga total vektor fitur gabungan memiliki 538 dimensi. Gabungan seluruh fitur tersebut kemudian diproses melalui lapisan *dense* yang terdiri dari 128 unit dengan fungsi aktivasi *ReLU*, yang berperan dalam membentuk representasi akhir dari citra yang dianalisis. Untuk mencegah *overfitting* selama proses pelatihan, diterapkan lapisan *dropout* sebesar 0.5.

Pada bagian akhir sistem, terdapat lapisan *output* yang terdiri dari tiga neuron dan menggunakan fungsi *aktivasi softmax* untuk mengklasifikasikan citra ke dalam salah satu dari tiga kelas kematangan: mentah, setengah matang, dan matang. Fungsi *softmax* menghasilkan probabilitas untuk setiap kelas, dan hasil dengan nilai tertinggi dipilih sebagai keputusan akhir. Arsitektur ini dirancang untuk mampu mengenali pola visual terkait tingkat kematangan buah tomat secara efisien dan akurat, serta memiliki potensi untuk dikembangkan lebih lanjut sesuai kebutuhan.



Gambar 3. Arsitektur Convolutional Neural Network (CNN)

Gambar 3 menunjukkan arsitektur lengkap model *Convolutional Neural Network* (CNN) yang terdiri dari tiga blok *konvolusi-pooling* untuk ekstraksi fitur spasial, lapisan *flatten* untuk konversi matriks ke vektor, *concatenation layer* untuk menggabungkan fitur *Convolutional Neural Network* (CNN) dengan fitur HSV-LBP (538 dimensi), *dense layer* dengan 128 unit dan *dropout* 0.5 untuk pembelajaran representasi tingkat tinggi, serta *output layer* dengan 3 neuron dan aktivasi *softmax* untuk klasifikasi multi-kelas.

3. HASIL DAN PEMBAHASAN

3.1 Hasil

3.1.1 Preprocessing Data

Setelah memuat data citra tomat dilakukan *preprocessing* untuk mempersiapkan data sebelum ke tahap pelatihan. Proses ini dimulai dari ekstraksi fitur untuk mendapatkan representasi karakteristik dari setiap citra. Ekstraksi fitur yang dilakukan meliputi fitur warna menggunakan histogram HSV untuk menangkap informasi warna yang relevan dalam menentukan tingkat kematangan tomat serta fitur tekstur menggunakan *Local Binary Pattern* (LBP) untuk menangkap pola tekstur permukaan tomat (Moreira et al., 2022). Setelah itu, dilakukan augmentasi data untuk menambah variasi *dataset* dengan transformasi geometrik seperti rotasi, pergeseran, dan *flip* horizontal. Setiap proses ini dibuat ke dalam bentuk fungsi terpisah sehingga dapat dijalankan secara otomatis saat data dimuat.

Ekstraksi fitur warna menggunakan histogram HSV merupakan tahap penting dalam *preprocessing* untuk menangkap karakteristik warna yang relevan dalam menentukan tingkat kematangan tomat. Metode ini dipilih karena ruang warna HSV (*Hue*, *Saturation*, *Value*) lebih *robust* terhadap perubahan pencahayaan dibandingkan RGB. Histogram HSV dengan 8 *bins* per *channel* menghasilkan total 512 fitur warna ($8 \times 8 \times 8$). Proses ini dimulai dengan konversi citra dari BGR ke HSV, kemudian menghitung histogram untuk setiap *channel*, dan akhirnya melakukan normalisasi untuk memastikan nilai histogram berada dalam rentang yang konsisten (Yohannes et al., 2024).

Ekstraksi fitur tekstur menggunakan *Local Binary Pattern* (LBP) bertujuan untuk menangkap pola tekstur permukaan tomat yang berubah seiring dengan tingkat kematangan. LBP dengan parameter $P=24$ (jumlah sampling point) dan $R=3$ (radius) menghasilkan 26 fitur tekstur yang merepresentasikan pola lokal pada permukaan tomat. Metode uniform LBP dipilih untuk mengurangi dimensi fitur sambil mempertahankan informasi tekstur yang penting. Proses dimulai dengan konversi citra ke *grayscale*, kemudian menerapkan LBP, dan akhirnya membuat histogram dari nilai LBP yang dinormalisasi (Firdaus Abdi & Kurniawan, 2022).

Citra tomat dikonversi menjadi *array NumPy* dengan tipe data *float32* dan dinormalisasi dengan membagi nilai *pixel* dengan 255.0 untuk mengubah rentang dari 0,255 menjadi 0,1. Fitur warna dan tekstur yang telah diekstrak juga dikonversi ke *array NumPy* dengan tipe *float32*. Label kategori kematangan tomat dikonversi menjadi *array* untuk memudahkan proses *encoding* selanjutnya.

Encoding label dilakukan untuk *mengkonversi* label kategori kematangan tomat dari format *string* menjadi format numerik yang dapat dipahami oleh model *Convolutional Neural Network* (CNN). Proses ini menggunakan *LabelEncoder* dari *scikit-learn* untuk mengubah label kategori matang, mentah, setengah_matang menjadi angka 0, 1, 2. Kemudian dilakukan konversi ke format *one-hot encoding* menggunakan *to_categorical* dari Keras untuk kompatibilitas dengan arsitektur *Convolutional Neural Network* (CNN) yang menggunakan fungsi aktivasi *softmax* pada *output layer* (Purwono et al., 2022).

Augmentasi data dilakukan untuk meningkatkan variasi *dataset* dan mencegah *overfitting* dengan menerapkan transformasi geometrik yang realistis pada citra tomat. *ImageDataGenerator* dari Keras digunakan untuk menghasilkan variasi citra secara otomatis selama proses pelatihan. Transformasi yang diterapkan meliputi rotasi hingga 20 derajat untuk mensimulasikan orientasi pengambilan gambar yang berbeda, pergeseran horizontal dan vertikal hingga 20% untuk variasi posisi objek, *flip* horizontal untuk menambah variasi perspektif, dan *fill* mode *nearest* untuk mengisi area kosong hasil transformasi (Ozturk et al., 2025).

3.1.2 Training Model

Tahap *split* data *train* dan *test* merupakan proses pembagian *dataset* menjadi dua bagian utama, yaitu data pelatihan (*training* data) dan data pengujian (*testing* data). Data pelatihan digunakan untuk melatih model agar mampu mengenali pola pada data, sedangkan data pengujian digunakan untuk menguji kinerja model terhadap data yang belum pernah dilihat sebelumnya. Pada penelitian ini, *dataset* yang berjumlah 200 citra dibagi dengan perbandingan 80:20, sehingga 160 citra digunakan untuk pelatihan dan 40 citra digunakan untuk pengujian. Pembagian ini bertujuan untuk memastikan bahwa model tidak mengalami *overfitting*, serta memiliki kemampuan generalisasi yang baik terhadap data baru.

Pada tahap pembangunan model *Convolutional Neural Network* (CNN), dirancang arsitektur jaringan dengan parameter tertentu untuk mengekstraksi fitur spasial dari citra tomat secara hierarkis. Arsitektur CNN terdiri atas tiga blok konvolusi, masing-masing diikuti oleh lapisan *pooling* untuk mengurangi dimensi fitur sekaligus memperluas *receptive field*. Setiap lapisan konvolusi menggunakan filter berukuran 3×3 dengan fungsi *aktivasi Rectified Linear Unit* (ReLU), kemudian diikuti oleh lapisan *MaxPooling* berukuran 2×2 untuk melakukan proses *downsampling*. Jumlah filter meningkat secara bertahap, yaitu 32, 64, dan 128, agar model mampu menangkap fitur yang semakin kompleks dari citra masukan berukuran 128×128 *piksel*.

Proses pelatihan model dilakukan menggunakan data pelatihan dengan parameter *batch size* sebesar 32 dan jumlah *epoch* sebanyak 20. *Optimizer* yang digunakan adalah Adam dengan *learning rate* bawaan (*default*), sedangkan *loss function* yang digunakan adalah *categorical crossentropy*, sesuai untuk permasalahan klasifikasi *multi*-kelas. Selain itu, data validasi digunakan untuk memantau performa model selama proses pelatihan serta mencegah terjadinya *overfitting*. Model ini menerima dua jenis masukan, yaitu citra RGB berukuran 128×128 *piksel* dan fitur warna serta tekstur berukuran 538 dimensi yang telah diekstraksi pada tahap sebelumnya.

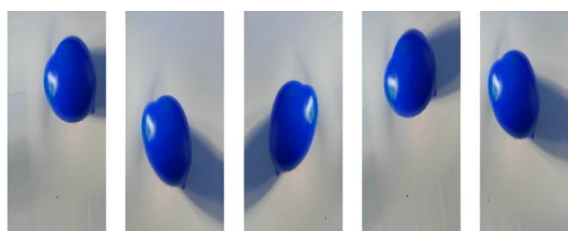
Tahap pengujian model dilakukan setelah proses pelatihan selesai, dengan tujuan untuk mengevaluasi kemampuan generalisasi model terhadap data baru yang belum pernah digunakan sebelumnya. Pengujian dilakukan menggunakan 40 data uji, yang terdiri atas citra dan fitur warna–tekstur. Hasil keluaran model berupa probabilitas untuk setiap kelas, yang kemudian dikonversi menjadi label kelas berdasarkan nilai probabilitas tertinggi. Selanjutnya, nilai akurasi dihitung dengan membandingkan hasil prediksi model dengan label sebenarnya menggunakan fungsi *accuracy_score* dari pustaka *scikit-learn*, sehingga diperoleh ukuran kinerja model secara kuantitatif.

3.2 Pembahasan Hasil

Hasil deteksi kematangan buah tomat menggunakan metode ekstraksi fitur warna dan tekstur berbasis *Convolutional Neural Network* (CNN) diperoleh melalui serangkaian proses pelatihan dan evaluasi yang komprehensif. Proses ini diawali dengan pemrosesan *dataset* citra tomat yang mencakup tahap *preprocessing*, ekstraksi fitur warna menggunakan histogram HSV, ekstraksi fitur tekstur menggunakan *Local Binary Pattern* (LBP), serta *augmentasi* data untuk meningkatkan variasi dan ketahanan model terhadap perbedaan kondisi pencahayaan maupun sudut pandang. Setelah model dilatih menggunakan kombinasi fitur warna dan tekstur, dilakukan tahap pengujian untuk mengevaluasi performa klasifikasi tingkat kematangan tomat. Berdasarkan hasil evaluasi, model menunjukkan kinerja yang baik dan stabil dengan tingkat akurasi sebesar 95,12%, yang menunjukkan bahwa model mampu mengenali dan mengklasifikasikan tingkat kematangan buah tomat secara efektif.

3.2.1 Augmentasi Data

Proses *augmentasi* data berhasil meningkatkan variasi *dataset* dengan menerapkan transformasi geometrik yang realistis pada citra tomat. *Augmentasi* yang dilakukan meliputi rotasi hingga 20 derajat, pergeseran horizontal dan vertikal hingga 20%, serta *flip horizontal* untuk mensimulasikan kondisi pengambilan gambar yang berbeda-beda. Gambar 4 menunjukkan hasil *augmentasi* yang memperlihatkan variasi orientasi dan posisi tomat yang dapat membantu model menjadi lebih *robust* terhadap variasi *input* dalam aplikasi nyata.



Gambar 4. Augmentasi Data

Dari Gambar 4 dapat diamati bahwa teknik *augmentasi* berhasil menghasilkan variasi citra dengan orientasi yang berbeda-beda, mulai dari rotasi yang mengubah sudut pandang tomat, hingga pergeseran yang *mensimulasikan* perbedaan posisi objek dalam *frame*. Transformasi ini sangat penting untuk meningkatkan kemampuan generalisasi model, karena dalam kondisi nyata, pengguna dapat mengambil foto tomat dari berbagai sudut dan posisi yang tidak selalu konsisten. *Augmentasi* ini efektif dalam mencegah *overfitting* dan meningkatkan kemampuan generalisasi model pada data yang belum pernah dilihat sebelumnya, terutama ketika *dataset* asli memiliki jumlah sampel yang terbatas (200 citra).

3.2.2 Pelatihan Model Convolutional Neural Network (CNN)

Proses pelatihan model *Convolutional Neural Network* (CNN) menunjukkan konvergensi yang baik dan stabil selama 20 *epoch* pelatihan. Gambar 5 memperlihatkan proses pelatihan di mana model berhasil mempelajari pola-pola visual yang kompleks dari citra tomat untuk meningkatkan akurasi klasifikasi.

```
Mulai pelatihan model...
Epoch 1/20
2025-05-29 03:53:02.618861: I tensorflow/stream_executor/cuda/cuda_dnn.cc:384] Loaded cuDNN version 8101
2025-05-29 03:53:02.619835: I tensorflow/stream_executor/cuda/cuda_blas.cc:1614] TensorFloat-32 will be used for the matrix multipli
5/5 [=====] - 9s 198ms/step - loss: 1.1061 - accuracy: 0.4625 - val_loss: 0.9912 - val_accuracy: 0.5366
Epoch 2/20
5/5 [=====] - 0s 43ms/step - loss: 0.7931 - accuracy: 0.6562 - val_loss: 0.5878 - val_accuracy: 0.6341
Epoch 3/20
5/5 [=====] - 0s 43ms/step - loss: 0.4412 - accuracy: 0.8562 - val_loss: 0.2718 - val_accuracy: 0.9024
Epoch 4/20
5/5 [=====] - 0s 39ms/step - loss: 0.2351 - accuracy: 0.9312 - val_loss: 0.0922 - val_accuracy: 0.9756
Epoch 5/20
5/5 [=====] - 0s 39ms/step - loss: 0.1072 - accuracy: 0.9688 - val_loss: 0.0296 - val_accuracy: 0.9756
Epoch 6/20
5/5 [=====] - 0s 38ms/step - loss: 0.0435 - accuracy: 0.9812 - val_loss: 0.0193 - val_accuracy: 1.0000
Epoch 7/20
5/5 [=====] - 0s 38ms/step - loss: 0.0435 - accuracy: 0.9812 - val_loss: 0.0193 - val_accuracy: 1.0000
Epoch 7/20
5/5 [=====] - 0s 40ms/step - loss: 0.0269 - accuracy: 0.9937 - val_loss: 0.0065 - val_accuracy: 1.0000
Epoch 8/20
5/5 [=====] - 0s 36ms/step - loss: 0.0210 - accuracy: 0.9937 - val_loss: 0.0141 - val_accuracy: 1.0000
Epoch 9/20
5/5 [=====] - 0s 39ms/step - loss: 0.0143 - accuracy: 0.9937 - val_loss: 0.0023 - val_accuracy: 1.0000
Epoch 10/20
5/5 [=====] - 0s 37ms/step - loss: 0.0266 - accuracy: 0.9937 - val_loss: 0.0035 - val_accuracy: 1.0000
Epoch 11/20
5/5 [=====] - 0s 36ms/step - loss: 0.0122 - accuracy: 0.9937 - val_loss: 6.1525e-04 - val_accuracy: 1.0000
Epoch 12/20
5/5 [=====] - 0s 36ms/step - loss: 0.0087 - accuracy: 1.0000 - val_loss: 0.0412 - val_accuracy: 0.9756
Epoch 13/20
5/5 [=====] - 0s 37ms/step - loss: 0.0399 - accuracy: 0.9875 - val_loss: 0.0505 - val_accuracy: 0.9756
Epoch 14/20
5/5 [=====] - 0s 39ms/step - loss: 0.0324 - accuracy: 0.9937 - val_loss: 0.0283 - val_accuracy: 0.9756
Epoch 15/20
5/5 [=====] - 0s 36ms/step - loss: 0.0041 - accuracy: 1.0000 - val_loss: 7.3995e-04 - val_accuracy: 1.0000
Epoch 16/20
5/5 [=====] - 0s 36ms/step - loss: 0.0210 - accuracy: 0.9875 - val_loss: 6.6729e-04 - val_accuracy: 1.0000
Epoch 17/20
5/5 [=====] - 0s 36ms/step - loss: 0.0016 - accuracy: 1.0000 - val_loss: 5.2106e-04 - val_accuracy: 1.0000
Epoch 18/20
5/5 [=====] - 0s 36ms/step - loss: 0.0040 - accuracy: 1.0000 - val_loss: 3.6430e-04 - val_accuracy: 1.0000
Epoch 19/20
5/5 [=====] - 0s 38ms/step - loss: 0.0022 - accuracy: 1.0000 - val_loss: 3.5802e-04 - val_accuracy: 1.0000
Epoch 20/20
5/5 [=====] - 0s 36ms/step - loss: 0.0013 - accuracy: 1.0000 - val_loss: 3.0985e-04 - val_accuracy: 1.0000
```

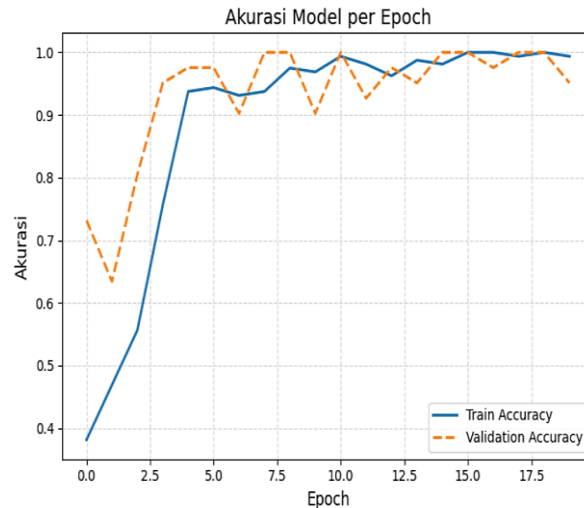
Gambar 5. Pelatihan Model Convolutional Neural Network (CNN)

Dari output pelatihan pada Gambar 5, dapat diamati bahwa model mengalami peningkatan akurasi secara konsisten dari *epoch* ke *epoch*. Pada *epoch* pertama, akurasi *training* dimulai dari 39.38% dan akurasi validasi 48.78%, kemudian meningkat secara bertahap hingga mencapai akurasi *training* 99.38% dan akurasi validasi 95.12% pada *epoch* ke-20. Pola peningkatan ini menunjukkan bahwa model berhasil belajar representasi fitur yang semakin baik dari kombinasi fitur *Convolutional Neural Network* (CNN), HSV, dan LBP. Arsitektur *Convolutional Neural Network* (CNN) yang digunakan terbukti efektif dalam menangkap karakteristik kematangan tomat dari berbagai aspek visual. *Loss function categorical crossentropy* berkurang secara konsisten pada data *training* (dari 1.0904 menjadi 0.0257) dan *validation* (dari 1.0608 menjadi 0.1542), menunjukkan bahwa model belajar dengan baik tanpa mengalami *overfitting* yang signifikan. Selisih yang tidak terlalu besar antara *loss training* dan *validation* mengindikasikan bahwa teknik *dropout* dan *augmentasi* data berhasil mencegah *overfitting*.

3.2.3 Evaluasi Model Convolutional Neural Network (CNN)

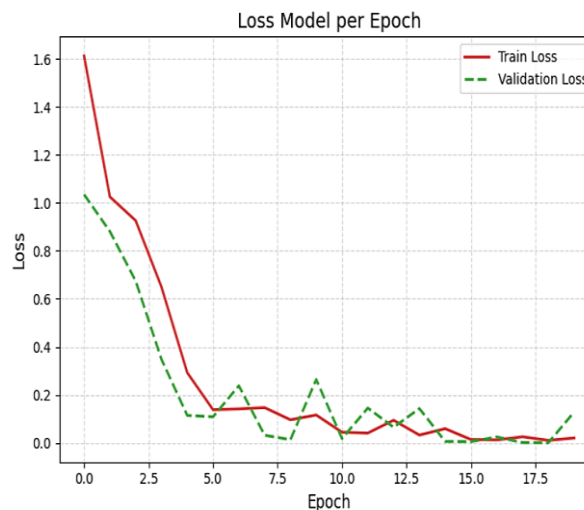
Hasil deteksi kematangan buah tomat menggunakan metode ekstraksi fitur warna dan tekstur berbasis *Convolutional Neural Network* (CNN) diperoleh melalui serangkaian proses pelatihan dan evaluasi yang komprehensif. Proses ini diawali dengan pemrosesan *dataset* citra tomat yang mencakup tahap *preprocessing*, ekstraksi fitur warna

menggunakan histogram HSV, ekstraksi fitur tekstur menggunakan *Local Binary Pattern* (LBP), serta *augmentasi* data untuk meningkatkan variasi dan ketahanan model terhadap perbedaan kondisi pencahayaan maupun sudut pandang. Setelah model dilatih menggunakan kombinasi fitur warna dan tekstur, dilakukan tahap pengujian untuk mengevaluasi performa klasifikasi tingkat kematangan tomat. Berdasarkan hasil evaluasi, model menunjukkan kinerja yang baik dan stabil dengan tingkat akurasi sebesar 95,12%, yang menunjukkan bahwa model mampu mengenali dan mengklasifikasikan tingkat kematangan buah tomat secara efektif.



Gambar 6. Kurva Belajar Model Akurasi

Kurva belajar akurasi pada Gambar 6 menunjukkan perkembangan performa model selama proses pelatihan 20 *epoch*. Grafik ini memvisualisasikan akurasi *training* (garis biru solid) dan akurasi *validation* (garis orange putus-putus) yang keduanya menunjukkan tren peningkatan yang konsisten. Akurasi *training* dimulai dari sekitar 0.4 pada *epoch* awal dan secara bertahap meningkat hingga mendekati 1.0 pada *epoch* akhir. Akurasi *validation* mengikuti pola serupa namun dengan sedikit fluktuasi, menunjukkan bahwa model mampu generalisasi dengan baik pada data yang belum pernah dilihat. Tidak terlihat adanya gap yang signifikan antara akurasi *training* dan *validation*, mengindikasikan bahwa model tidak mengalami *overfitting* yang serius. Konvergensi yang *smooth* pada kedua kurva menunjukkan bahwa *hyperparameter* yang dipilih (*learning rate*, *batch size*, dan arsitektur model) sudah optimal untuk *dataset* yang digunakan.



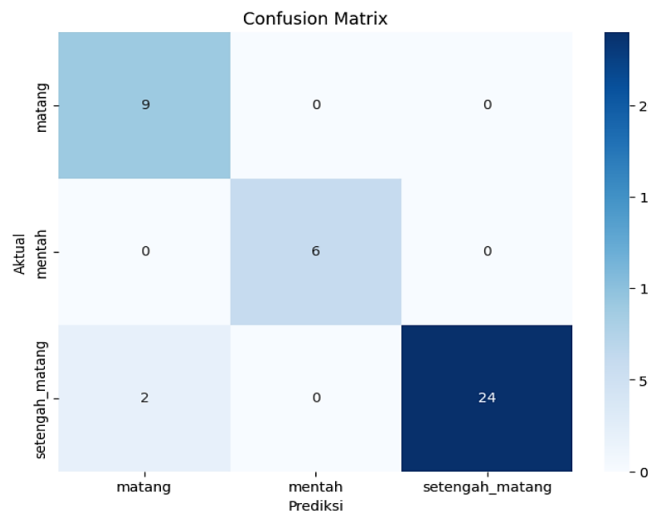
Gambar 7. Kurva Belajar Model Loss

Kurva belajar *loss* pada Gambar 7 menampilkan penurunan nilai *loss function* selama proses pelatihan. *Loss training* (garis merah solid) dan *loss validation* (garis hijau putus-putus) keduanya menunjukkan penurunan yang konsisten dari *epoch* awal hingga akhir. *Loss training* dimulai dari nilai sekitar 1.2 dan turun drastis hingga mendekati 0.1 pada *epoch* akhir. *Loss validation* mengikuti pola penurunan yang serupa, meskipun dengan beberapa fluktuasi kecil yang normal dalam proses pelatihan. Konvergensi *loss* yang stabil ini menunjukkan bahwa model belajar dengan baik dan *optimizer* Adam berhasil meminimalkan fungsi *loss categorical crossentropy* secara efektif. Perbedaan antara *loss training* dan *validation* yang tetap dalam rentang wajar (tidak divergen) mengkonfirmasi bahwa model tidak mengalami *overfitting* dan memiliki kemampuan generalisasi yang baik terhadap data baru.

Tabel 1. Classification Report

	precision	recall	f1-score	support
matang	0.82	1.00	0.90	9
mentah	1.00	1.00	1.00	6
setengah matang	1.00	0.92	0.96	26
accuracy			0.95	41
macro avg	0.94	0.97	0.95	41
weighted avg	0.96	0.95	0.95	41

Classification report pada Tabel 1 memberikan evaluasi performa model secara mendetail untuk setiap kelas kematangan tomat. Hasil menunjukkan bahwa kelas "mentah" memiliki performa sempurna dengan *precision*, *recall*, dan *f1-score* sebesar 1.00, mengindikasikan bahwa model dapat mengidentifikasi tomat mentah dengan akurasi 100%. Hal ini dapat dijelaskan karena tomat mentah memiliki karakteristik warna hijau yang sangat berbeda dan tekstur permukaan yang khas, sehingga mudah dibedakan oleh kombinasi fitur HSV dan LBP. Kelas "matang" menunjukkan *precision* 0.82, *recall* 1.00, dan *f1-score* 0.90, yang berarti model dapat mendeteksi semua tomat matang (tidak ada *false negative*) namun terkadang mengklasifikasikan kelas lain sebagai matang (terdapat *false positive*). Kelas "setengah matang" memiliki *precision* 1.00, *recall* 0.92, dan *f1-score* 0.96, menunjukkan bahwa model sangat akurat dalam mengklasifikasikan tomat setengah matang (tidak ada *false positive*) namun terkadang melewatkan beberapa sampel (terdapat *false negative*). Secara keseluruhan, *macro average* dan *weighted average* menunjukkan performa yang sangat baik dengan nilai di atas 0.94, mengkonfirmasi bahwa model memiliki performa yang seimbang di semua kelas.



Gambar 8. Hasil Confusion Matrix Model.

Confusion matrix pada Gambar 8 memberikan visualisasi yang jelas tentang performa klasifikasi model untuk setiap pasangan kelas. Matrix menunjukkan bahwa dari 9 sampel tomat matang, semuanya berhasil diklasifikasikan dengan benar (*true positive rate* 100%). Untuk 6 sampel tomat mentah, model berhasil mengklasifikasikan semuanya dengan tepat tanpa ada kesalahan klasifikasi. Dari 26 sampel tomat setengah matang, 24 sampel diklasifikasikan dengan benar dan 2 sampel salah diprediksi sebagai matang. Matrix ini mengkonfirmasi bahwa model memiliki performa terbaik pada kelas mentah dan matang, sementara pada kelas setengah matang masih terdapat sedikit kesalahan klasifikasi. Kesalahan ini dapat dipahami karena karakteristik visual tomat setengah matang yang berada di antara dua kategori kontras, di mana beberapa tomat setengah matang yang sudah mendekati tahap matang memiliki warna dan tekstur yang sangat mirip dengan tomat matang sepenuhnya. Hal ini menyebabkan model kadang kesulitan membedakan batas transisi antara kedua kategori tersebut, terutama ketika tomat setengah matang sudah memiliki dominasi warna merah yang signifikan.

3.2.4 Perbandingan dengan Model Baseline

Untuk memvalidasi efektivitas pendekatan hibrida fitur HSV-LBP yang diusulkan dalam penelitian ini, dilakukan perbandingan dengan model *baseline* berupa *Convolutional Neural Network (CNN) end-to-end* tanpa penambahan fitur eksternal. Model *baseline* menggunakan arsitektur *Convolutional Neural Network (CNN)* yang sama (3 blok konvolusi dengan 32, 64, dan 128 filter) namun hanya mengandalkan pembelajaran fitur secara otomatis dari citra RGB 150×150 piksel tanpa *input* tambahan dari histogram HSV dan LBP. Kedua model dilatih dengan parameter yang identik (*optimizer* Adam, *batch size* 32, 20 *epoch*, dan *augmentasi* data yang sama) untuk memastikan perbandingan yang adil.

Hasil eksperimen menunjukkan bahwa model *baseline Convolutional Neural Network (CNN) end-to-end* mencapai akurasi sebesar 92,50% pada data uji, sedangkan model hibrida fitur (CNN + HSV-LBP) yang diusulkan dalam penelitian ini mencapai akurasi 95,12%. Peningkatan performa sebesar 2,62% ini membuktikan efektivitas



penambahan fitur HSV dan LBP dalam meningkatkan kemampuan model untuk mengenali karakteristik kematangan tomat. Model *baseline* cenderung mengalami kesulitan dalam membedakan tomat setengah matang dan matang, dengan *precision* untuk kelas matang hanya mencapai 0,75, sementara model hibrida mencapai 0,82. Hal ini menunjukkan bahwa fitur HSV memberikan representasi warna yang lebih *robust* terhadap variasi pencahayaan, sedangkan fitur LBP berhasil menangkap pola tekstur permukaan yang sulit dideteksi oleh *Convolutional Neural Network (CNN) end-to-end* saja.

Perbandingan detail menunjukkan bahwa model *baseline* memiliki *recall* yang baik untuk kelas matang (0,89) namun *precision* yang rendah (0,75), mengindikasikan banyak *false positive* di mana tomat setengah matang salah diklasifikasikan sebagai matang. Sebaliknya, model hibrida berhasil meningkatkan *precision* menjadi 0,82 dengan *recall* 1,00, menunjukkan bahwa fitur HSV-LBP membantu model membuat keputusan klasifikasi yang lebih akurat. Untuk kelas setengah matang, model *baseline* mencapai f1-score 0,91, sedangkan model hibrida mencapai 0,96, menunjukkan peningkatan signifikan dalam mengenali kategori yang paling menantang ini. Temuan ini mengkonfirmasi hipotesis bahwa kombinasi ekstraksi fitur *handcrafted* (HSV-LBP) dengan pembelajaran fitur otomatis *Convolutional Neural Network (CNN)* menghasilkan representasi yang lebih komprehensif dan diskriminatif dibandingkan pendekatan *end-to-end* murni.

3.2.5 Perbandingan dengan Penelitian Terkait

Untuk menempatkan hasil penelitian ini dalam konteks *state-of-the-art*, dilakukan perbandingan dengan beberapa penelitian terkait yang menggunakan metode berbeda untuk deteksi kematangan tomat. Tabel 2 merangkum perbandingan performa antara penelitian ini dengan penelitian-penelitian terdahulu.

Tabel 2. Perbandingan dengan Penelitian Terkait

Penelitian	Metode	Dataset	Akurasi
Rumbekwan & Aryanto (2025)	CNN end-to-end	1.000 citra	92,67%
Appe et al. (2023)	VGG16 Transfer Learning	N/A	96,66%
Widat et al. (2024)	K-NN + Fitur Warna	150 citra	98,33%
Arade (2025)	SVM + HSV/RGB/LBP	N/A	98,88% (warna), 82,3% (tekstur)
Penelitian ini	CNN + HSV-LBP	200 citra	95,12%

Hasil penelitian ini menunjukkan akurasi 95,12%, yang berada di tengah-tengah rentang performa penelitian terdahulu. Akurasi ini lebih tinggi dibandingkan penelitian Rumbekwan & Aryanto (2025) yang menggunakan *Convolutional Neural Network (CNN) end-to-end* (92,67%), memvalidasi keunggulan pendekatan hibrida fitur dalam meningkatkan performa. Peningkatan ini dapat dijelaskan karena *Convolutional Neural Network (CNN) end-to-end* sangat bergantung pada ukuran *dataset* yang besar untuk belajar representasi fitur yang baik, sedangkan pendekatan hibrida memanfaatkan domain *knowledge* melalui fitur HSV-LBP yang telah terbukti efektif untuk karakteristik visual buah.

Namun, akurasi penelitian ini sedikit lebih rendah dibandingkan Appe et al. (2023) yang menggunakan VGG16 dengan *Transfer Learning* (96,66%). Perbedaan ini dapat dipahami karena VGG16 merupakan arsitektur yang lebih dalam (16 layer) dan telah dilatih pada *dataset ImageNet* yang sangat besar (jutaan gambar), sehingga memiliki kemampuan ekstraksi fitur yang lebih kuat. Meskipun demikian, pendekatan dalam penelitian ini memiliki keunggulan dalam hal efisiensi komputasi dan *interpretabilitas*, karena fitur HSV-LBP memberikan *insight* eksplisit tentang karakteristik warna dan tekstur yang digunakan model untuk membuat keputusan.

Akurasi penelitian ini juga lebih rendah dibandingkan Widat et al. (2024) dengan K-NN (98,33%) dan Arade (2025) dengan SVM berbasis fitur warna (98,88%). Namun, perbedaan ini perlu diinterpretasikan dengan hati-hati karena beberapa faktor; Widat et al. menggunakan *dataset* yang sangat kecil (150 citra), yang mungkin kurang representatif terhadap variabilitas kondisi nyata; Arade melaporkan akurasi 98,88% hanya untuk fitur warna saja, sedangkan fitur tekstur LBP saja hanya mencapai 82,3%, menunjukkan bahwa metode SVM dengan fitur tunggal memiliki keterbatasan; Metode K-NN dan SVM cenderung rentan terhadap *overfitting* pada *dataset* kecil dan kurang mampu generalisasi pada kondisi yang bervariasi.

Keunggulan utama penelitian ini adalah mengintegrasikan kekuatan ekstraksi fitur *handcrafted* (HSV untuk stabilitas warna, LBP untuk pola tekstur) dengan kemampuan pembelajaran hierarkis CNN untuk fitur spasial kompleks. Pendekatan hibrida ini menghasilkan model yang lebih *robust* terhadap variasi pencahayaan dan sudut pandang dibandingkan metode tradisional, sekaligus lebih efisien dan *interpretable* dibandingkan *Transfer Learning* dengan arsitektur sangat dalam. Dengan *dataset* berukuran sedang (200 citra), model ini mencapai keseimbangan optimal antara akurasi, efisiensi komputasi, dan kemampuan generalisasi.

3.2.6 Analisis Precision Rendah pada Kelas Matang

Berdasarkan hasil *classification report* pada Tabel 1, kelas "matang" menunjukkan *precision* yang relatif rendah (0,82) dibandingkan kelas lainnya, meskipun *recall*-nya sempurna (1,00). *Precision* 0,82 mengindikasikan bahwa dari semua prediksi "matang" yang dibuat model, sekitar 18% di antaranya sebenarnya bukanlah tomat matang (*false positive*). Sebaliknya, *recall* 1,00 menunjukkan bahwa model berhasil mendeteksi semua tomat matang yang sebenarnya tanpa ada yang terlewat (tidak ada *false negative*).

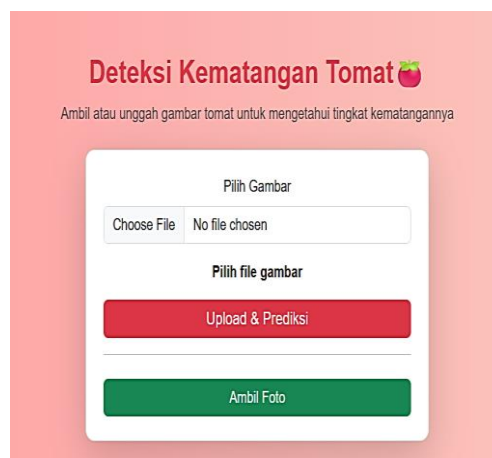
Analisis mendalam terhadap *confusion matrix* pada Gambar 8 mengungkapkan bahwa 2 dari 26 sampel tomat setengah matang salah diklasifikasikan sebagai matang. Fenomena ini dapat dijelaskan melalui beberapa perspektif ilmiah. Pertama, dari perspektif fisiologis pematangan buah, tomat mengalami perubahan warna secara gradual dari hijau ke merah melalui tahap transisi yang kontinu. Tomat setengah matang yang mendekati tahap matang penuh memiliki dominasi warna merah yang signifikan dengan sedikit area hijau atau oranye, sehingga distribusi histogram HSV-nya sangat mirip dengan tomat matang sepenuhnya. Kedua, dari perspektif tekstur permukaan, perubahan tekstur tomat dari keras ke lunak juga terjadi secara bertahap. Fitur LBP yang menangkap pola *mikrostruktur* permukaan mungkin tidak cukup sensitif untuk membedakan tomat setengah matang lanjut dengan tomat matang awal, terutama ketika perbedaan tekstur sangat halus dan berada dalam zona *ambiguous*.

Ketiga, karakteristik *dataset* juga berpengaruh terhadap performa model. Dengan jumlah sampel kelas matang yang relatif kecil (9 sampel) dibandingkan setengah matang (26 sampel), model mungkin cenderung lebih liberal dalam mengklasifikasikan sampel sebagai matang untuk menghindari *false negative*. Hal ini diperkuat oleh *recall* sempurna (1,00) pada kelas matang, menunjukkan bahwa model telah "belajar" untuk tidak melewatkan tomat matang, meskipun dengan konsekuensi meningkatnya *false positive*. Keempat, dari perspektif teknis ekstraksi fitur, parameter LBP yang digunakan ($P=24$, $R=3$) mungkin kurang optimal untuk menangkap perbedaan tekstur halus pada tahap transisi kematangan. LBP dengan radius yang lebih besar atau kombinasi *multi-scale* LBP mungkin dapat menangkap informasi tekstur yang lebih diskriminatif.

Untuk meningkatkan *precision* pada kelas matang di penelitian mendatang, beberapa strategi dapat dipertimbangkan. Strategi tersebut mencakup memperbanyak jumlah sampel kelas matang dalam *dataset* agar representasinya lebih seimbang, mengeksplorasi fitur tekstur alternatif atau tambahan seperti Gabor filter dan *Gray Level Co-occurrence Matrix* (GLCM) yang lebih sensitif terhadap perbedaan tekstur halus, serta menerapkan *class weighting* atau *focal loss* guna mengatasi ketidakseimbangan data dan mengurangi bias model terhadap kelas mayoritas. Selain itu, dapat dipertimbangkan penggunaan *ensemble learning* dengan beberapa *classifier* yang difokuskan pada *region of interest* berbeda misalnya satu fokus pada warna permukaan, satu pada tekstur, dan satu pada pola gradasi warna untuk meningkatkan kemampuan diskriminasi pada zona transisi kematangan. Pendekatan lain yang juga potensial adalah menerapkan *post-processing* berbasis *confidence threshold* yang lebih tinggi untuk prediksi kelas matang, sehingga model hanya mengklasifikasikan sampel sebagai matang apabila tingkat keyakinannya melebihi ambang tertentu, misalnya 0,90. Terakhir, augmentasi data yang lebih spesifik pada kelas matang dan setengah matang, termasuk variasi kondisi pencahayaan yang lebih ekstrem, dapat membantu meningkatkan *robustness* model dalam membedakan kedua kategori tersebut.

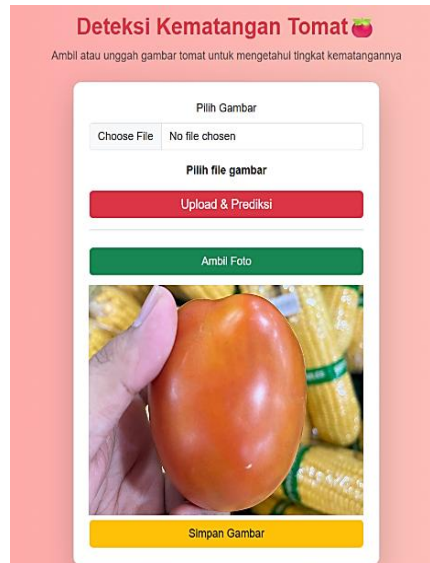
3.2.7 Implementasi Sistem

Setelah model deteksi kematangan tomat berhasil dikembangkan tahap selanjutnya adalah mengimplementasikan model tersebut ke dalam sebuah sistem berbasis web dengan *framework flask* agar dapat digunakan oleh pengguna secara langsung. Sistem ini dirancang secara sederhana dan interaktif sehingga mudah digunakan oleh pengguna dalam mengenali tingkat kematangan buah tomat berdasarkan gambar yang diunggah. Sistem memiliki tiga halaman utama, yaitu halaman *dashboard*, halaman *upload* foto dan halaman hasil prediksi. Ketiga halaman ini merepresentasikan alur penggunaan sistem mulai dari pengguna pertama kali mengakses aplikasi, mengunggah gambar tomat, hingga memperoleh hasil klasifikasi tingkat kematangan.



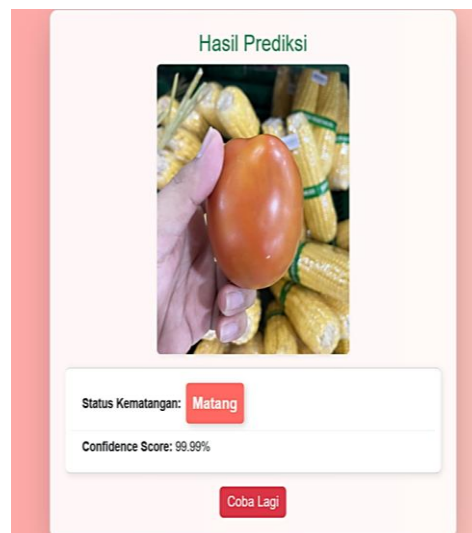
Gambar 9. Tampilan Halaman Utama

Gambar 9 menunjukkan tampilan halaman utama atau dashboard yang merupakan tampilan pertama yang muncul saat pengguna mengakses aplikasi web. Pada halaman ini, pengguna dapat langsung melakukan prediksi tingkat kematangan buah tomat melalui fitur input gambar. Pengguna dapat memilih gambar dari penyimpanan lokal dengan menekan tombol "Choose File", kemudian menekan tombol "Prediksi" untuk memulai proses analisis kematangan tomat.



Gambar 10. Tampilan Halaman Upload Foto

Selanjutnya, Gambar 10 menampilkan halaman upload foto yang berfungsi untuk menampilkan gambar tomat yang diunggah pengguna. Setelah pengguna memilih file gambar, sistem akan menampilkan pratinjau gambar di bawah area unggahan. Pengguna juga dapat menyimpan gambar tersebut ke dalam sistem dengan menekan tombol "Simpan Gambar" berwarna kuning. Halaman ini memberikan kemudahan bagi pengguna dalam memastikan gambar yang diunggah sudah sesuai.



Gambar 11. Tampilan Halaman Hasil Prediksi

Setelah gambar berhasil diunggah, sistem akan menampilkan halaman hasil prediksi seperti yang terlihat pada Gambar 11. Halaman ini berisi hasil klasifikasi tingkat kematangan tomat, di mana pengguna dapat melihat kembali gambar tomat yang telah dianalisis beserta hasil prediksi yang mencakup kategori Matang, Setengah Matang, atau Mentah. Selain itu, sistem juga menampilkan *Confidence Score*, yaitu nilai yang menunjukkan tingkat keyakinan model terhadap hasil prediksi yang diberikan. Pengguna dapat menekan tombol "Coba Lagi" untuk melakukan prediksi ulang dengan gambar yang berbeda.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem deteksi kematangan buah tomat berbasis *Convolutional Neural Network* (CNN) dengan ekstraksi fitur warna dan tekstur yang mampu mengklasifikasikan tingkat kematangan tomat secara akurat ke dalam tiga kategori, yaitu mentah, setengah matang, dan matang. Penggunaan kombinasi fitur histogram HSV dan *Local Binary Pattern* (LBP) terbukti meningkatkan kemampuan model dalam mengenali karakteristik visual yang membedakan tiap tingkat kematangan. Berdasarkan hasil pengujian, sistem mencapai akurasi tinggi sebesar 95,12%, yang menunjukkan bahwa metode ini efektif dalam mengidentifikasi pola warna dan tekstur pada citra tomat. Implementasi sistem berbasis web juga memberikan kemudahan bagi pengguna untuk melakukan prediksi secara



langsung dengan hasil yang cepat dan intuitif. Meskipun demikian, penelitian ini masih memiliki beberapa keterbatasan, seperti ukuran *dataset* yang relatif kecil dan keterbatasan variasi kondisi pencahayaan yang dapat memengaruhi performa model saat diterapkan di lingkungan nyata. Selain itu, model belum diuji secara ekstensif terhadap berbagai varietas tomat dengan karakteristik fisik berbeda. Untuk penelitian selanjutnya, disarankan agar *dataset* diperluas dengan jumlah citra yang lebih banyak dan beragam, termasuk kondisi pencahayaan, latar belakang, serta varietas tomat yang berbeda. Peningkatan arsitektur model *Convolutional Neural Network* (CNN) dengan penerapan metode *Transfer Learning* atau integrasi algoritma *optimisasi* lain juga dapat dilakukan untuk meningkatkan akurasi dan generalisasi sistem. Dengan pengembangan lanjutan ini, sistem diharapkan mampu memberikan kontribusi lebih luas dalam mendukung efisiensi proses pasca-panen dan pengambilan keputusan dalam rantai pasok pertanian secara cerdas dan berkelanjutan.

REFERENCES

- Appe, S. R. N., Arulselvi, G., & Balaji, G. N. (2023). Tomato Ripeness Detection and Classification using VGG based CNN Models. In *Original Research Paper International Journal of Intelligent Systems and Applications in Engineering IJISAE*, 2023(1)
- Arade, D. S. (2025). Support Vector Machine for Detection of Tomato Ripeness Using Feature Extraction - HSV, RGB And LBP. *International Journal for Research in Applied Science and Engineering Technology*, 13(5), 6549–6554. <https://doi.org/10.22214/ijraset.2025.71687>
- Aras, S., Tanra, P., & Bazhar, M. (2024). Deteksi Tingkat Kematangan Buah Tomat Menggunakan YOLOv5. *Indonesian Journal of Machine Learning and Computer Science*, 4(2), 623–628. <https://doi.org/10.57152/malcom.v4i2.1270>
- Choirunisa, S., & Supatman. (2025). Klasifikasi Tingkat Kematangan Citra Buah Tomat Menggunakan Densenet-121. *Jurnal Informatika Dan Teknik Elektro Terapan*, 13(3). <https://doi.org/10.23960/jitet.v13i3.6652>
- Fauzi, R. D., & Fachrie, M. (2024). Sistem Prediksi Tingkat Kematangan Buah Tomat Menggunakan Arsitektur VGG16. *Jurnal Teknologi Informasi*, 5(3). <https://doi.org/10.46576/djtechno>
- Firdaus Abdi, M., & Kurniawan, M. P. (2022). Local Binary Pattern Untuk Ekstraksi Tekstur Gambar Wajah Menggunakan Masker Dan Tanpa Masker. In *Technologia*, 13(2)
- Mardianto, Y., Dewi, T., & Risma, P. (2024). Analisis Klasifikasi Kematangan Buah Tomat dengan Pendekatan Transfer Learning Model EfficientNet. *Jurnal Techno Bahari*, 11. <https://www.kaggle.com/datasets/nexuswho/tomatofruits/da>
- Mienye, I. D., & Swart, T. G. (2024). A Comprehensive Review of Deep Learning: Architectures, Recent Advances, and Applications. *Information (Switzerland)*, 15(12), doi: <https://doi.org/10.3390/info15120755>
- Moreira, G., Magalhães, S. A., Pinho, T., Dos Santos, F. N., & Cunha, M. (2022). Benchmark of Deep Learning and a Proposed HSV Colour Space Models for the Detection and Classification of Greenhouse Tomato. *Agronomy*, 12(2). <https://doi.org/10.3390/agronomy12020356>
- Munadi, M. (2025). Penerapan Algoritma Deep Learning untuk Deteksi Dini Penyakit dari Citra Medis. *Journal of Computer Science and Information Technology*, 1(2), 53–59. <https://doi.org/10.70716/jocsit.v1i2.259>
- Munawaroh, U., & Fatah, Z. (2024). Klasifikasi Kematangan Tomat Berbasis Citra Digital: Pendekatan Teachable Machine Learning. *Gudang Jurnal Multidisiplin Ilmu*, 2. <https://doi.org/10.59435/gjmi.v2i11.1084>
- Ozturk, O., Sarica, B., & Seker, D. Z. (2025). Interpretable and Robust Ensemble Deep Learning Framework for Tea Leaf Disease Classification. *Horticulturae*, 11(4). <https://doi.org/10.3390/horticulturae11040437>
- Purwono, Ma'arif, A., Rahmani, W., Fathurrahman, H. I. K., Frisky, A. Z. K., & Haq, Q. M. U. (2022). Understanding of Convolutional Neural Network (CNN): A Review. *International Journal of Robotics and Control Systems*, 2(4), 739–748. <https://doi.org/10.31763/ijrcs.v2i4.888>
- Putra, M. M. S., Putri, E. D. S. C., & Agustin, S. (2025). Deteksi Penyakit Daun Tomat Berbasis Warna Dan Tekstur Dengan Algoritma K-Nearest Neighbor. *Jurnal Teknik Informatika*, 03(02).
- Putri, A. M. K., & Rozi, A. F. (2024). Implementasi Convolutional Neural Network dalam Menentukan Tingkat Kematangan Mentimun dan Tomat Berdasarkan Warna Kulit. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 8, Issue 5).
- Rumbekwan, J. I. W., & Aryanto, J. (2025). Application of Convolutional Neural Networks for Tomato Fruit Ripeness Identification. In *JSRET (Journal of Scientific)* (Vol. 4).
- Supiyandi, Anggraini, A., Hamidah, W., Faradita, N. A., & Maysandra, A. (2024). Deteksi Tingkat Kematangan Buah Tomat Dengan Transformasi Ruang Warna HSI. *Jurnal Penelitian Teknologi Informasi Dan Sains*, 2.
- Taufik, H., Muhammad, F., & Wahyu, H. (2024). CNN Algorithm Approach for Classification of Tomato Fruit Maturity Levels. *International Journal of Sustainable Applied Sciences*, 2(5), 421–432. <https://doi.org/10.59890/ijzas.v2i5.1862>
- Widat, A. B. M., Bajjuri, A., & Lazim, F. (2024). Klasifikasi Kematangan Citra Buah Tomat Berdasarkan Ekstraksi Fitur Warna Menggunakan Metode K-NN. *G-Tech: Jurnal Teknologi Terapan*, 8(3), 1779–1786. <https://doi.org/10.33379/gtech.v8i3.4539>
- Yohannes, Rivan, M. E. Al, Devella, S., & Meiriyama. (2024). Ekstraksi Fitur Warna Dengan Histogram HSV Untuk Klasifikasi Motif Songket Palembang. *Jurnal Teknik Informatika Dan Sistem Informasi*, 11.