Jurnal Teknologi Pembelajaran Interaktif JTPI

https://journalversa.com/s/index.php/jtpi Vol. 5, No. 1 Maret 2025

IMPLEMENTASI NEURAL NETWORK UNTUK DETEKSI OTOMATIS JENIS BUAH BERDASARKAN CITRA WARNA DAN BENTUK MENGGUNAKAN MATLAB

Ihza Reza Fachrezzy S.A[1], Eka Nuryanto Budisusila[2]

[1,2]Universitas Islam Sultan Agung

Email: ihza.rfch@gmail.com[1], ekanbs@unissula.ac.id[2]

Abstrak: Penelitian ini bertujuan untuk mengembangkan sistem deteksi otomatis jenis buah dengan memanfaatkan metode jaringan syaraf tiruan (Neural Network) yang diimplementasikan pada platform MATLAB. Sistem ini mengklasifikasikan lima jenis buah, yaitu pisang, alpukat, apel, jeruk, dan kiwi, berdasarkan ciri-ciri warna dan bentuk buah. Gambar buah diambil menggunakan kamera smartphone dalam kondisi pencahayaan yang terkontrol dan latar belakang putih. Pengolahan citra dilakukan dengan teknik Convolutional Neural Network (CNN) untuk deteksi warna, serta deteksi tepi untuk ekstraksi fitur bentuk (eksentrisitas dan area). Integrasi logika fuzzy digunakan untuk menggabungkan hasil deteksi warna dan bentuk guna memperoleh keputusan akhir. Pendekatan eksperimental digunakan dalam penelitian ini, di mana sistem dirancang, diimplementasikan, dan diuji untuk mengukur akurasi deteksi. Hasil pengujian menunjukkan bahwa sistem ini mampu mendeteksi jenis buah dengan tingkat akurasi lebih dari 90%, yang mengindikasikan potensi penerapannya dalam otomatisasi proses penyortiran buah di industri pertanian dan pangan. Penelitian ini memberikan kontribusi signifikan dalam pengembangan teknologi berbasis kecerdasan buatan untuk pengolahan citra digital, serta membuka peluang inovasi di bidang pertanian dan logistik.

Kata Kunci: Neural Network, Convolutional Neural Network (CNN), MATLAB, Deteksi Buah, Warna, Bentuk, Logika Fuzzy

Abstract: *This study aims to develop an automated fruit type detection system based on color and shape features using Neural Network methods implemented on the MATLAB platform. The system utilizes dominant features such as color and shape (eccentricity and area) to classify five types of fruits: bananas, avocados, apples, oranges, and kiwis. Fruit image data were collected using a smartphone camera in a controlled environment and processed using Convolutional Neural Network (CNN) methods for training and feature detection of color. Meanwhile, shape features were evaluated using edge detection and statistical analysis methods. This research adopts an experimental approach, where the system is designed, implemented, and tested to measure detection accuracy. The color and shape detection models were integrated using fuzzy logic to produce the final decision. The test results showed that the system could detect fruit types with an accuracy rate of over 90%, demonstrating its potential application in automating fruit sorting processes in the industry. This study significantly contributes to the development of artificial intelligence-based technologies for digital image processing while opening up further innovation opportunities in agriculture and food sectors.*

Keywords: *Neural Network, Convolutional Neural Network (CNN), MATLAB, Fruit Detection, Color, Shape, Fuzzy Logic*

PENDAHULUAN

Klasifikasi buah secara otomatis merupakan salah satu topik penting dalam bidang pertanian dan pengolahan pangan. Buah-buahan memiliki variasi warna, bentuk, ukuran, dan tekstur yang sangat bervariasi, yang dapat menyulitkan proses pengelompokan dan penyortiran jika dilakukan secara manual. Metode manual dalam klasifikasi buah memiliki kelemahan, seperti ketergantungan pada kondisi fisik pengamat, kelelahan, dan subyektivitas. Oleh karena itu, diperlukan sistem otomatis yang lebih efisien dan dapat memberikan hasil yang lebih konsisten.

Dalam era Revolusi Industri 4.0, teknologi berbasis kecerdasan buatan (AI) dan visi komputer berkembang pesat, menawarkan solusi yang sangat efektif untuk berbagai aplikasi, termasuk deteksi dan klasifikasi objek. Salah satu metode yang sangat efektif dalam pengolahan citra adalah Convolutional Neural Network (CNN), yang telah banyak digunakan dalam berbagai penelitian terkait klasifikasi citra. Sebagai contoh, CNN telah digunakan dalam berbagai aplikasi seperti pengenalan wajah, identifikasi objek, dan klasifikasi buah.

Namun, meskipun berbagai penelitian telah menunjukkan keberhasilan CNN dalam klasifikasi gambar, tantangan masih ada terutama dalam mengklasifikasikan buah yang memiliki kesamaan warna dan bentuk. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan sistem deteksi otomatis jenis buah dengan menggunakan CNN yang menggabungkan ekstraksi fitur warna dan bentuk buah. Hasil dari deteksi warna dan bentuk ini kemudian diproses menggunakan logika fuzzy untuk meningkatkan akurasi dan konsistensi deteksi.

Tujuan Penelitian

1. Mengembangkan model jaringan syaraf tiruan untuk mengklasifikasikan buah berdasarkan fitur warna dan bentuk.
2. Mengimplementasikan model tersebut menggunakan platform MATLAB untuk pengolahan citra dan pelatihan jaringan syaraf tiruan.
3. Menguji akurasi sistem dan mengevaluasi potensi penerapannya dalam industri pertanian dan pangan.

METODE PENELITIAN

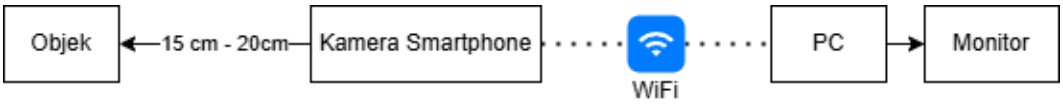
Desain Sistem

Penelitian ini mengembangkan sistem deteksi jenis buah yang terdiri dari dua komponen utama: perangkat keras dan perangkat lunak. Desain sistem secara rinci dijelaskan dalam subbagian berikut.

1. Perangkat Keras

Perangkat keras utama yang digunakan dalam penelitian ini meliputi:

- PC/Laptop: Digunakan untuk pemrosesan gambar dan pelatihan model jaringan syaraf tiruan menggunakan MATLAB dan Python.
- Kamera Smartphone: Digunakan untuk mengambil gambar buah dengan jarak 15-20 cm dari buah. Kamera smartphone dipilih karena kemampuan resolusinya yang cukup tinggi dan kemudahan penggunaan dalam pengambilan gambar secara langsung.
- Studio Mini: Sebuah studio mini dengan latar belakang putih dan pencahayaan LED untuk memastikan kondisi pencahayaan yang terkontrol. Studio ini bertujuan untuk menghilangkan gangguan visual yang dapat memengaruhi hasil deteksi.



Gambar 1. Diagram blok rancangan hardware

Keterangan:

- Objek berupa buah yang dideteksi dengan jarak 15cm – 20cm dari lensa kamera. Objek diletakkan pada studio mini.
- Kamera smartphone dengan spesifikasi 50MP, f1.8, 720 x 480
- PC/Laptop dengan spesifikasi AMD Ryzen 7 3700U 2.3 ~ 4.0 GHz, 16GB, AMD Radeon RX Vega 10, Windows 11 64-bit
- Monitor merupakan komponen display bawaan dari PC dengan spesifikasi FHD 1920 x 1080, 15 inch, 60Hz
- Studio Mini berbentuk kotak dengan latar belakang putih, pencahayaan LED, dimensi 41cm x 30cm x 20cm

2. Perangkat Lunak

Platform yang digunakan untuk pengembangan sistem ini adalah MATLAB, yang menyediakan toolbox pengolahan citra dan jaringan syaraf tiruan. MATLAB adalah platform komputasi numerik yang kuat, yang menyediakan berbagai fungsi dan toolbox untuk pengembangan dan pelatihan model jaringan syaraf tiruan. Deep Learning Toolbox dalam MATLAB memungkinkan pengguna untuk membangun, melatih, dan menguji model CNN dengan mudah, serta mendukung penggunaan GPU untuk akselerasi komputasi [1]. Python digunakan untuk pengambilan gambar dan pemrosesan citra sebelum data diteruskan ke MATLAB. Model CNN untuk klasifikasi gambar dibangun menggunakan Deep Learning Toolbox di MATLAB.

Beberapa aplikasi yang terinstal pada smartphone dan juga PC/Laptop, pada PC yaitu:

1. MATLAB versi R22a
2. VsCode versi 1.95.3
3. Python versi 3.13.0
4. Droidcam Windows Client versi 6.5.2

Pada smartphone hanya satu aplikasi saja, yaitu Droidcam Classic versi 6.27 (Android).

Pengumpulan Data

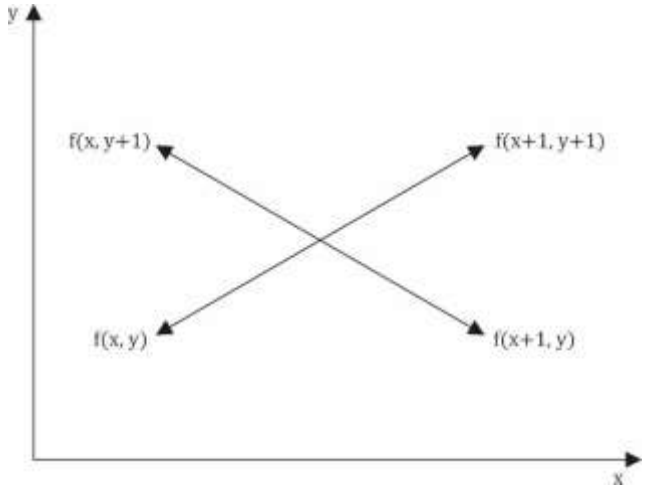
Data yang digunakan dalam penelitian ini adalah gambar dari lima jenis buah: pisang, alpukat, apel, jeruk, dan kiwi. Setiap jenis buah memiliki 90 sampel gambar, yang diambil dalam kondisi pencahayaan yang terkontrol dengan latar belakang putih. Gambar diambil menggunakan kamera smartphone dengan jarak antara kamera dan buah sekitar 15-20 cm.

Pengolahan Citra

Pengolahan citra digital merujuk kepada cara memproses citra dua dimensi mempergunakan media computer[2]. Proses pengolahan citra dimulai dengan pra-pemrosesan gambar untuk mempersiapkan citra agar dapat digunakan oleh jaringan syaraf tiruan. Langkah-langkah pengolahan citra meliputi:

1. Cropping dan Resizing: Gambar dipotong menjadi format persegi dan diubah ukurannya menjadi 200x200 piksel untuk standar input jaringan.
2. Konversi Grayscale: Citra dikonversi ke grayscale untuk analisis bentuk, karena fitur bentuk tidak memerlukan informasi warna.
3. Ekstraksi Fitur:

- Warna: Digunakan fitur RGB untuk ekstraksi warna yang kemudian diproses menggunakan CNN. Warna dapat dengan mudah diukur dalam ruang warna seperti RGB atau HSV, dan sering digunakan untuk segmentasi gambar [3].
- Bentuk: Deteksi tepi dilakukan menggunakan operator Roberts, dan nilai eksentrisitas serta area dihitung untuk mengidentifikasi bentuk buah.



(1)

Roberts bekerja dengan menerapkan dua kernel (filter) 2x2 pada citra untuk menghitung gradien intensitas dalam dua arah diagonal, yaitu:

Kernel untuk arah diagonal utama:

$$G_x = \begin{bmatrix} +1 & 0 \\ 0 & -1 \end{bmatrix} \quad (2)$$

Kernel untuk arah diagonal sekunder:

$$G_x = \begin{bmatrix} 0 & +1 \\ -1 & 0 \end{bmatrix} \quad (3)$$

Gradien gabungan dihitung sebagai:

$$G = \sqrt{G_x^2 + G_y^2} \quad (4)$$

Eksentrisitas, adalah ukuran yang menunjukkan seberapa bulat atau lonjong bentuk suatu objek.

$$E = \sqrt{1 - \left(\frac{b}{a}\right)^2} \quad (5)$$

Luas atau Area, adalah ukuran total luas permukaan suatu objek dalam domain gambar.

$$A = \sum_{i,j} f(i,j) \tag{6}$$

Model Jaringan Saraf Tiruan (CNN)

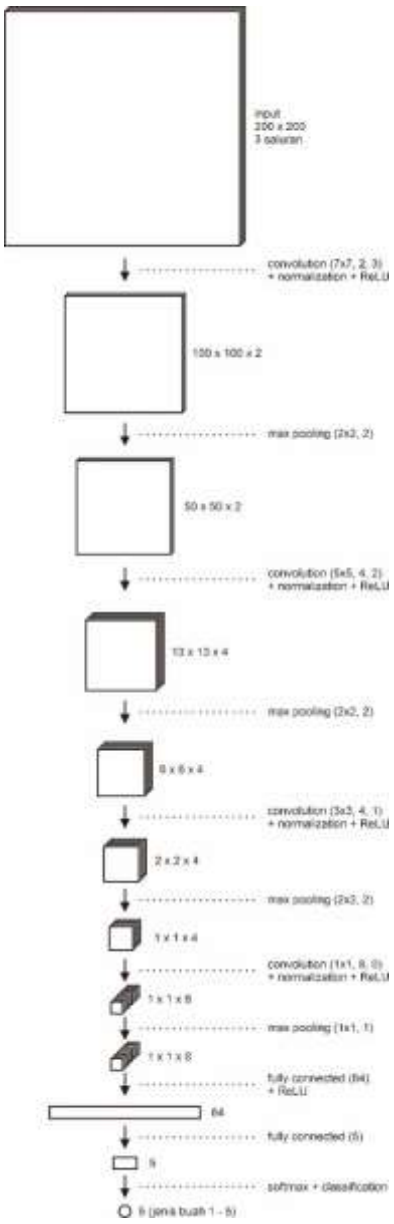
Metode Convolutional Neural Network memiliki hasil yang paling signifikan dalam pengenalan citra digital [4]. CNN termasuk dalam jenis Deep Neural Network karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra [5]. Model CNN yang digunakan dalam penelitian ini memiliki beberapa lapisan sebagai berikut:

1. Lapisan Input: Ukuran input adalah 200x200x3 untuk citra warna.
2. Lapisan Konvolusi: Terdiri dari tiga lapisan konvolusi dengan ukuran filter 7x7, 5x5, dan 3x3, masing-masing diikuti oleh lapisan ReLU dan lapisan pooling untuk ekstraksi fitur.

$$Output\ convolution = \frac{inputsize + 2 \times padding - kernel}{stride} + 1 \tag{7}$$

$$Output\ Pooling = \frac{inputsize - kernel}{stride} + 1 \tag{8}$$

3. Lapisan Fully Connected: Setelah ekstraksi fitur, data dihubungkan menggunakan lapisan fully connected untuk pengklasifikasian.
4. Lapisan Output: Menggunakan lapisan softmax untuk menghasilkan prediksi kelas buah (pisang, alpukat, apel, jeruk, dan kiwi).



Gambar 2. Arsitektur CNN

Integrasi Logika Fuzzy

Logika fuzzy digunakan untuk menggabungkan hasil deteksi dari fitur warna dan bentuk. Setiap hasil deteksi diberikan nilai fuzzy dan diolah dengan aturan-aturan fuzzy yang dirancang berdasarkan ambang batas eksentrisitas dan area serta dari hasil klasifikasi berdasarkan warna. Hasil akhirnya adalah keputusan tentang jenis buah yang terdeteksi.

HASIL DAN PEMBAHASAN

Image Processing

Processing citra berjalan pada dua platform, yaitu MATLAB dan Python. Pada Python berjalan preprocessing citra, dengan hasil berikut:

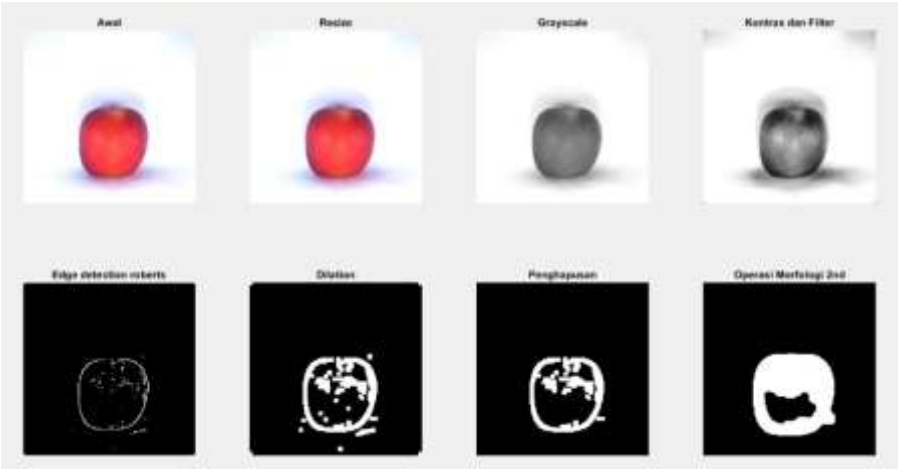
Gambar 4. Citra asli dari kamera smartphone

Gambar 5. Hasil resize preprocessing

Gambar 6. Hasil rotating preprocessing

Gambar 7. Hasil citra akhir preprocessing

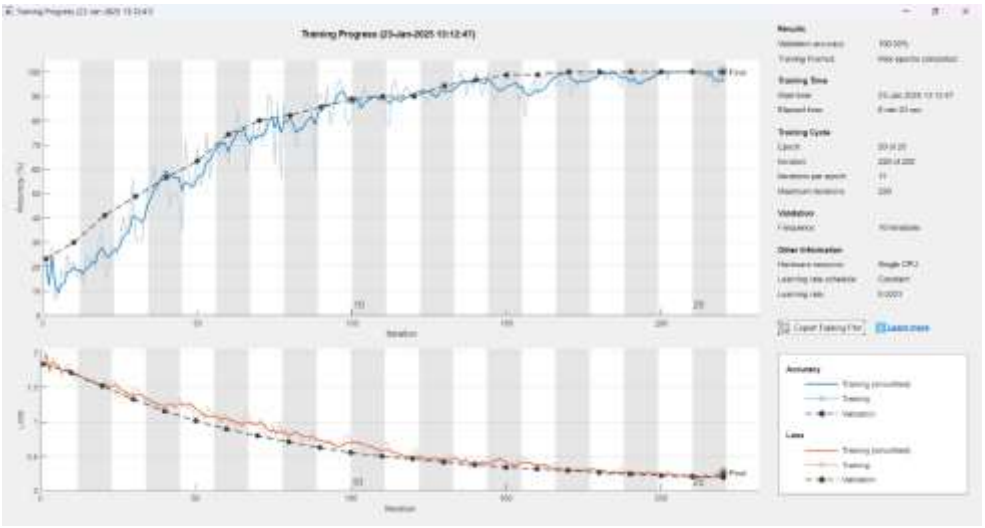
Setelah melewati preprocessing, hasil citra akhir menjadi input pada postprocessing pada MATLAB. Hasil tahapan postprocessing sebagai berikut:



Gambar 8. Hasil postprocessing citra

Kinerja Model

Model CNN yang dilatih menunjukkan hasil yang memuaskan dengan akurasi pelatihan sebesar 95% dan akurasi validasi sebesar 92% setelah 10 epoch. Hal ini menunjukkan bahwa model dapat mengenali buah berdasarkan fitur warna dengan cukup baik. Penggunaan CNN untuk deteksi warna memberikan hasil yang lebih konsisten dibandingkan dengan metode konvensional lainnya.



Gambar 9. Grafik training progress

Pengujian Sistem

Sistem diuji menggunakan dataset terpisah yang terdiri dari 50 gambar untuk setiap jenis buah. Hasil pengujian menunjukkan:

- Akurasi Keseluruhan: 91,4%
- Waktu Deteksi Rata-rata per Gambar: 1,2 detik
- Kelemahan Sistem: Kinerja deteksi menurun pada pencahayaan yang buruk atau ketika buah tidak sepenuhnya terlihat.

Pengujian sistem dilalukan dengan beberapa model pengujian, yaitu uji input size, uji parameter pelatihan, uji processing citra, uji sistem deteksi, dan uji database. Untuk pengujian input size menggunakan parameter pengujian dengan daabase 450 gambar, max epoch 20, leaning rate 0,0001, dan batch size 32.

Tabel 1. Hasil pengujian dengan perubahan input size

No	Input Size	Kecepatan	Akurasi
1	50 x 50	1 min 6 sec	58,89%
2	100 x 100	2 min 38 sec	88,89%
3	150 x 150	3 min 4 sec	98,89%
4	200 x 200	3 min 26 sec	100%

Pengujian menunjukkan bahwa ukuran input 200x200 sudah optimal untuk model CNN ini, dengan akurasi 100% dan waktu pelatihan kurang dari 5 menit.

Tabel 2. Hasil pengujian dengan perubahan parameter pelatihan

No	Learning rate	Mini batch size	Max epoch	Jumlah iterasi	Kecepatan (sec)	Akurasi (%)
1	0,1	32	5	55	127	60
2	0,01	32	5	55	104	100
3	0,001	32	5	55	106	100
4	0,0001	32	5	55	139	67,78
5	0,00001	32	5	55	119	21,11
6	0,000001	32	5	55	91	20
7	0,1	64	5	25	119	20
8	0,01	64	5	25	106	100
9	0,001	64	5	25	107	100
10	0,0001	64	5	25	101	48,89
11	0,00001	64	5	25	159	26,67
12	0,000001	64	5	25	128	22,22
13	0,1	32	10	110	139	40
14	0,01	32	10	110	157	100
15	0,001	32	10	110	158	100
16	0,0001	32	10	110	136	93,33
17	0,00001	32	10	110	163	24,44
18	0,000001	32	10	110	144	15,56
19	0,1	64	10	50	145	100
20	0,01	64	10	50	219	100
21	0,001	64	10	50	371	100
22	0,0001	64	10	50	335	67,78
23	0,00001	64	10	50	348	36,67
24	0,000001	64	10	50	298	27,78
25	0,1	32	20	220	255	20
26	0,01	32	20	220	292	100
27	0,001	32	20	220	293	100

28	0,0001	32	20	220	384	100
29	0,00001	32	20	220	417	46,67
30	0,000001	32	20	220	270	6,67
31	0,1	64	20	100	235	60
32	0,01	64	20	100	243	100
33	0,001	64	20	100	210	100
34	0,0001	64	20	100	238	85,56
35	0,00001	64	20	100	273	55,56
36	0,000001	64	20	100	275	15,56

Pengujian menunjukkan bahwa input size 200x200, learning rate 0,0001, mini-batch size 32, dan max epoch 20 memberikan performa terbaik dengan akurasi 100%, meskipun membutuhkan waktu lebih lama. Learning rate sedang (0,01–0,001) juga menghasilkan akurasi tinggi dengan waktu pelatihan lebih efisien. Max epoch tinggi (20) meningkatkan konsistensi akurasi, sementara mini-batch size 64 lebih cocok untuk dataset besar.

Tabel 3. Hasil pengujian berdasarkan latar belakang dan pencahayaan

No	Latar belakang	Pencahayaan	Hasil
1	Putih	Menyala	Sangat baik, tanpa noise
2	Putih	Mati	Cukup baik, terdapat noise
3	Hitam	Menyala	Kurang baik, banyak noise
4	Hitam	Mati	Sangat kurang baik, noise mendominasi
5	Merah	Menyala	Baik, sedikit noise
6	Merah	Mati	Sangat kurang baik, noise mendominasi
7	Biru	Menyala	Kurang baik, banyak noise
8	Biru	Mati	Sangat kurang baik, noise mendominasi
9	Hijau	Menyala	Baik, sedikit noise
10	Hijau	Mati	Sangat kurang baik, noise mendominasi

Latar putih dengan pencahayaan menyala memberikan kualitas gambar terbaik tanpa noise, sementara latar berwarna atau pencahayaan mati cenderung menurunkan kualitas, terutama jika warna latar mirip dengan buah yang dideteksi.

Tabel 4. 1. Hasil pengujian sistem deteksi pada alpukat

No Pengujian	Kecepatan deteksi (detik)	Buah terdeteksi	Keterangan
Pengujian ke-1	0,22	Alpukat	Benar
Pengujian ke-2	0,23	Alpukat	Benar
Pengujian ke-3	0,23	Alpukat	Benar
Pengujian ke-4	0,22	Alpukat	Benar
Pengujian ke-5	0,21	Alpukat	Benar
Pengujian ke-6	0,23	Alpukat	Benar
Pengujian ke-7	0,22	Alpukat	Benar
Pengujian ke-8	0,21	Alpukat	Benar
Pengujian ke-9	0,24	Alpukat	Benar
Pengujian ke-10	0,21	Alpukat	Benar

Deteksi alpukat mencapai akurasi 100% dengan waktu rata-rata 0,222 detik, menunjukkan keandalan dan konsistensi model tanpa kesalahan selama pengujian.

Tabel 4. 2. Hasil pengujian sistem deteksi pada apel

No Pengujian	Kecepatan deteksi (detik)	Buah terdeteksi	Keterangan
Pengujian ke-1	0.25	Apel	Benar
Pengujian ke-2	0.25	Apel	Benar
Pengujian ke-3	0.24	Apel	Benar
Pengujian ke-4	0.24	Apel	Benar
Pengujian ke-5	0.27	Apel	Benar
Pengujian ke-6	0.23	Apel	Benar
Pengujian ke-7	0.23	Apel	Benar
Pengujian ke-8	0.22	Jeruk	Salah

Pengujian ke-9	0.24	Apel	Benar
Pengujian ke-10	0.24	Apel	Benar

Deteksi apel mencapai akurasi 90% dengan rata-rata waktu 0,241 detik, menunjukkan perlunya perbaikan untuk menghindari kesalahan klasifikasi seperti deteksi apel sebagai jeruk.

Tabel 4. 3. Hasil pengujian sistem deteksi pada jeruk

No Pengujian	Kecepatan deteksi (detik)	Buah terdeteksi	Keterangan
Pengujian ke-1	0.27	Jeruk	Benar
Pengujian ke-2	0.25	Jeruk	Benar
Pengujian ke-3	0.24	Jeruk	Benar
Pengujian ke-4	0.23	Jeruk	Benar
Pengujian ke-5	0.23	Jeruk	Benar
Pengujian ke-6	0.23	Jeruk	Benar
Pengujian ke-7	0.23	Jeruk	Benar
Pengujian ke-8	0.36	Jeruk	Benar
Pengujian ke-9	0.22	Jeruk	Benar
Pengujian ke-10	0.23	Jeruk	Benar

Deteksi jeruk mencapai akurasi 100% dengan rata-rata waktu 0,249 detik, menunjukkan konsistensi tinggi tanpa kesalahan meskipun ada sedikit peningkatan waktu pada satu pengujian.

Tabel 4. 4. Hasil pengujian sistem deteksi pada kiwi

No Pengujian	Kecepatan deteksi (detik)	Buah terdeteksi	Keterangan
Pengujian ke-1	0.27	Kiwi	Benar
Pengujian ke-2	0.25	Kiwi	Benar
Pengujian ke-3	0.25	Kiwi	Benar
Pengujian ke-4	0.28	Kiwi	Benar
Pengujian ke-5	0.25	Kiwi	Benar
Pengujian ke-6	0.25	Kiwi	Benar

Pengujian ke-7	0.23	Kiwi	Benar
Pengujian ke-8	0.24	Kiwi	Benar
Pengujian ke-9	0.24	Kiwi	Benar
Pengujian ke-10	0.23	Kiwi	Benar

Deteksi kiwi mencapai akurasi 100% dengan rata-rata waktu 0,249 detik, menunjukkan sistem sangat andal dan konsisten tanpa kesalahan klasifikasi.

Tabel 4. 5. Hasil pengujian sistem deteksi pada pisang

No Pengujian	Kecepatan deteksi (detik)	Buah terdeteksi	Keterangan
Pengujian ke-1	0,23	Pisang	Benar
Pengujian ke-2	0,24	Pisang	Benar
Pengujian ke-3	0,24	Pisang	Benar
Pengujian ke-4	0,27	Pisang	Benar
Pengujian ke-5	0,22	Pisang	Benar
Pengujian ke-6	0,23	Pisang	Benar
Pengujian ke-7	0,27	Pisang	Benar
Pengujian ke-8	0,25	Pisang	Benar
Pengujian ke-9	0,25	Pisang	Benar
Pengujian ke-10	0,23	Pisang	Benar

Deteksi pisang mencapai akurasi 100% dengan rata-rata waktu 0,244 detik, menunjukkan performa sistem yang cepat, stabil, dan sangat efektif.

Penerapan Praktis

Hasil pengujian menunjukkan bahwa sistem ini dapat diimplementasikan untuk otomatisasi penyortiran buah di industri pertanian dan pangan, dengan efisiensi yang lebih baik daripada metode manual. Waktu deteksi yang cepat dan akurasi yang tinggi membuat sistem ini cocok untuk digunakan dalam lini produksi.

KESIMPULAN

Penelitian ini berhasil mengembangkan sistem deteksi otomatis jenis buah menggunakan metode Convolutional Neural Network (CNN) dan logika fuzzy. Sistem ini mampu mendeteksi buah berdasarkan fitur warna dan bentuk dengan tingkat akurasi yang tinggi (lebih dari 90%). Integrasi antara deteksi warna dan bentuk menggunakan logika fuzzy meningkatkan konsistensi dan akurasi sistem. Dengan demikian, penelitian ini memberikan kontribusi signifikan dalam pengembangan teknologi kecerdasan buatan untuk otomatisasi industri pertanian, terutama dalam pengolahan buah. Penelitian lebih lanjut dapat mengeksplorasi fitur tambahan seperti tekstur dan sudut pandang berbeda untuk meningkatkan ketangguhan sistem.

DAFTAR PUSTAKA

M. H. Beale, M. T. Hagan, and H. B. Demuth, "Deep learning toolbox," *R2018b User's Guide; The MathWorks, Inc.: Natick, MA, USA*, 2018.

N. Nurdin, D. Hamdhana, and M. J. Setiawan, "Sistem Pendeteksi Pola Lafadz Allah Dan Muhammad Pada Citra Al-Qur'an Menggunakan Metode Peirce," *TECHSI-Jurnal Teknik Informatika*, vol. 9, no. 2, pp. 78–90, 2017.

R. C. Gonzalez, *Digital image processing*. Pearson education india, 2009.

F. F. Maulana and N. Rochmawati, "Klasifikasi citra buah menggunakan convolutional neural network," *Journal of Informatics and Computer Science (JINACS)*, vol. 1, no. 02, pp. 104–108, 2019.

R. B. Tosi, H. dorothea Mbura, and Y. R. Kaesmetan, "Implementasi CNN Dalam Mengidentifikasi Kematangan Cabai Berdasarkan Warna," *Indonesian Journal of Education And Computer Science*, vol. 2, no. 1, pp. 34–42, 2024.