



Implementasi Metode Backpropagation Neural Network untuk Mendeteksi Kualitas Cabai Berdasarkan Citra Digital

Tugas Akhir

**Oleh:
Christoper Modji (4212201095)**

**Teknik Mekatronika
Jurusan Teknik Elektro
Politeknik Negeri Batam
2025**

Pernyataan Keaslian Tugas Akhir

Saya yang bertandatangan dibawah ini menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya yang berjudul : "Implementasi Metode Backpropagation Neural Network untuk Mendeteksi Kualitas Cabai Berdasarkan Citra Digital" adalah hasil karya sendiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan, dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri. Semua referensi yang dikutip atau dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan saya ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Batam, 5 juli 2026



Christoper Modji
NIM: 4212201095

Lembar Pengesahan

Tugas Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar

Sarjana Terapan Teknik (S.Tr.T)

di

Politeknik Negeri Batam

Disusun oleh:

Christoper Modji (4212201095)

Tanggal Sidang: 6 Juli 2026

Disetujui oleh :



**1. Daniel Sutopo Pamungkas,
S.T., Ph.D
NIK:100006**



**1. ADLIAN JEFIZA, S.Pd., M.T.
NIK: 199112022019031016**

**2. Dr. Abdurahman Dwijotomo,
S.ST, M
NIK: 122257**



**2. Fadli Firdaus, M.Pd
NIK: 122271**

Implementasi Metode Backpropagation Neural Network untuk Mendeteksi Kualitas Cabai Berdasarkan Citra Digital

Abstrak

Klasifikasi kondisi cabai merupakan salah satu aspek penting dalam proses seleksi bahan baku pada sektor pertanian dan industri kuliner. Penilaian kondisi cabai yang dilakukan secara manual cenderung bersifat subjektif dan kurang konsisten sehingga diperlukan sistem klasifikasi otomatis yang mampu memberikan hasil secara cepat dan akurat. Penelitian ini bertujuan mengembangkan sistem klasifikasi kondisi cabai berbasis citra digital menggunakan metode Backpropagation Neural Network (BPNN). Sistem yang dikembangkan memanfaatkan YOLO (You Only Look Once) untuk mendeteksi dan melokalisasi objek cabai sehingga objek dapat dipisahkan dari latar belakang melalui proses *bounding box*. Selanjutnya, citra hasil deteksi diproses melalui tahapan *preprocessing* yang meliputi *resizing* berukuran 224×224 piksel, segmentasi, penghilangan *noise* menggunakan median filter, serta konversi ruang warna ke HSV dan grayscale. Tahap berikutnya adalah ekstraksi fitur warna dan tekstur yang menghasilkan 10 fitur, kemudian dinormalisasi menggunakan metode Min-Max Scaling sebelum digunakan sebagai masukan model BPNN. Model dilatih menggunakan algoritma *backpropagation* dengan variasi jumlah *epoch* dan jumlah neuron pada *hidden layer* untuk memperoleh konfigurasi terbaik. Hasil penelitian menunjukkan bahwa model BPNN memperoleh akurasi klasifikasi sebesar 97% dalam membedakan cabai ke dalam tiga kelas, yaitu mentah, matang, dan busuk. Pada pengujian implementasi secara *realtime*, sistem berhasil melakukan penyortiran 106 dari 118 cabai dengan tingkat keberhasilan sebesar 89%. Hasil tersebut menunjukkan bahwa metode BPNN mampu mengenali pola fitur warna dan tekstur cabai secara efektif serta berpotensi diterapkan sebagai sistem pendukung penyortiran cabai secara otomatis.

Kata kunci: Klasifikasi cabai, YOLO, citra digital, Backpropagation, jaringan syaraf tiruan, kematangan, ekstraksi fitur

Implementation of Backpropagation Neural Network Method to Detect Chili Quality Based on Digital Images

Abstract

Classification of chili conditions is an important aspect in the selection of raw materials in the agricultural and culinary industries. Manual assessment of chili conditions tends to be subjective and inconsistent; therefore, an automated classification system is required to provide fast and accurate results. This study aims to develop a digital image-based chili condition classification system using the Backpropagation Neural Network (BPNN) method. The proposed system utilizes YOLO (You Only Look Once) to detect and localize chili objects, enabling the objects to be separated from the background through a bounding box process. Subsequently, the detected images undergo several preprocessing stages, including resizing to 224×224 pixels, segmentation, noise removal using a median filter, and color space conversion to HSV and grayscale. The next stage involves extracting color and texture features, resulting in ten features, which are then normalized using the Min-Max Scaling method before being used as inputs to the BPNN model. The model is trained using the backpropagation algorithm with variations in the number of epochs and hidden layer neurons to obtain the optimal configuration. The experimental results show that the BPNN model achieved a classification accuracy of 97% in distinguishing chili images into three classes: unripe, ripe, and rotten. In the real-time implementation test, the system successfully sorted 106 out of 118 chili samples, achieving an overall success rate of 89%. These results demonstrate that the BPNN method is capable of effectively recognizing color and texture patterns of chili images and has the potential to be implemented as an automated chili sorting support system

Keywords: Chili classification, YOLOv8 , digital image, Backpropagation, artificial neural network, feature extraction

Kata Pengantar

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Laporan Tugas Akhir ini dengan baik. Laporan Tugas Akhir ini disusun sebagai salah satu syarat untuk menyelesaikan pendidikan jenjang Diploma IV pada Program Studi Teknik Mekatronika, Politeknik Negeri Batam.

Tugas akhir ini membahas pengembangan sistem klasifikasi kualitas cabai berbasis citra digital dengan mengimplementasikan metode Backpropagation Neural Network (BPNN) serta memanfaatkan YOLO sebagai metode deteksi objek. Melalui penelitian ini diharapkan dapat dihasilkan sistem yang mampu melakukan proses klasifikasi kualitas cabai secara otomatis, cepat, objektif, dan akurat sehingga dapat mendukung proses sortasi pada sektor pertanian maupun industri pengolahan cabai.

Dalam penyusunan Laporan Tugas Akhir ini, penulis menyadari bahwa keberhasilan penyelesaian laporan ini tidak terlepas dari bantuan, bimbingan, dan dukungan dari berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Tuhan Yang Maha Esa yang senantiasa memberikan kesehatan, kekuatan, dan penyertaan kepada penulis.
2. Kedua orang tua yang selalu memberikan doa, dukungan moral maupun material selama proses perkuliahan hingga penyusunan Tugas Akhir ini
3. Bapak Diono, S.T., M.Eng, selaku Kepala Program Studi Teknik Mekatronika.
4. Bapak Indra Hardian Mulyadi, S.T., M.Eng, selaku Kepala Jurusan Teknik Elektro
5. Bapak Adlian Jefiza, S.Pd, M.T, dan bapak Fadli Firdaus, S.Pd selaku dosen pembimbing tugas akhir yang telah memberikan arahan dan bimbingan selama proses pengerjaan laporan akhir ini.
6. Seluruh dosen Teknik Elektro yang telah memberikan ilmu, pengalaman, dan bimbingan selama masa perkuliahan
7. Teman-teman Program Studi Teknik Mekatronika Politeknik Negeri Batam angkatan 2022 yang telah memberikan semangat, bantuan dan kebersamaan selama proses kuliah
8. Teman-teman saya Ezi, Ariel, Arya, Lamba, Lojo, Gery, Trison, Ikel, dan Putra yang telah banyak memberikan bantuan selama proses penyelesaian Tugas Akhir ini

Batam, 5 Juli 2026



Christoper Modji

Daftar Isi

Pernyataan Keaslian Tugas Akhir	i
Lembar Pengesahan.....	ii
Abstrak.....	iii
<i>Abstract</i>	iv
Kata Pengantar	v
Daftar Isi.....	vi
Daftar Gambar.....	ix
Daftar Tabel.....	ix
Bab 1. Pendahuluan	1
1.1.Latar Belakang.....	1
1.2.Rumusan Masalah	2
1.3. Tujuan	2
1.4. Manfaat.....	2
1.5. Batasan	3
Bab 2. Tinjauan Pustaka	4
2.1.Pengolahan Citra Digital.....	4
2.2 Preprocessing Data	4
2.2.1 Resizing.....	4
2.2.2 Segmentasi	4
2.2.3 Noise Removal	5
2.2.4 Konversi Ruang Warna	5
2.3. Cabai	6
2.4. Jaringan Syaraf Tiruan (JST)	6
2.5 YOLO (You Only Look Once)	6
2.5. Backpropagation Neural Network.....	7
2.5.1 Struktur dan Alur Kerja BPNN	8
2.5.2 Mekanisme Backpropagation.....	8
2.5.3 Fungsi Aktivasi.....	9
2.5.4 Penentuan Arsitektur dan Hyperparameter.....	9
2.6. Penelitian Terkait.....	9

Bab 3. Metodologi Penelitian	12
3.1 Perancangan Penelitian	12
3.1.1 Studi Literatur	13
3.1.2 Pengumpulan Data	13
3.1.3 Perancangan Sistem	14
3.1.3.1 Input citra cabai	15
3.1.3.2 Preprocessing Data	15
3.1.3.3 Ekstraksi Fitur	15
3.1.3.4 Normalisasi Data	17
3.1.3.5 Pelatihan Model BPNN	17
3.1.3.6 Output Sistem	18
3.1.4 Evaluasi Model dan Analisis Hasil	18
3.1.5 Implementasi Sistem	19
3.1.6 Kesimpulan	19
3.2. Alat dan Bahan	19
3.3 Perancangan Alat	20
3.4. Pengujian Sistem	20
3.4.1 Pengujian Hasil Klasifikasi Model	20
3.4.2 Pengujian Implementasi Sistem	21
3.4.3 Pengujian Variasi Jumlah Epoch	21
3.4.4 Pengujian Variasi Jumlah Hidden Layer	21
3.4.5 Pengujian Metode Cross Validation	21
Bab 4. Hasil dan Pembahasan	23
4.1 Pengumpulan Data	23
4.2 Preprocessing	24
4.2.1 Resize Image	24
4.2.2 Segmentasi	24
4.2.3 Filtering (Noise Reduction)	26
4.2.4 Konversi Ruang Warna	26
4.3 Ekstraksi Fitur	27
4.3.1 Fitur Warna (HSV)	27
4.3.2 Fitur Warna (RGB Ratio)	28

4.3.3 Fitur Tekstur.....	28
4.3.4 Normalisasi Data	29
4.4 Pelatihan Model BPNN.....	29
4.4.1 Arsitektur Jaringan	29
4.4.2 Inisialisasi Bobot.....	30
4.4.3 Proses Forward Propagation	30
4.4.4 Proses Backpropagation	31
4.4.5 Loss Function	31
4.5 Implementasi Sistem GUI	32
4.5.1 Hasil Pengujian Mode Upload	33
4.5.2 Pengujian Realtime Mode.....	33
4.6 Evaluasi Model dan Hasil.....	34
4.6.1 Hasil Klasifikasi Model	35
4.6.2 Pengujian Implementasi Sistem	37
4.6.3 Pengujian Variasi Jumlah Epoch	41
4.6.4 Pengujian Variasi Jumlah Hidden Neurons.....	42
4.6.5 Evaluasi Metode Cross Validation	43
BAB 5. Kesimpulan dan Saran	44
5.1 Kesimpulan.....	44
5.2 Saran.....	45
Daftar Pustaka	46
Biodata.....	49
Lampiran	50

Daftar Gambar

Gambar 1. Arsitektur ANN pada Backpropagation.....	8
Gambar 2. Flowchart Perancangan Penelitian.....	12
Gambar 3. Flowchart Perancangan Sistem.....	14
Gambar 4. Arsitektur Backpropagation Neural Network.....	18
Gambar 5. Desain Alat.....	20
Gambar 6. Pengumpulan Cabai	23
Gambar 7. Cabai Setelah di Resize	24
Gambar 8. Cabai Sebelum di Resize	24
Gambar 9. Cabai Sebelum di Segmentasi	25
Gambar 10. Cabai Setelah di Segmentasi	25
Gambar 11. Contoh Segmenetasi Gagal	26
Gambar 12. Cabai Setelah Filtering.....	26
Gambar 13. Cabai Sebelum Filtering	26
Gambar 14. Hasil konversi Grayscale dan HSV.....	27
Gambar 15. Tampilan GUI	32
Gambar 16. Hasil Deteksi YOLO	32
Gambar 17 GUI Upload Image	33
Gambar 18. GUI Mode Realtime.....	34
Gambar 19. Confusion Matrix.....	35
Gambar 20. Grafik Kurva Loss dan Accuracy.....	42

Daftar Tabel

Tabel 1. Tabel Penelitian Terkait.....	11
Tabel 2. Estimasi Biaya	19
Tabel 3. Hasil Beberapa Features Setelah di Ekstrasi	27
Tabel 4. Rata-rata Fitur Warna HSV	27
Tabel 5. Rata-rata Fitur RGB Ratio	28
Tabel 6. Rata-rata Fitur Tekstur	28
Tabel 7. Data Fitur Setelah di Normalisasi	29
Tabel 8. Hasil Akurasi pada Setiap Kelas Cabai	35
Tabel 9. Analisis Kesalahan Klasifikasi Model.....	36
Tabel 10. Pengujian Upload Mode	38
Tabel 11. Pengujian Implementasi pada Realtime Mode	40
Tabel 12. Hasil Pengujian Variasi Jumlah Epoch.....	41
Tabel 13. Hasil Pengujian Variasi Jumlah Hidden Layer	42
Tabel 14. Hasil Pengujian Menggunakan Cross Validation.....	43

Bab 1. Pendahuluan

1.1. Latar Belakang

Cabai (*Capsicum annuum* L.) merupakan salah satu komoditas pertanian strategis di Indonesia karena banyak digunakan dalam masakan sehari-hari dan industri makanan, serta menjadi sumber pendapatan penting bagi petani lokal. Tahap identifikasi tingkat kematangan cabai merupakan aspek penting dalam budidaya dan penanganan pasca panen. Hal ini disebabkan karena tingkat kematangan secara signifikan mempengaruhi kualitas, kandungan nutrisi, dan nilai pasar cabai[1]. Namun menentukan tingkat kematangan cabai secara akurat menjadi tantangan karena berbagai faktor seperti suhu, penyinaran dan penanganan pasca panen [2]. Di lapangan, petani sering menghadapi sejumlah tantangan yang memengaruhi kualitas cabai. Cabai merah mudah rusak karena kadar air yang sangat tinggi dan penanganan pasca panen yang masih tradisional. Selain itu, tanaman juga rentan terhadap serangan hama dan penyakit seperti patek dan kutu kebul, terutama saat musim hujan atau kekeringan. Hal ini berdampak pada menurunnya produktivitas dan kualitas cabai di tingkat petani. Realita di lapangan menunjukkan bahwa penanganan pasca panen cabai masih belum optimal. Menurut laporan ASEAN, kerugian pasca panen cabai nasional mencapai 30 hingga 40 %, dan baru dapat ditekan hingga di bawah 15 % dengan penerapan teknologi tepat guna. Hal ini menunjukkan tingginya potensi kerusakan akibat kesalahan sortir kualitas cabai sejauh penanganan masih mengandalkan cara manual dan tradisional.

Dengan berkembangnya teknologi pengolahan citra digital, klasifikasi objek berbasis visual menjadi lebih mudah dan akurat. Perkembangan teknologi pengolahan citra digital telah membuka peluang signifikan dalam klasifikasi objek berbasis visual. Kemajuan algoritma syaraf tiruan, khususnya pada Backpropagation Neural Network, memungkinkan proses klasifikasi visual dengan akurasi yang tinggi[2]. Khususnya pada cabai, teknik seperti ekstraksi fitur warna, tekstur, dan bentuk memungkinkan evaluasi kematangan atau kualitas secara lebih objektif dan cepat. Untuk melakukan klasifikasi otomatis, metode Jaringan Syaraf Tiruan dengan algoritma backpropagation menunjukkan hasil yang menjanjikan.

Proses pemilihan bahan baku cabai pada usaha makanan rumahan seperti bakso mercon masih sering dilakukan secara manual tanpa adanya penyortiran atau pemilihan cabai yang jelas. Berdasarkan pengalaman pribadi saat membantu usaha milik saudara yang bergerak dibidang kuliner tersebut, ditemukan bahwa semua cabai yang digunakan yang diambil dari penyimpanan baik itu matang ataupun busuk, semua langsung dicampur dan diolah menjadi kuah bakso mercon tanpa melalui proses sortir terlebih dahulu. Hal ini sangat beresiko menurunkan kualitas rasa dan aroma sambal yang dihasilkan dan dapat berpengaruh pada kesehatan konsumen.

Penelitian terkini menunjukkan bahwa pengolahan citra digital yang dikombinasikan dengan kecerdasan buatan mampu meningkatkan akurasi klasifikasi objek pertanian, termasuk indentifikasi tingkat kematangan maupun kualitas cabai secara otomatis[3]. Beberapa penelitian di Indonesia juga telah membuktikan bahwa metode serupa cukup efektif untuk mengklasifikasi kematangan cabai. Contohnya, penelitian oleh Hardiansah yang menggunakan metode BPNN untuk klasifikasi citra daun herbal berdasarkan fitur bentuk dan memperoleh akurasi yang tinggi yaitu 88,8% [4]. Juga, penelitian yang dilakukan Ajib Susanto menerapkan metode BPPN untuk mengklasifikasi kualitas buah apel berdasarkan warna RGB. Hasil yang diperoleh cukup baik dengan akurasi 93% dan menunjukkan bahwa BPNN dapat diterapkan untuk citra buah lainnya[5]. Berdasarkan penelitian-penelitian tersebut, penggunaan metode BPNN sudah cukup relevan untuk mengklasifikasikan tingkat kematangan cabai

Tujuan dari penelitian ini adalah merancang dan mengembangkan sistem klasifikasi tingkat kematangan cabai berbasis citra digital menggunakan metode BPNN. Sistem ini diharapkan dapat membantu petani dan masyarakat dalam melakukan sortasi tingkat kematangan secara lebih objektif, konsisten, dan cepat, sehingga meningkatkan efisiensi dan nilai jual hasil panen.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka rumusan masalah pada penelitian ini adalah sebagai berikut:

1. Bagaimana membuat sistem klasifikasi kualitas cabai menggunakan metode Backpropagation Neural Network?
2. Seberapa akurat metode BPNN dalam mengklasifikasi kualitas cabai ?

1.3. Tujuan

Dari rumusan masalah diatas, maka tujuan untuk Tugas akhir ini adalah:

1. Mengembangkan sistem klasifikasi otomatis kualitas cabai menggunakan metode BPNN.
2. Mengukur akurasi sistem klasifikasi berbasis BPNN.
3. Menganalisis di sistem klasifikasi pada confusion matrix dan evaluasi kinerja pelatihan

1.4. Manfaat

Penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan teknologi klasifikasi otomatis berbasis citra digital, khususnya dalam bidang pertanian modern. Selain itu, dapat menjadi referensi dalam implementasi metode BPNN untuk klasifikasi objek berbasis fitur warna dan tekstur.

1.5. Batasan

Agar Tugas akhir ini lebih terarah dan sesuai, maka harus dibatasi dengan hal-hal berikut:

1. Kategori kualitas dibatasi menjadi tiga kelas yaitu mentah, matang, dan buruk.
2. Pengambilan citra dilakukan dengan latar belakang netral dan pencahayaan terkendali.
3. Penelitian ini tidak membandingkan metode BPNN dengan algoritma klasifikasi lainnya
4. Pengujian dilakukan secara individual untuk setiap data uji, yaitu satu per satu untuk setiap file citra

Bab 2. Tinjauan Pustaka

2.1. Pengolahan Citra Digital

Pengolahan citra digital adalah teknik untuk memproses gambar digital menggunakan komputer agar kualitas gambar meningkat atau untuk mengekstrak informasi yang relevan dari citra tersebut. Teknik ini sangat bermanfaat dalam berbagai aplikasi seperti deteksi objek, klasifikasi, dan pengenalan pola. Dalam konteks klasifikasi buah cabai, pengolahan citra digunakan untuk mengekstraksi fitur visual seperti warna, tekstur, dan bentuk yang menjadi indikator kualitas cabai[6]. Pengolahan citra digital tidak hanya digunakan untuk meningkatkan kualitas visual suatu citra, tetapi juga untuk mengekstraksi informasi penting yang dapat dimanfaatkan dalam proses identifikasi, klasifikasi dan pengambilan keputusan secara otomatis pada berbagai bidang termasuk pertanian[7].

2.2 Preprocessing Data

Preprocessing data merupakan Langkah untuk menyiapkan gambar agar siap diolah sebelum masuk ke tahap klasifikasi dan ekstraksi fitur. Tahap ini bertujuan untuk meningkatkan kualitas citra, mengurangi noise, dan memastikan input yang dimasukkan ke dalam model klasifikasi konsisten. Berikut adalah tahapan dari preprocessing data yang diterapkan dalam penelitian ini:

2.2.1 Resizing

Tahapan pertama dalam preprocessing adalah melakukan resizing terhadap citra. Tahap ini bertujuan untuk menyeragamkan ukuran semua citra yang akan di analisis. Dalam penelitian ini ukuran citra disesuaikan menjadi 224 x 224 piksel, karena penelitian sebelumnya menunjukkan bahwa ukuran 224 x 224 piksel mampu memberikan hasil presisi dan F1-Score lebih baik dalam klasifikasi kualitas citra cabai[8]

2.2.2 Segmentasi

Tahap berikutnya adalah segmentasi, yaitu pemisahan objek utama dari latar belakang. Proses ini dilakukan dengan menganalisis nilai warna pada setiap piksel sehingga citra dapat dibagi menjadi beberapa bagian sesuai dengan fitur yang ingin diidentifikasi[9]. Segmentasi merupakan tahapan penting dalam pengolahan citra karena keberhasilan proses ini akan menentukan kualitas fitur yang di ekstraksi pada tahap berikutnya. Objek yang tersegmentasi dengan baik akan menghasilkan informasi warna dan tekstur yang lebih representatif sehingga meningkatkan performa proses klasifikasi[10]. Teknik segmentasi yang digunakan adalah thresholding berbasis warna menggunakan grayscale dan HSV. Segmentasi berbasis grayscale digunakan karena metode ini cocok untuk citra dengan latar

belakang polos dan HSV digunakan untuk memperjelas bagian warna yang menunjukkan Tingkat kematangan buah [11]

2.2.3 Noise Removal

Setelah tahap segmentasi dilakukan, tahap selanjutnya adalah penghilangan noise yang dapat mengganggu saat ekstraksi fitur. Noise pada citra dapat disebabkan oleh berbagai faktor, seperti kondisi pencahayaan yang kurang baik, kualitas kamera, dan jarak pengambilan gambar. Keberadaan noise dapat mengurangi kualitas citra sehingga diperlukan proses *noise removal* sebelum dilakukan konversi ruang warna agar informasi visual menjadi lebih akurat[12]. Noise yang muncul berupa bintik-bintik atau gangguan lain akibat kualitas kamera atau pencahayaan. Untuk mengatasi hal ini maka digunakan metode Median Filter. Median filter bekerja dengan mengganti nilai piksel yang terganggu dengan nilai median dari piksel sekitarnya, yang terbukti efektif menghilangkan noise tanpa merusak tepi objek[13].

2.2.4 Konversi Ruang Warna

Tahap akhir adalah konversi ruang warna, yaitu mengubah warna dari RGB ke dua ruang warna yang berbeda yaitu grayscale dan HSV. Konversi ke grayscale dilakukan untuk menekankan informasi tekstur dan intensitas cahaya dalam citra. Citra grayscale dapat membantu mendeteksi pola kontras antara bagian terang dan gelap yang penting dalam membedakan kondisi cabai, seperti bercak hitam atau bagian yang busuk. Berikut adalah rumus standar konversi RGB ke grayscale.

$$Gray = 0.229 \times R + 0.587 \times G + 0.114 \times B \quad [11]$$

Sementara itu, konversi ke *Hue, Saturation, Value* (HSV) digunakan untuk memberikan informasi warna dari cabai, yang sangat relevan untuk mengidentifikasi tingkat kematangan. Hue menunjukkan jenis warna, saturation menunjukkan kejenuhan warna, dan value menunjukkan kecerahan. HSV dianggap lebih stabil dalam kondisi pencahayaan yang berubah, sehingga warna seperti merah matang, hijau mentah, ataupun merah gelap busuk dapat terdeteksi dengan baik. Ruang warna HSV banyak digunakan pada sistem klasifikasi citra karena mampu merepresentasikan informasi warna secara lebih stabil terhadap perubahan intensitas pencahayaan. Selain itu, kombinasi fitur HSV dan tekstur mampu meningkatkan performa klasifikasi objek pada citra digital[14]. Berikut adalah perhitungan matematis perubahan warna RGB ke HSV :

$$V = \frac{\max(R, G, B)}{255} \times 100$$

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \times 100$$

$$H = 60 \times \frac{G - B}{\max - \min} \text{ , jika maks = R}$$

$$H = 60 \times \left(2 + \frac{B - R}{maks - min} \right), \text{ jika maks} = G$$

$$H = 60 \times \left(4 + \frac{R - G}{maks - min} \right), \text{ jika maks} = B[11]$$

2.3. Cabai

Tanaman cabai merupakan salah satu anggota *genus Capsicum* (*Capsicum annum* L.) yang termasuk dalam keluarga *Solanaceae* atau suku terong-terongan. Cabai dikenal sebagai tanaman yang mudah dibudidayakan, baik di dataran rendah maupun dataran tinggi. Selain itu, cabai mengandung berbagai senyawa bermanfaat bagi kesehatan, seperti vitamin A, vitamin C, serta minyak atsiri *capsaicin* yang memberikan rasa pedas khas pada cabai [6].

Setelah dipanen, buah cabai akan mengalami berbagai perubahan yang dipengaruhi oleh faktor fisiologis, fisika, kimia, dan mikrobiologis. Perubahan tersebut menyebabkan penurunan mutu buah sehingga diperlukan penanganan pascapanen yang tepat untuk mempertahankan kualitas cabai selama proses penyimpanan[15]. Perubahan mutu tersebut dapat dikenali melalui karakteristik visual, seperti perubahan warna kulit, tingkat kesegaran, tekstur permukaan, maupun munculnya kerusakan pada buah. Karakteristik visual tersebut banyak dimanfaatkan sebagai informasi dalam sistem pengolahan citra warna dan teksturnya.

2.4. Jaringan Syaraf Tiruan (JST)

Jaringan syaraf tiruan (JST) merupakan model matematis yang terinspirasi oleh struktur jaringan saraf biologis dalam otak manusia dan memiliki kemampuan untuk mengenali pola serta relasi kompleks dalam data. Model ini terdiri atas lapisan input, satu atau lebih hidden layer, dan lapisan output. Proses pembelajaran pada JST dilakukan dengan menyesuaikan bobot antar neuron menggunakan algoritma tertentu, salah satunya adalah *backpropagation*. Algoritma ini bekerja secara feedforward dan kemudian mengoreksi bobot secara berulang berdasarkan kesalahan output, sehingga model dapat belajar memetakan input menjadi output yang benar[16].

2.5 YOLO (You Only Look Once)

Deteksi objek merupakan salah satu cabang dari *computer vision* yang bertujuan untuk mengidentifikasi keberadaan suatu objek sekaligus menentukan Lokasi objek tersebut di dalam citra atau video. Informasi Lokasi objek umumnya direpresentasikan dalam bentuk *bounding box* yang disertai dengan label kelas.

Metode ini banyak digunakan pada aplikasi realtime karena mampu memberikan informasi objek secara langsung [17]

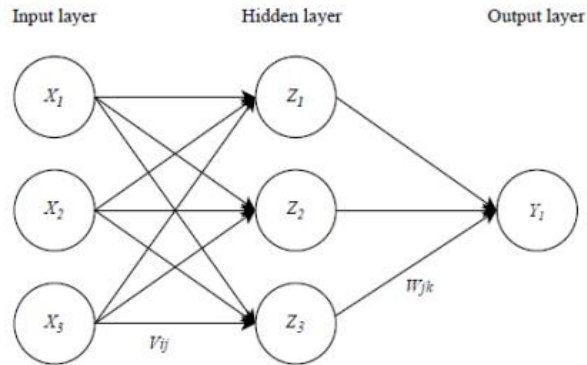
YOLOv8 merupakan versi terbaru dari keluarga YOLO yang dikembangkan untuk meningkatkan efisiensi dan akurasi deteksi objek. Dibandingkan metode deteksi dua tahap, YOLO menggunakan pendekatan *single-stage detector* sehingga proses inferensi menjadi lebih cepat dan sesuai untuk aplikasi yang membutuhkan deteksi secara *realtime* [18]. Beberapa penelitian di Indonesia menunjukkan bahwa YOLOv8 mampu menghasilkan nilai presisi dan *recall* yang tinggi serta stabil pada dataset dengan latar belakang bervariasi.[19] Oleh karena itu, pada penelitian ini YOLOv8 digunakan sebagai metode deteksi objek cabai untuk memisahkan objek cabai dari latar belakang sebelum dilakukan proses klasifikasi lebih lanjut

Dalam penelitian ini, YOLOv8 hanya digunakan pada tahap deteksi objek untuk menentukan Lokasi cabai dalam citra. Proses klasifikasi tingkat kematangan cabai tidak dilakukan oleh YOLOv8, melainkan menggunakan metode *Backpropagation Neural Network* berdasarkan fitur citra yang diekstraksi dari objek cabai hasil deteksi

2.5. Backpropagation Neural Network

Backpropagation Neural Network (BPNN) adalah salah satu jenis dari Jaringan Syaraf Tiruan yang sangat populer. Algoritma ini bekerja dengan menghitung selisih antara output prediksi dan target sebenarnya, kemudian memperbaiki bobot antar neuron melalui proses propagasi mundur. Dengan cara ini, BPNN mampu mempelajari dan memodelkan hubungan non-linear yang kompleks antara fitur input dan kelas output, sehingga sangat efektif untuk tugas-tugas klasifikasi citra[4]. Dalam konteks klasifikasi citra cabai, BPNN mampu mempelajari pola fitur warna dan tekstur yang diekstraksi dari gambar untuk membedakan tingkat kematangan buah.

Proses pelatihan pada Backpropagation Neural Network dilakukan secara iteratif hingga diperoleh bobot yang menghasilkan nilai kesalahan minimum. Pada setiap iterasi, bobot dan bias diperbarui berdasarkan hasil propagasi balik sehingga keluaran jaringan semakin mendekati nilai target. Proses ini terus dilakukan sampai nilai error telah memenuhi kriteria penghentian. Dengan mekanisme tersebut, BPNN mampu meningkatkan kemampuan generalisasi model terhadap data yang belum pernah dikenali sebelumnya[20].



Gambar 1. Arsitektur ANN pada Backpropagation

2.5.1 Struktur dan Alur Kerja BPNN

BPNN merupakan jenis Jaringan Syaraf Tiruan yang paling umum digunakan. Struktur dasarnya terdiri dari tiga bagian utama yaitu *Input Layer*, *Hidden Layer*, dan *Output Layer*. Ketika data dimasukkan ke dalam jaringan, proses pertama yang dilakukan adalah *forward propagation*, yaitu proses mengalirkan data dari lapisan input menuju lapisan output melalui *hidden layer*. Pada setiap neuron dilakukan perhitungan menggunakan bobot dan bias, kemudian hasilnya diproses oleh fungsi aktivasi untuk menghasilkan nilai keluaran. Nilai keluaran kemudian dibandingkan dengan nilai target sehingga menghasilkan nilai *error*. Besarnya *error* tersebut digunakan sebagai dasar untuk melakukan proses *backpropagation*, yaitu memperbaiki bobot jaringan agar kesalahan prediksi semakin kecil pada setiap pelatihan [21].

2.5.2 Mekanisme Backpropagation

Tahapan paling penting dalam pelatihan ini adalah proses *Backward Propagation*, dimana *error* yang sudah dihitung pada output akan dialirkan kembali ke lapisan sebelumnya untuk menyesuaikan bobot-bobot yang ada. Penyesuaian bobot ini dilakukan dengan algoritma *Gradient Descent* yang bertujuan untuk meminimalkan nilai *error*. Secara umum, proses pembelajaran BPNN terdiri dari tiga tahap yaitu *feedforward*, *backpropagation*, dan pembaruan bobot. Proses ini dilakukan berulang hingga nilai fungsi *loss* mencapai titik minimum. Salah satu keunggulan BPNN adalah kemampuannya dalam mengenali pola non linear yang kompleks dengan catatan parameter seperti jumlah *hidden layer*, *learning rate*, dan jumlah diatur dengan benar [22].

2.5.3 Fungsi Aktivasi

Fungsi aktivasi yang digunakan pada algoritma Backpropagation sebaiknya memiliki tiga karakter utama yaitu kontinu, mudah untuk diturunkan secara matematis, dan bukan merupakan fungsi monoton menurun. Salah satu fungsi yang sering digunakan dalam Jaringan Syaraf Tiruan adalah fungsi sigmoid biner. Fungsi ini didefinisikan sebagai berikut $f(x) = \frac{1}{1 + e^{-x}}$ dan turunan $f'(x) = f(x)(1-f(x))$. Nilai keluaran dari fungsi sigmoid biner berada dalam rentang 0 hingga 1 sehingga apabila data input maupun target output memiliki nilai diluar rentang tersebut, perlu dilakukan proses normalisasi terlebih dahulu. Hal ini bertujuan agar seluruh nilai dalam dataset berada dalam jangkauan fungsi aktivasi, sehingga proses pelatihan dapat berjalan secara optimal dan hasil yang diperoleh menjadi lebih akurat[23]

2.5.4 Penentuan Arsitektur dan Hyperparameter

Ketika merancang arsitektur jaringan, pemilihan jumlah neuron pada *Hidden Layer* merupakan hal penting. Salah satu pendekatan yang sering digunakan adalah rumus $i < m < 2i$, dimana i merupakan jumlah neuron input dan m adalah jumlah neuron pada hidden layer. Rumus ini merupakan pedoman awal, yang kemudian akan disesuaikan kembali melalui proses optimasi manual[24]. Selain itu, parameter seperti learning rate dan jumlah epoch juga diuji coba secara bertahap untuk memperoleh kombinasi yang paling ideal antara akurasi model dan waktu pelatihan yang efisien[25]

2.6. Penelitian Terkait

No	Tahun	Penulis	Jurnal	Metode	Kelemahan
1	2019	Herdiansah, A. et al.	Pemrosesan Citra Digital untuk Klasifikasi Tanaman Cabai Menggunakan Jaringan Syaraf Tiruan	BPNN	Hanya fitur RGB, tanpa grayscale, adaptasi parameter terbatas
2	2021	Khairullah & Putra	Identifikasi Kematangan Cabai Menggunakan Operasi Morfologi dan	BPNN	Akurasi hanya 70 %, dataset kecil (10 data latih)

			Backpropagation		
3	2018	Nugrahanto Aji Wicaksono	Klasifikasi Mutu Cabai Merah Berbasis Citra & JST	BPNN	Lebih fokus pada mutu umum, bukan kematangan spesifik
4	2024	Liantoni & Annisa	Fuzzy K-NN pada Klasifikasi Kematangan Cabai Berdasarkan Fitur HSV	Fuzzy K-NN	Tidak mempertimbangkan fitur tekstur
5	2020	Irma	Klasifikasi Tingkat Kematangan Cabai Merah Keriting Menggunakan SVM Multi-class berdasarkan Warna	SVM	Hanya fitur warna RGB, tanpa HSV atau tekstur
6	2023	Lutfi dkk	Klasifikasi Kematangan Cabai Menggunakan YCbCr	Transformasi YCbCr	Cb & Cr terbatas pada intensitas warna, tanpa evaluasi tekstur
7	2023	Mahdiyah	Klasifikasi Kualitas Citra Cabai dengan Gradient Boosting	Gradient Boosting	Akurasi rendah (76%), performa kurang konsisten
8	2022	Yasmin	K-Means Clustering untuk Klasifikasi Cabai Keriting	K-Means + GLCM	Tidak menggunakan pembelajaran terawasi/ Supervised Model
9	2014	Indra & Murinto	Aplikasi Deteksi Kematangan Cabai dengan Color Transform YCbCr	Warna YCbCr	Tidak menggunakan JST tanpa evaluasi model

10	2018	Betris	Implementasi CNN untuk Identifikasi Kematangan Cabai Berdasarkan Warna	CNN	Memerlukan dataset yang besar
----	------	--------	--	-----	-------------------------------

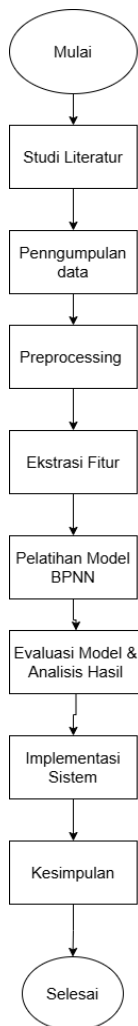
Tabel 1. Tabel Penelitian Terkait

Berbagai penelitian sebelumnya menunjukkan bahwa penggunaan metode BPNN dalam klasifikasi citra cabai masih memiliki beberapa keterbatasan. Misalnya, penelitian oleh Khairullah & Putta (2021) dalam Jurnal Sistem Informasi berhasil mengkombinasikan operasi morfologi dengan BPNN, namun akurasi hanya mencapai 70% karena dataset yang kecil serta fitur yang hanya mengandalkan warna RGB tanpa memperhatikan tekstur.

Penelitian ini mengatasi keterbatasan tersebut dengan menggabungkan fitur warna (HSV) dan fitur tekstur dari citra grayscale sebagai input ke dalam model BPNN, serta menerapkan proses preprocessing lengkap sebelum pelatihan. Diharapkan, dengan penggabungan fitur ini mampu menghasilkan performa klasifikasi yang lebih stabil dan akurat dibandingkan penelitian sebelumnya

Bab 3. Metodologi Penelitian

3.1 Perancangan Penelitian



Gambar 2. Flowchart Perancangan Penelitian

3.1.1 Studi Literatur

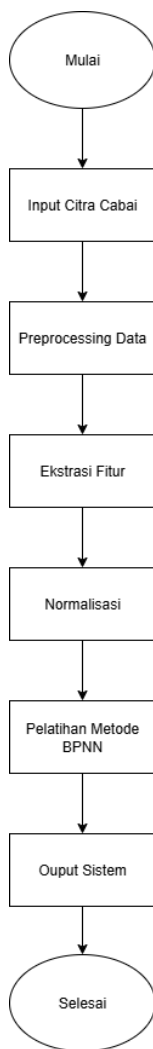
Studi literatur dilakukan untuk memahami konsep dasar dan metode yang telah diterapkan pada penelitian sebelumnya. Tahap ini dilakukan untuk mempelajari jurnal-jurnal nasional yang relevan dengan Jaringan Syaraf Tiruan, klasifikasi buah, dan objek berbasis visual. Studi ini penting untuk mengetahui kekurangan dan kelebihan metode-metode yang telah digunakan dalam penelitian sebelumnya. Selain itu, tahap ini juga membantu dalam menentukan Teknik processing yang efektif dan konfigurasi jaringan JST yang sesuai untuk data citra cabai. Kajian juga difokuskan pada struktur pelatihan dari algoritma Backpropagation neural network, agar implementasi dilakukan sesuai pendekatan yang terbukti efektif dalam penelitian sebelumnya

Hasil dari studi literatur ini menjadi acuan dalam menentukan struktur pemrosesan citra, penyesuaian parameter, dan evaluasi performa system. Dengan membandingkan metode dan hasil yang diambil dari penelitian sebelumnya, penelitian ini dapat mengidentifikasi potensi kelebihan serta kekurangan metode yang digunakan menyesuaikan dengan kebutuhan klasifikasi kematangan cabai secara digital

3.1.2 Pengumpulan Data

Tahap ini merupakan proses pengambilan citra cabai dalam tiga kondisi Tingkat kematangan yaitu mentah, matang, dan buruk. Citra diambil dengan kamera smartphone berkualitas tinggi dengan lingkungan pencahayaan yang baik dan latar belakang berwarna putih polos agar menghasilkan dataset yang konsisten. Selain itu, pengumpulan data juga dilakukan dalam beberapa sesi dengan sudut pengambilan yang berbeda untuk menghindari bias data. Data yang telah dikumpulkan kemudian disimpan dalam folder yang berbeda sesuai dengan kelas masing-masing. Cabai yang mentah dan sebagian cabai busuk diambil langsung dari kebun cabai yang ada di wilayah batam , sementara cabai yang matang dan sebagian cabai busuk diperoleh dari pasar tradisional terdekat.

3.1.3 Perancangan Sistem



Gambar 3. Flowchart Perancangan Sistem

3.1.3.1 Input citra cabai

Pada tahap awal sistem klasifikasi ini, dilakukan proses pemasukan dataset gambar citra cabai sebagai sumber data utama yang akan digunakan dalam pelatihan dan pengujian model klasifikasi. Dataset ini berisi kumpulan gambar cabai dengan berbagai kondisi tingkat kematangan, yaitu cabai mentah, matang, dan busuk. Gambar-gambar ini dapat diperoleh dari hasil dokumentasi lapangan secara langsung menggunakan kamera digital.

3.1.3.2 Preprocessing Data

Dalam tahap ini, dataset terlebih dahulu diolah agar siap digunakan untuk input model klasifikasi melalui beberapa proses yaitu:

1. Resize : Semua ukuran gambar diubah menjadi seragam agar jaringan syaraf dapat menerima input dengan struktur yang konsisten
2. Segmentasi: Latar belakang dipisahkan dengan metode thresholding
3. Noise Filtering : Filter median akan digunakan untuk mengurangi bintik-bintik pada gambar yang mengganggu citra gambar
4. Grayscale dan HSV : Citra diubah ke grayscale untuk fokus pada tekstur cabai, dan HSV untuk informasi warna yang lebih penting

Dengan tahap ini, citra yang masuk akan menjadi lebih bersih serta meminimalisir kesalah saat klasifikasi akibat faktor eksternal

3.1.3.3 Ekstraksi Fitur

Setelah data diproses, maka dilakukan proses ekstraksi fitur untuk memperoleh representasi numerik dari citra cabai yang akan digunakan sebagai input. Pada penelitian ini, digunakan 2 jenis fitur utama yaitu fitur warna dan fitur tekstur. Fitur warna terdiri atas *Hue Mean*, *Hue Standard Deviation*, *Saturation Mean*, *Saturation Standard Deviation*, *Value Mean*, *Value Standard Deviation*, *Red Green Ratio*, dan *Green Red Ratio*. Nilai rata-rata pada masing-masing komponen warna digunakan untuk merepresentasikan karakteristik warna dominan cabai. Perhitungan *Hue Mean*, *Saturation Mean*, dan *Value Mean* dilakukan menggunakan persamaan yang sama, yaitu rata-rata seluruh nilai piksel pada masing-masing kanal. Rumus perhitungan nilai rata-rata sebagai berikut

$$Mean(X) = \frac{1}{N} \sum_{i=1}^N X_i \quad (1)$$

Rumus tersebut digunakan secara umum untuk menghitung nilai rata-rata setiap kanal warna dengan mengganti variabel X_i sesuai kanal yang digunakan. Apabila variabel X_i menyatakan nilai Hue (H_i), maka diperoleh fitur Hue Mean. Jika X_i menyatakan nilai Saturation (S_i), maka diperoleh fitur Saturation Mean. Sementara itu, jika X_i menyatakan nilai Value (V_i), maka diperoleh fitur Value Mean. Ketiga fitur tersebut digunakan untuk merepresentasikan karakteristik warna dominan objek cabai sebagai masukan bagi model BPNN.

Selanjutnya, tingkat penyebaran nilai pada masing-masing kanal dihitung menggunakan Standard Deviation. Perhitungan ini digunakan untuk memperoleh

Hue Standard Deviation, Saturation Standard Deviation, Value Standard Deviation, serta Texture Standard Deviation. Nilai standar deviasi menunjukkan seberapa besar variasi nilai piksel terhadap nilai rata-ratanya. Semakin besar nilai standar deviasi, semakin tinggi variasi karakteristik warna atau tekstur pada objek cabai. Rumus perhitungan standar deviasi sebagai berikut

$$Std(X) = \sqrt{\frac{1}{N} \sum_{i=1}^N (X_i - meanX)^2} \quad (2)$$

Rumus tersebut digunakan untuk menghitung nilai penyebaran pada setiap kanal warna HSV dan Grayscale. Apabila variabel X_i menyatakan nilai Hue (H_i), maka diperoleh Hue Standard Deviation. Jika X_i menyatakan nilai Saturation (S_i), maka diperoleh Saturation Standard Deviation. Jika X_i menyatakan nilai Value (V_i), maka diperoleh Value Standard Deviation. Sedangkan apabila X_i menyatakan nilai intensitas grayscale ($Gray_i$), maka diperoleh Texture Standard Deviation. Nilai standar deviasi digunakan untuk mengukur tingkat penyebaran nilai piksel terhadap nilai rata-ratanya sehingga dapat menggambarkan variasi warna maupun tekstur pada permukaan cabai.

Kemudian, *Red Green Ratio* dan *Green Red Ratio* digunakan untuk menggambarkan dominasi warna merah dan hijau pada permukaan cabai. Kedua fitur ini dihitung berdasarkan perbandingan rata-rata intensitas piksel pada kanal merah dan kanal hijau dari citra hasil segmentasi. Nilai *Red Green Ratio* menunjukkan seberapa dominan warna merah dibandingkan warna hijau, sedangkan *Green Red Ratio* menunjukkan dominasi warna hijau terhadap warna merah. Kedua fitur tersebut membantu model BPNN membedakan tingkat kematangan cabai, karena cabai matang umumnya memiliki intensitas warna merah yang lebih tinggi, sedangkan cabai mentah memiliki intensitas warna hijau yang lebih dominan. Perhitungan kedua fitur tersebut dilakukan menggunakan rumus.

$$Red\ Green\ Ratio = \frac{MeanR}{MeanG} \quad (3)$$

$$Green\ Red\ Ratio = \frac{MeanG}{MeanR} \quad (4)$$

Selain fitur warna, penelitian ini juga menggunakan Texture Range sebagai fitur tekstur yang diperoleh dari citra grayscale hasil segmentasi. Texture Range digunakan untuk mengukur rentang intensitas piksel pada permukaan objek cabai berdasarkan selisih antara nilai intensitas piksel maksimum dan nilai intensitas piksel minimum. Nilai Texture Range yang besar menunjukkan adanya variasi tekstur yang tinggi pada permukaan cabai, sedangkan nilai yang kecil menunjukkan bahwa intensitas piksel cenderung lebih seragam. Fitur ini digunakan untuk membantu model BPNN membedakan karakteristik permukaan

cabai pada setiap kelas. Perhitungan Texture Range dilakukan menggunakan rumus berikut

$$Texture\ Range = Gray_{max} - Gray_{min} \quad (5)$$

dengan $Gray_{max}$ merupakan nilai intensitas piksel grayscale maksimum dan $Gray_{min}$ merupakan nilai intensitas piksel grayscale minimum pada area objek cabai hasil segmentasi.

3.1.3.4 Normalisasi Data

Setelah fitur-fitur diekstraksi, langkah penting selanjutnya adalah normalisasi data. Normalisasi bertujuan untuk menyetarakan skala seluruh nilai fitur agar memiliki rentang yang seragam, biasanya antara 0 dan 1. Hal ini penting karena fitur yang memiliki skala nilai besar bisa mendominasi proses pembelajaran jika tidak dinormalisasi. Dalam sistem klasifikasi berbasis BPNN, nilai input yang tidak distandarisasi dapat menyebabkan proses pelatihan tidak optimal, lambat, atau bahkan gagal menemukan pola yang tepat

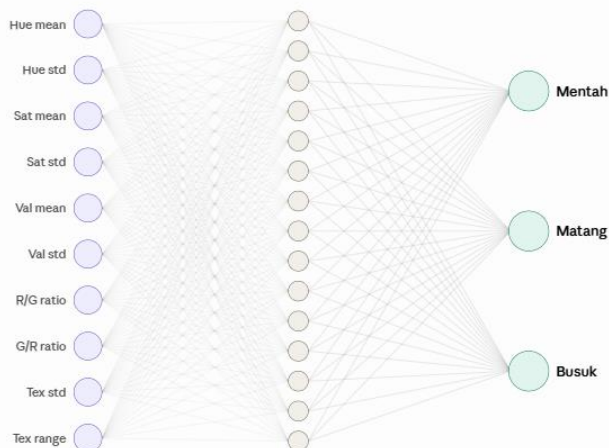
Metode normalisasi yang digunakan dalam penelitian ini adalah min-max normalisasi, dimana setiap nilai fitur x diubah dengan rumus:

$$x = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (6)$$

3.1.3.5 Pelatihan Model BPNN

Model klasifikasi dibangun menggunakan struktur jaringan syaraf tiruan Backpropagation dimana jaringan tersebut terdiri dari:

1. Lapisan input dengan 10 Neuron, yang mempresentasikan *Hue mean*, *Hue std*, *Saturation mean*, *Saturation std*, *Value mean*, *Value std*, *Red Green Ration*, *Green Red Ratio*, *Texture std*, *Texture Range*.
2. Hidden Layer terdiri atas 15 neuron. Jumlah neuron ini diperoleh berdasarkan hasil pengujian beberapa konfigurasi jaringan, dimana konfigurasi 15 neuron menghasilkan performa terbaik
3. Lapisan output dengan 3 neuron yaitu mentah, matang, dan busuk



Gambar 4. Arsitektur Backpropagation Neural Network

Model dilatih menggunakan algoritma propagasi balik yang memperbaiki bobot berdasarkan error yang dihitung dari output yang dihasilkan. Fungsi aktivasi yang dipakai adalah sigmoid dan proses optimasi dilakukan dengan gradient descent. Proses training dilakukan selama beberapa epoch yang dimana epoch adalah siklus penuh dari feedforward dan backpropagation. Rumus error total yang digunakan adalah berikut:

$$E = \frac{1}{2} \sum_i (t_i - o_i)^2 \quad (7)$$

Dimana t_i adalah target output dan o_i adalah output aktual. Dataset dibagi menjadi 80% untuk pelatihan dan 20% untuk pengujian. Cross-validation digunakan untuk menghindari overfitting dan hasil pelatihan akan dipantau melalui grafik loss function

3.1.3.6 Output Sistem

Tahap akhir dari sistem ini adalah keluaran dari hasil klasifikasi, yaitu kategori kematangan cabai. Sistem akan mengeluarkan prediksi berupa salah satu dari tiga kelas: cabai mentah, cabai matang, dan cabai busuk. Output ini dapat ditampilkan dalam bentuk tampilan antarmuka grafis (GUI)

3.1.4 Evaluasi Model dan Analisis Hasil

Tahap ini merupakan bagian penting karena menjadi tolak ukur untuk mengetahui seberapa baik sistem klasifikasi yang dirancang. Hasil prediksi dari model terhadap citra yang baru dibandingkan dengan label sebenarnya untuk menghitung akurasi dan kesalahan pada klasifikasi

Hasil klasifikasi yang diperoleh dari model dianalisis menggunakan confusion matrix. Confusion matrix berfungsi untuk menghitung matrix evaluasi utama. Perhitungan dilakukan berdasarkan jumlah data yang berhasil diklasifikasikan dengan benar dan yang salah oleh model. Berikut adalah rumus dari masing-masing matrix:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

$$Precision = \frac{TP}{TP + FP} \quad (9)$$

$$Recall = \frac{TP}{TP + FN} \quad (10)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

3.1.5 Implementasi Sistem

Setelah model BPNN mendapatkan performa yang memuaskan, selanjutnya adalah meimplementasikan sistem ke dalam bentuk antarmuka (GUI). Melalui tampilan GUI ini, penggunaan dapat mengunggah gambar cabai secara langsung dan mendapatkan hasil berupa label yang ditampilkan yaitu Mentah, Matang, dan Busuk. Dengan adanya tahap ini, penelitian tidak hanya berhenti pada analisis mode, namun dilanjutkan dengan penyediaan antarmuka yang aplikatif dan mudah digunakan di dunia nyata

3.1.6 Kesimpulan

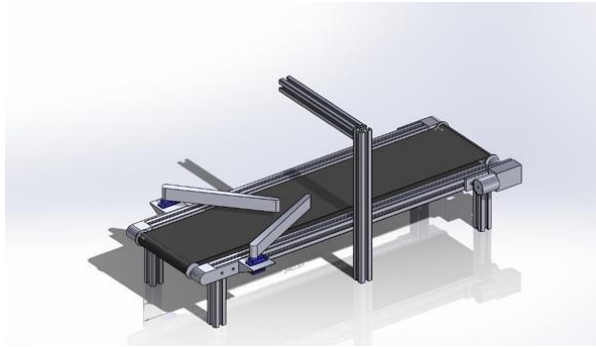
Jika sistem dapat mengklasifikasi citra cabai mentah, matang, dan busuk dengan akurasi yang tinggi, maka metode Backpropagation Neural Network dinyatakan efektif dalam menyelesaikan permasalahan klasifikasi ini. Penelitian juga dapat dikembangkan lebih lanjut ke arah sistem klasifikasi yang lebih kompleks seperti cacat fisik pada cabai. Dengan adanya sistem klasifikasi ini, diharapkan dapat membantu dalam sektor pertanian melakukan sortir cabai secara objektif.

3.2. Alat dan Bahan

Tabel 2. Estimasi Biaya

No.	Alat/bahan	Harga Satuan (Rp.)	Jumlah	Total (Rp.)	Keterangan ¹
1	Laptop	-	1	-	Sudah dimiliki
2	Kamera HP	-	1	-	Sudah dimiliki
3	Visual Studio	-	1	-	Open Source
4	Cabai Merah	Rp. 60.000	1kg	Rp. 60.000	Untuk dataset
5	Conveyor	-	1	-	Borang
Total				Rp. 60.000	

3.3 Perancangan Alat



Gambar 5. Desain Alat

Conveyor yang digunakan pada penelitian ini sebagai media pemindahan objek cabai selama proses klasifikasi. Conveyor dirancang agar cabai bergerak secara *continue* melewati area deteksi kamera sehingga proses indentifikasi dapat dilakukan secara otomatis dan konsisten.

Sistem conveyor terdiri atas satu buah motor DC yang berfungsi sebagai penggerak utama belt conveyor sehingga cabai dapat berpindah dari titik masukan menuju area klasifikasi. Pada bagian akhir conveyor dipasang dua buah motor servo yang berfungsi sebagai aktuator penyortiran hasil klasifikasi. Sistem ini juga dilengkapi dengan satu buah sensor proximity yang dipasang sebelum area klasifikasi. Sensor ini berfungsi mendeteksi keberadaan cabai yang melintas pada conveyor serta memberikan sinyal kepada sistem agar proses pengambilan citra dan klasifikasi dilakukan pada waktu yang tepat

3.4. Pengujian Sistem

Pengujian sistem dilakukan untuk mengetahui seberapa baik sistem klasifikasi tingkat kematangan cabai ini dapat bekerja secara optimal. Pengujian dilakukan pada beberapa aspek yaitu pengujian akurasi model dan pengujian kestabilan model selama proses pelatihan dan pengujian. Pengujian ini bertujuan untuk memastikan sistem bekerja secara baik dan efisien saat digunakan dalam kondisi nyata

3.4.1 Pengujian Hasil Klasifikasi Model

Tahap ini bertujuan untuk mengetahui sejauh mana model BPNN ini dapat memprediksi tingkat kematangan cabai dengan baik. Hasil klasifikasi dibandingkan dengan label yang aktual. Untuk menampilkan hasil prediksi dan kesalahan klasifikasi digunakan confusion matrix. Kemudian akan dihitung akurasi untuk setiap kelas untuk mengetahui performa model secara detail terhadap tingkat kematangan.

3.4.2 Pengujian Implementasi Sistem

Pengujian implemementasi sistem dilakukan untuk memastikan bahwa aplikasi yang dikembangkan mampu mengintegrasikan seluruh proses, mulai dari deteksi objek cabai hingga klasifikasi kondisi cabai menggunakan metode Backpropagation neural Network. Pengujian ini bertujuan untuk mengetahui apakah sistem dapat berjalan sesuai dengan alur yang telah dirancang serta menghasilkan keluaran yang sesuai terhadap objek cabai yang diuji.

Pengujian dilakukan pada dua mode operasi, yaitu Upload Mode dan Realtime Mode. Pada upload mode, sistem diuji menggunakan citra cabai yang dipilih oleh pengguna melalui antarmuka aplikasi, sedangkan pada realtime mode, sistem melakukan deteksi dan klasifikasi secara langsung menggunakan kamera. Parameter yang diamati meliputi, keberhasilan deteksi objek, ketepatan hasil klasifikasi, serta kestabilan sistem selama proses pengujian.

3.4.3 Pengujian Variasi Jumlah Epoch

Setelah melakukan pengujian klasifikasi pada model, maka dilakukan pengujian terhadap pengaruh variasi jumlah epoch terhadap performa model. Epoch adalah siklus penuh pada pelatihan terhadap seluruh dataset. Dalam pengujian ini, model akan dilatih dengan tiga variasi epoch yaitu 50, 100 dan 200. Tujuannya agar dapat diketahui seberapa besar peningkatan akurasi, precision, recall dan F1-score seiring bertambahnya jumlah epoch. Jika nilai-nilai tersebut meningkat secara konsisten, maka model dikatakan memperoleh manfaat dari pelatihan yang lebih lama. Namun jika performa menurun maka dapat dikatakan bahwa terjadi overfitting. Selain itu, nilai *Loss Value* juga dicatat pada setiap epoch. Penurunan loss secara konsisten menunjukkan model berhasil meminimalkan kesalahan prediksi seiring bertambahnya epoch

3.4.4 Pengujian Variasi Jumlah Hidden Layer

Terakhir, dilakukan pengujian terhadap pengaruh jumlah layer terhadap performa model. Hidden layer mempunyai peran yang penting dalam memproses dan menyimpan pola dari data yang kompleks. Pada tahap ini, model akan diuji dengan variasi konfigurasi 1,2 dan 3 hidden layer untuk melihat mana yang lebih optimal. Evaluasi dilakukan menggunakan matriks akurasi, precision, recall, dan F1-Score. Penambahan hidden layer diharapkan dapat meningkatkan kemampuan model dalam menangkap pola yang kompleks, namun juga dapat menyebabkan overfitting. Oleh karena itu, pengujian ini membantu menentukan konfigurasi paling efektif untuk digunakan

3.4.5 Pengujian Metode Cross Validation

Pengujian terakhir difokuskan pada metode validasi yang akan digunakan. Dalam tahap ini akan digunakan metode K-Fold Cross Validation sebagai teknik

validasi untuk menguji kemampuan generalisasi dari model. Pada teknik ini dataset akan dibagi menjadi beberapa bagian dan kemudian akan dilatih dan diuji sebanyak jumlah fold. Setiap iterasi 1 fold digunakan sebagai data uji dan sisanya sebagai data latih, hasil dari setiap fold akan dicatat lalu dirata-ratakan untuk mendapatkan hasil akhir

Bab 4. Hasil dan Pembahasan

4.1 Pengumpulan Data

Pengumpulan data pada penelitian ini dilakukan dengan mengambil citra cabai yang diklasifikasikan ke dalam tiga kelas, yaitu mentah, matang, dan busuk. Proses pengambilan data dilaksanakan pada bulan Desember 2025 di beberapa lokasi berbeda untuk memperoleh variasi kondisi objek yang representatif.

Cabai mentah dan sebagian cabai busuk diperoleh langsung dari kebun cabai yang berlokasi di Perumahan Cemara Asri, Kecamatan Sagulung. Sementara itu, cabai matang dan sebagian cabai busuk diperoleh dari dua pasar yang berlokasi di daerah Batu Aji dan Marina, Kota Batam. Pengambilan dari dua sumber berbeda dilakukan untuk meningkatkan variasi data dan mengurangi bias terhadap satu lokasi tertentu.

Sampel yang digunakan terdiri atas 89 buah cabai mentah (hijau), 80 buah cabai matang (merah), dan 80 buah cabai busuk, sehingga total cabai yang digunakan sebanyak 249 buah cabai. Setiap cabai difoto sebanyak 2 hingga 3 kali dengan variasi posisi objek. Variasi pengambilan citra tersebut bertujuan untuk memperoleh keragaman data, sehingga model dapat mempelajari karakteristik objek dari berbagai sudut pengambilan gambar.

Proses akuisisi citra dilakukan menggunakan kamera *smartphone* dengan resolusi 48 MP. Pengambilan gambar dilakukan dengan latar belakang putih menggunakan kertas HVS untuk memudahkan proses segmentasi. Pencahayaan dibuat merata dengan memanfaatkan cahaya alami pada siang hari dan dibantu dengan *flash* kamera aktif untuk meminimalkan bayangan. Jarak pengambilan gambar dijaga relatif konsisten, yaitu sekitar 25-30 cm dari objek. Sudut pengambilan gambar dilakukan secara tegak lurus dari atas ke bawah untuk menjaga konsistensi perspektif. Setiap cabai difoto secara terpisah guna menghindari tumpang tindih objek dalam satu citra.

Dari proses akuisisi tersebut, diperoleh sebanyak 495 citra, yang terdiri atas 165 citra cabai mentah, 165 citra cabai matang, dan 165 citra cabai. Seluruh data kemudian dikumpulkan dan dipisahkan berdasarkan kelas untuk dilanjutkan ke tahap preprocessing



Gambar 6. Pengumpulan Cabai

4.2 Preprocessing

Tahap ini bertujuan untuk mempersiapkan data citra cabai agar siap diproses pada saat pelatihan dengan menyamakan ukuran gambar, mengurangi noise, dan memisahkan objek dari latar belakang

4.2.1 Resize Image

Tahap pertama ialah mengubah ukuran semua citra menjadi ukuran yang seragam yaitu 224 x 224 piksel. Proses resize dilakukan dengan mempertahankan aspek rasio asli gambar untuk menghindari distorsi objek. Proses ini penting untuk memastikan jaringan syaraf tiruan menerima input dengan dimensi yang konsisten pada setiap data latih

Pemilihan ukuran 224 x 224 piksel didasarkan pada pertimbangan keseimbangan antara kualitas informasi citra dan efisiensi komputasi. Ukuran citra yang lebih besar dapat menyimpan detail visual yang lebih lengkap, namun membutuhkan waktu pemrosesan yang lebih lama dan beban komputasi yang lebih tinggi. Sebaliknya, ukuran citra yang terlalu kecil dapat menyebabkan hilangnya informasi penting yang berpengaruh terhadap akurasi klasifikasi. Oleh karena itu, ukuran 224 x 224 piksel dipilih karena mampu mempertahankan informasi visual yang cukup membedakan karakteristik cabai, sekaligus menjaga efisiensi waktu pelatihan dan pengujian model.



Gambar 8. Cabai Sebelum di Resize



Gambar 7. Cabai Setelah di Resize

4.2.2 Segmentasi

Segmentasi dilakukan untuk memisahkan objek cabai dari latar belakang menggunakan metode thresholding pada ruang warna HSV. Metode HSV dipilih karena lebih robust terhadap perubahan pencahayaan dibandingkan RGB.

Proses segmentasi menggunakan beberapa range warna HSV untuk menangkap variasi warna cabai dari hijau hingga kemerahan dan coklat gelap. Setelah mendapatkan mask awal, dilakukan operasi morfologi closing dan opening untuk menghilangkan noise kecil. Kernel yang digunakan berukuran 5 x 5 dengan iterasi closing sebanyak 3 kali dan opening sebanyak 2 kali.

Selain menggunakan metode segmentasi berbasis HSV, pada penelitian ini juga digunakan model YOLO untuk membantu mendeteksi lokasi objek cabai dalam citra. YOLO berfungsi untuk menghasilkan *bounding box* yang mengidentifikasi posisi cabai secara otomatis. Hasil deteksi ini kemudian digunakan sebagai acuan dalam menentukan area objek sebelum proses segmentasi dilakukan. Dengan demikian, kombinasi antara objek menggunakan YOLO dan segmentasi berbasis warna HSV dapat meningkatkan akurasi pemisahan objek cabai dari latar belakang, karena proses segmentasi difokuskan pada area yang telah terdeteksi sebagai objek cabai.



Gambar 9. Cabai Sebelum di Segmentasi



Gambar 10. Cabai Setelah di Segementasi

Pada beberapa kasus, kegagalan segmentasi terjadi akibat kondisi citra yang kurang optimal, seperti latar belakang yang tidak cukup kontras dengan objek serta warna cabai yang cenderung gelap. Kondisi ini menyebabkan nilai warna pada ruang HSV sulit dibedakan antara objek dan latar belakang, sehingga proses thresholding tidak menghasilkan mask yang sesuai.

Untuk mengatasi masalah tersebut, dilakukan perbaikan pada proses pengambilan citra dengan memastikan penggunaan latar belakang putih yang lebih bersih serta pencahayaan yang lebih merata. Selain itu, posisi pencahayaan diatur agar tidak menimbulkan bayangan yang kuat pada objek cabai. Dengan

peningkatan kualitas citra, perbedaan warna antara objek dan latar belakang menjadi lebih jelas sehingga proses segmentasi dapat berjalan lebih optimal.



Gambar 11. Contoh Segmenetasi Gagal

4.2.3 Filtering (Noise Reduction)

Setelah segmentasi, dilakukan filtering menggunakan Median Filter dengan kernel 5 x 5 untuk mengurangi noise pada citra. Median filter dipilih karena lebih efektif menghilangkan *salt and pepper noise* sambil mempertahankan tepi objek dengan baik. Filter ini bekerja mengganti nilai piksel dengan nilai median dari piksel-piksel tetangga dalam kernel 5 x 5. Hasilnya citra akan menjadi lebih hasil dan bersih dari noise yang didapat saat mengganggu proses ekstraksi fitur

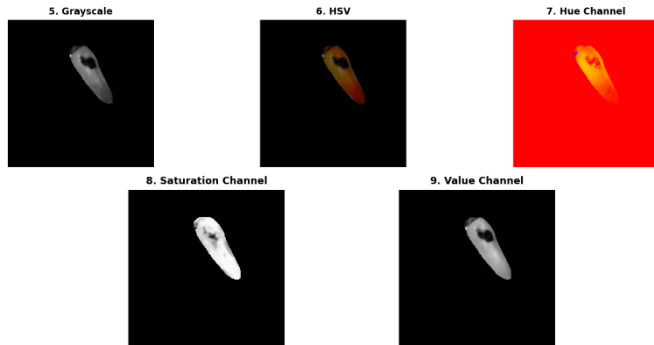


Gambar 13. Cabai Sebelum Filtering Gambar 12. Cabai Setelah Filtering

4.2.4 Konversi Ruang Warna

Pada tahap ini, citra hasil segmentasi dikonversi ke dua ruang warna untuk kebutuhan ekstraksi fitur. Citra dikonversi ke HSV untuk mengambil warna cabai, serta ke grayscale untuk mengambil informasi permukaan.

Konversi dilakukan menggunakan fungsi bawaan OpenCV, dimana citra RGB diubah menjadi HSV untuk memisahkan komponen warna (Hue), kejenuhan (Saturation), dan kecerahan (Value). Sementara itu, citra grayscale digunakan untuk menyederhanakan informasi citra menjadi satu kanal intensitas yang akan digunakan pada ekstraksi fitur tekstur.



Gambar 14. Hasil konversi Grayscale dan HSV

4.3 Ekstraksi Fitur

Ekstraksi fitur merupakan proses mengubah data citra menjadi representasi numerik yang dapat diolah oleh algoritma machine learning. Pada penelitian ini, diekstraksi 10 fitur yang terdiri dari fitur warna dan fitur tekstur untuk mempresentasikan karakteristik cabai

Tabel 3. Hasil Beberapa Features Setelah di Ekstraksi

Hue mean	Saturation mean	Value mean	Red Green Ratio	Texture range
7.597843450	165.8909744	59.30591054	1.529047131	6.705965508
10.11081441	172.9319092	50.50333778	1.359327624	14.79060581
10.85995623	168.6214442	60.32399368	1.726660703	15.77151910
10.85995623	180.7066259	50.50333778	2.429105327	15.20496022

4.3.1 Fitur Warna (HSV)

Tabel 4. Rata-rata Fitur Warna HSV

Kelas	Hue Mean	Saturation Mean	Value Mean
Mentah	0,1374	0,4049	0,6035
Matang	0,3020	0,7491	0,5819
Busuk	0,0795	0,7272	0,3041

Berdasarkan hasil ekstraksi fitur nilai hue_mean pada cabai matang memiliki nilai yang lebih tinggi (sekitar 0.3020) dibandingkan cabai mentah (sekitar 0.1374) dan cabai busuk (sekitar 0.0795). Hal ini menunjukkan bahwa distribusi warna

pada cabai matang dalam ruang HSV cenderung berupa pada rentang nilai yang lebih tinggi dibandingkan kelas lainnya.

Selain itu, nilai *saturation_mean* pada cabai matang juga lebih tinggi sekitar (0.7491) dibandingkan cabai mentah (sekitar 0.4049), yang menunjukkan bahwa cabai matang memiliki warna yang lebih pekat dan jenuh. Sementara itu, cabai busuk memiliki nilai *saturation* yang lebih rendah dibandingkan cabai matang, yang mengindikasikan warna yang mulai memudar.

Pada fitur *value_mean*, cabai mentah memiliki nilai tertinggi (sekitar 0.6035), diikuti cabai matang (sekitar 0.5819), dan cabai busuk yang memiliki nilai paling rendah (sekitar 0.3041). Hal ini menunjukkan bahwa cabai busuk memiliki tingkat kecerahan yang lebih rendah dibandingkan cabai segar.

4.3.2 Fitur Warna (RGB Ratio)

Tabel 5. Rata-rata Fitur RGB Ratio

Kelas	Red Green Ratio	Green Red Ratio
Mentah	0,0058	0,9204
Matang	0,1993	0,2494
Busuk	0,1242	0,4602

Berdasarkan hasil perhitungan fitur RGB ratio, terlihat bahwa nilai *red_green_ratio* cabai matang memiliki nilai paling tinggi yaitu sekitar 0,1993, dibandingkan cabai mentah sebesar 0.0058 dan cabai busuk sebesar 0.1242. Hal ini menunjukkan bahwa cabai matang memiliki dominasi warna merah yang lebih kuat dibandingkan kelas lainnya.

Sebaliknya, nilai *green_red_ratio* pada cabai mentah memiliki nilai paling tinggi yaitu sekitar 0.9204, dibandingkan cabai matang sebesar 0.2494 dan cabai busuk sebesar 0.4602. Hal ini menunjukkan bahwa cabai mentah di dominasi oleh warna hijau. Sementara itu, cabai busuk memiliki nilai yang berada di antara keduanya, yang menunjukkan percampuran warna akibat perubahan kondisi fisik cabai

4.3.3 Fitur Tekstur

Tabel 6. Rata-rata Fitur Tekstur

Kelas	Texture Std	Texture Range
Mentah	0,2004	0,5668
Matang	0,1957	0,4909
Busuk	0,2561	0,4084

Berdasarkan hasil ekstraksi fitur tekstur, nilai *texture_std* pada cabai busuk memiliki nilai paling tinggi yaitu sekitar 0.2561, dibandingkan cabai mentah sebesar 0.1975 dan cabai matang sebesar 0.1957. Hal ini menunjukkan bahwa

permukaan cabai busuk cenderung lebih tidak merata dan memiliki variasi intensitas yang lebih tinggi.

4.3.4 Normalisasi Data

Setelah ekstraksi fitur, dilakukan normalisasi menggunakan metode Min-Max Scaling untuk menyeragamkan rentang nilai semua fitur ke dalam interval [0,1]. Rumus Min-Max Normalization yang digunakan adalah:

$$x = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Normalisasi memastikan bahwa tidak ada fitur yang mendominasi proses pelatihan karena memiliki skala nilai yang lebih besar. Berikut Adalah gambar tabel excel setelah di normalisasi

Tabel 7. Data Fitur Setelah di Normalisasi

Hue mean	Saturation mean	Value mean	Red Green Ratio	Texture range
0,02695872	0,552288896	0,383416089	0,122876887	0,241935484
0,041949404	0,595002579	0,309643125	0,119010874	0,150537634
0,046418276	0,568853233	0,28008048	0,10429378	0,204301075
0,046418276	0,568853233	0,28008048	0,10429378	0,204301075

4.4 Pelatihan Model BPNN

Pelatihan model klasifikasi dilakukan menggunakan algoritma Backpropagation Neural Network dengan arsitektur feedforward tiga layer yaitu input layer, hidden layer, dan output layer. Proses pelatihan melibatkan *Forward Propagation* untuk menghasilkan prediksi dan *Backpropagation* untuk memperbarui bobot berdasarkan error yang dihasilkan. Selama proses pelatihan, dilakukan pengamatan terhadap nilai akurasi dan loss pada data pelatihan maupun data validasi. Pengamatan ini bertujuan untuk mengetahui apakah model mengalami overfitting. Model dikatakan mengalami overfitting apabila akurasi data pelatihan terus meningkat, sedangkan akurasi data validasi menurun atau tidak mengalami peningkatan. Pada penelitian ini, nilai akurasi pelatihan dan validasi menunjukkan pola yang relatif seimbang sehingga model memiliki kemampuan generalisasi yang baik.

4.4.1 Arsitektur Jaringan

Arsitektur jaringan syaraf tiruan yang digunakan memiliki struktur sebagai berikut:

- Input Layer: Terdiri dari 10 neuron yang memiliki 10 fitur hasil ekstraksi

- Hidden Layer: Terdiri dari sejumlah neuron yang ditentukan melalui eksperimen hyperparameter tuning. Jumlah neuron yang diuji adalah 10, 12, dan 15 neuron. Fungsi aktivasi yang digunakan adalah sigmoid, yang berfungsi untuk mengubah nilai input menjadi rentang 0 hingga 1 serta memungkinkan jaringan mempelajari hubungan non-linear antar fitur.
- Output Layer: Terdiri dari 3 neuron yang merepresentasikan 3 kelas klasifikasi yaitu Mentah, Matang, dan Busuk. Fungsi aktivasi yang digunakan adalah sigmoid, yang menghasilkan nilai probabilitas untuk setiap kelas dengan total probabilitas sama dengan 1

Dengan arsitektur ini, model mampu mempelajari hubungan non-linear antara fitur citra cabai dan tingkat kematangannya, sehingga dapat menghasilkan keputusan klasifikasi yang akurat

Output dari jaringan berupa tiga nilai probabilitas yang menunjukkan tingkat keyakinan model terhadap masing-masing kelas. Nilai probabilitas ini berada pada rentang 0 hingga 1 dan jumlahnya selalu sama dengan 1. Sebagai contoh, model dapat menghasilkan output 0.10, 0.85, 0.05, nilai tersebut masing masing mempresentasikan probabilitas kelas Mentah, Matang, dan Busuk. Pada contoh tersebut, nilai tertinggi adalah 0.85 sehingga citra diklasifikasikan sebagai cabai matang. Penentuan kelas akhir dilakukan menggunakan metode *argmax*, yaitu memilih indeks dengan nilai probabilitas terbesar. Pendekatan ini memungkinkan model tidak hanya memberikan hasil klasifikasi, tetapi juga tingkat keyakinan terhadap hasil tersebut.

4.4.2 Inisialisasi Bobot

Inisialisasi bobot menggunakan metode Xavier (Glorot) Initialization yang terbukti efektif untuk jaringan dengan fungsi aktivasi sigmoid. Metode ini menginisialisasi bobot secara random dengan distribusi uniform dalam rentang tertentu untuk mencegah vanishing gradient atau exploding gradient.

Rumus Xavier initialization untuk bobot antara layer l dan j adalah:

$$w_{lj} \sim \mathcal{U}\left(-\frac{\text{limit}}{\sqrt{n_{\text{input}} + n_{\text{output}}}}, \frac{\text{limit}}{\sqrt{n_{\text{input}} + n_{\text{output}}}}\right) \quad (2)$$

Dimana n_{input} adalah jumlah neuron di layer input dan n_{output} adalah jumlah neuron di lay. Bias diinisialisasi dengan nilai nol untuk semua layer

4.4.3 Proses Forward Propagation

Proses ini melibatkan perhitungan dari layer input hingga output layer:

Langkah 1 : Hitung input ke hidden layer

$$H_{\text{input}} = X \cdot W_{ih} + b_h \quad (3)$$

Langkah 2 : Aktivasi hidden layer dengan sigmoid

$$h_{output} = \sigma(h_{input}) \quad (4)$$

Langkah 3: Hitung input ke output layer

$$o_{input} = h + output \cdot W_{ho} + b_o \quad (5)$$

Langkah 4: Aktivasi output layer dengan sigmoid

$$output = \sigma(o_{input}) \quad (6)$$

Dimana X adalah input (fitur), W_{ih} adalah bobot input-hidden, W_{ho} adalah bobot hidden output, b_h dan b_o adalah bias, dan σ adalah fungsi sigmoid

4.4.4 Proses Backpropagation

Proses ini menghitung gradient error terhadap setiap bobot dan melakukan update menggunakan gradient descent dengan momentum

Langkah 1 : Hitung error output layer

$$\delta_{output} = (y_{true} - y_{pred}) \cdot \sigma'(output) \quad (7)$$

Langkah 2 : Propagate error ke hidden layer

$$\delta_{hidden} = \delta_{output} \cdot W_{ho}^T \cdot \sigma'(h_{output}) \quad (8)$$

Langkah 3 : Update bobot dengan momentum

$$\begin{aligned} v &= \mu \cdot v_{old} + \alpha \cdot gradient \\ W_{new} &= W_{old} + v \end{aligned} \quad (9)$$

Dimana δ_{output} adalah *Error* atau kesalahan di output layer, y_{true} adalah target sebenarnya, y_{pred} adalah output yang dihasilkan model atau prediksi model, σ' adalah turunan sigmoid, μ (momentum) = 0.9 dan α (learning rate) = 0.05. Momentum membantu mempercepat konvergensi dan mengurangi oscillation dengan mempertahankan 90% dari velocity update sebelumnya

4.4.5 Loss Function

Loss function yang digunakan adalah Mean Squared Error (MSE) yang mengukur rata-rata kuadrat selisih antara output prediksi dengan target :

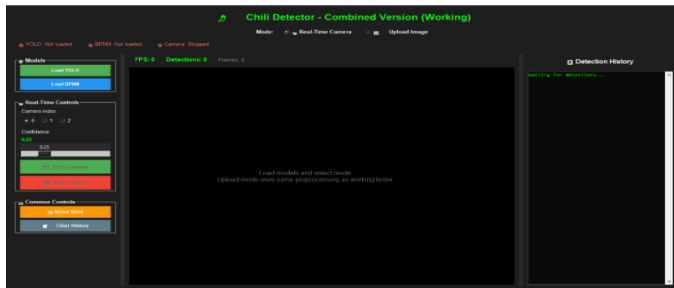
$$Loss = (0.5 \times N) \Sigma (y_{true} - y_{pred})^2 \quad (10)$$

Dimana N adalah jumlah sampel. MSE dipilih karena memberikan penalti yang lebih besar untuk error yang besar dan memiliki turunan yang baik untuk proses gradient descent

4.5 Implementasi Sistem GUI

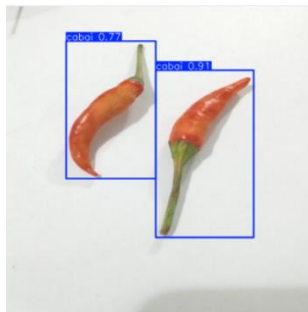
Setelah model YOLO dan BPNN berhasil dilatih dengan akurasi yang sangat baik, tahap selanjutnya adalah mengimplementasikan kedua model tersebut ke dalam Graphical User Interface (GUI) yang interaktif dan user-friendly. Implementasi GUI ini bertujuan untuk memudahkan pengguna dalam melakukan deteksi dan klasifikasi kualitas cabai tanpa perlu memahami algoritma yang digunakan.

Sistem GUI yang dikembangkan menggunakan framework Tkinter yang merupakan library standar Python untuk pembuatan aplikasi desktop. Tkinter dipilih karena kemudahan penggunaannya, kompatibilitas yang baik, serta tidak perlu library eksternal tambahan. Aplikasi dirancang dengan resolusi 1600 x 900 piksel untuk memberikan ruang yang cukup bagi semua komponen interface.



Gambar 15. Tampilan GUI

Integrasi model YOLO dan BPNN ke dalam antarmuka GUI dirancang untuk membentuk pemrosesan yang terstruktur dan efisien. Sistem menerima input berupa frame video pada mode realtime dan citra gambar pada mode upload, yang selanjutnya akan diproses oleh model YOLO untuk mendeteksi keberadaan objek cabai. Hasil deteksi YOLO kemudian di-crop untuk memperoleh *Region of Interest* (ROI) yang hanya berisi objek cabai. ROI tersebut selanjutnya melalui tahapan preprocessing lalu dilakukan ekstraksi fitur yang kemudian dinormalisasikan menggunakan scaler yang sama dengan yang digunakan pada tahap training. Fitur yang telah dinormalisasi diinputkan selama model BPNN untuk menentukan tingkat kematangan cabai yaitu mentah, matang, dan busuk.

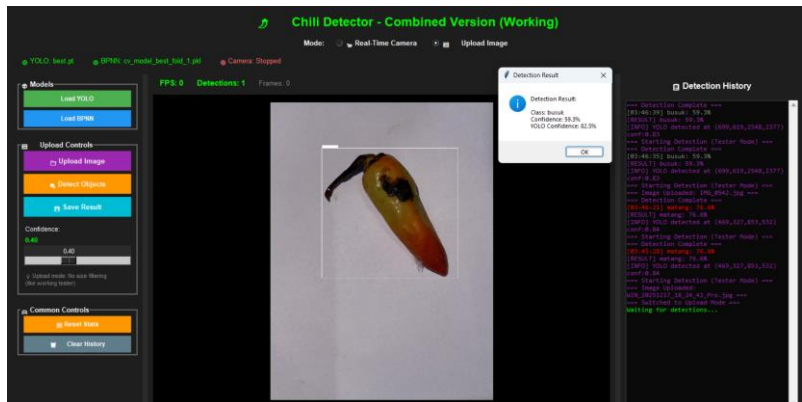


Gambar 16. Hasil Deteksi YOLO

4.5.1 Hasil Pengujian Mode Upload

Pengujian upload mode dilakukan untuk memverifikasi bahwa sistem dapat melakukan analisis gambar yang telah tersimpan dengan akurasi yang sama seperti pada model tester yang telah tervalidasi sebelumnya.

Proses penggunaan Upload Mode dimulai dengan user mengklik tombol 'Upload Image' yang akan membuka file dialog untuk memilih gambar cabai. Setelah gambar dipilih, kemudian sistem akan menampilkan gambar tersebut pada display untuk verifikasi visual



Gambar 17 GUI Upload Image

Pengujian mode upload dilakukan dengan mengupload gambar cabai busuk untuk dianalisis oleh sistem. Hasil menunjukkan bahwa YOLO berhasil mendeteksi objek cabai dengan baik dan menghasilkan bounding box yang presisi. Model BPNN juga berhasil mengklasifikasikan gambar sebagai cabai busuk dengan *confidence* yang cukup baik. Sistem menampilkan popup message box yang berisi informasi lengkap tentang hasil deteksi.

4.5.2 Pengujian Realtime Mode

Pengujian Realtime Mode dilakukan untuk memverifikasi kemampuan sistem dalam melakukan deteksi dan klasifikasi secara realtime pada video dari webcam ataupun webcam eksternal.

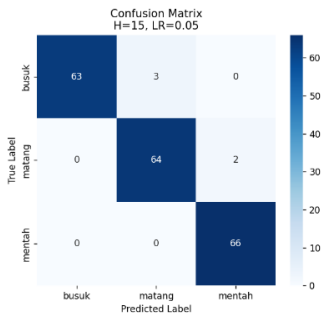
Ketika user mengklik tombol 'Start Camera', sistem akan melakukan inisialisasi kamera menggunakan OpenCV VideoCapture. Sistem akan membuka kamera dengan index yang dipilih kemudian kamera akan terbuka dan melakukan deteksi

4.6.1 Hasil Klasifikasi Model

Pengujian hasil klasifikasi model dilakukan untuk mengevaluasi kemampuan BPNN dalam mengklasifikasikan kondisi cabai menjadi tiga kelas, yaitu mentah, matang, dan busuk. Evaluasi dilakukan menggunakan data uji yang tidak digunakan selama proses pelatihan sehingga hasil yang diperoleh dapat menggambarkan kemampuan model dalam melakukan generalisasi terhadap data baru. Pada penelitian ini, hasil klasifikasi dianalisis menggunakan confusion matrix untuk mengetahui jumlah prediksi benar maupun kesalahan prediksi setiap kelas

Tabel 8. Hasil Akurasi pada Setiap Kelas Cabai

Kelas Cabai	Jumlah Prediksi Benar	Akurasi (%)
Mentah	63	95
Matang	64	96
Busuk	66	100



Gambar 19. Confusion Matrix

Hasil evaluasi perkelas menunjukkan performa yang sangat konsisten dan memuaskan. Kelas busuk mencapai akurasi sempurna 100% dengan 66 prediksi benar dari 66 data, menunjukkan bahwa karakteristik cabai busuk sangat mudah dibedakan oleh model. Kelas matang memiliki akurasi 96.97% dengan 64 prediksi benar, dan kelas Mentah mencapai 95.45% dengan 63 prediksi benar. Rata rata akurasi keseluruhan dari ketiga kelas adalah 97.47%, menunjukkan bahwa model sangat reliabel dan konsisten dalam mengklasifikasi semua tingkat kematangan cabai.

Tabel 9. Analisis Kesalahan Klasifikasi Model

Gambar Cabai	Label Aktual	Hasil Prediksi	Analisis
	Busuk	Matang	Warna merah masih mendominasi permukaan cabai sedangkan bagian yang mengalami pembusukan hanya terdapat pada sebagian area. Akibatnya, fitur warna yang diekstraksi lebih mendekati karakteristik cabai matang sehingga model mengklasifikasikannya sebagai matang
	busuk	Matang	Bercak pembusukan masih berukuran kecil dan warna masih menyerupai cabai matang
	Busuk	Matang	Proses pembusukan belum merata dan kondisi ciri visual yang dominan masih berupa warna merah
	Matang	Mentah	Warna cabai belum merata dan belum sepenuhnya warna merah

	Matang	Mentah	Sebagian permukaan cabai masih memiliki warna hijau dan kuning yang masih memiliki kemiripan dengan karakteristik kelas mentah.
---	--------	--------	---

Berdasarkan hasil analisis pada tabel, seluruh kesalahan klasifikasi terjadi pada sampel yang memiliki karakteristik visual yang berada di antara dua kelas. Kesalahan pada kelas busuk disebabkan oleh warna merah yang masih dominan sehingga fitur warna lebih menyerupai cabai matang, sedangkan kesalahan pada kelas matang terjadi karena perubahan warna belum berlangsung secara merata sehingga sebagian permukaan cabai masih menyerupai cabai mentah. Hasil tersebut menunjukkan bahwa performa model sangat dipengaruhi oleh kemiripan karakteristik warna dan tekstur antar kelas, meskipun secara keseluruhan model tetap mampu memberikan hasil klasifikasi dengan tingkat akurasi yang tinggi.

4.6.2 Pengujian Implementasi Sistem

Pada tahap ini dilakukan pengujian implementasi sistem untuk mengetahui kemampuan aplikasi dalam melakukan deteksi, klasifikasi, dan penyortiran cabai sesuai dengan rancangan yang telah dibuat. Pengujian dilakukan pada dua skenario, yaitu Upload Mode dan Realtime Mode.

Pengujian Upload Mode dilakukan untuk mengevaluasi kemampuan sistem dalam mengklasifikasikan citra cabai yang telah diambil diluar data training. Pengujian ini bertujuan untuk mengetahui konsistensi model BPNN dalam memberikan hasil prediksi terhadap citra yang sama ketika dilakukan pengujian berulang. Sebanyak 10 citra cabai yang mewakili 3 kelas, yaitu mentah, matang, dan busuk, digunakan sebagai data uji. Masing-masing citra diuji sebanyak 10 kali, sehingga diperoleh total 100 kali pengujian.

Tabel 10. Pengujian Upload Mode

Nama File	Label Aktual	Total Pengujian	Prediksi benar	Akurasi
	Matang	10	10	100%
	Matang	10	10	100%
	Mentah	10	10	100%
	Mentah	10	10	100%

	Mentah	10	10	100%
	Matang	10	10	100%
	Busuk	10	10	100%
	Busuk	10	10	100%
	Busuk	10	0	0%

	Matang	Matang	10	100%
--	--------	--------	----	------

Berdasarkan hasil pengujian diatas, sebanyak 9 dari 10 sampel cabai berhasil diprediksi dengan tingkat akurasi 100%, yang menunjukkan bahwa model BPNN mampu mengenali karakteristik warna dan tekstur cabai secara konsisten pada kelas mentah, matang, dan busuk. Namun, terdapat satu sampel cabai busuk tang memperoleh akurasi 0% karena seluruh pengujian sebanyak 10 kali menghasilkan prediksi yang tidak sesuai dengan label aktual. Kesalahan klasifikasi tersebut disebabkan oleh karakteristik visual cabai yang masih didominasi warna hijau, sedangkan bagian yang mengalami pembusukan hanya terdapat pada sebagian kecil permukaan buah. Selain itu fitur tekstur yang dihasilkan juga belum menunjukkan perbedaan yang cukup signifikan sehingga model BPNN mengelompokkan sampel tersebut ke dalam kelas mentah.

Selanjutnya dilakukan Pengujian pada Realtime mode dengan menjalankan seluruh komponen sistem secara bersamaan, meliputi kamera, conveyor, sensor proximity, motor servo, dan motor DC. Selama proses pengujian sistem berhasil memproses 118 buah cabai

Tabel 11. Pengujian Implementasi pada Realtime Mode

Parameter	Hasil
Lama Pengujian	15 Menit
Total cabai	118 buah
Jumlah sortir benar	106 buah
Jumlah salah sortir	12 buah
Tingkat keberhasilan	89%

Dari tabel tersebut, sistem berhasil melakukan penyortiran dengan benar sebanyak 106 cabai, sedangkan 12 cabai mengalami kesalahan penyortiran, sehingga tingkat keberhasilan sistem mencapai 89%. Kesalahan penyortiran disebabkan oleh dua faktor utama, faktor pertama adalah kondisi pencahayaan dari lingkungan, yang memengaruhi proses ekstraksi fitur warna sehingga hasil klasifikasi menjadi kurang konsisten. Faktor kedua adalah kecepatan conveyor yang terlalu tinggi, sehingga waktu yang tersedia untuk proses deteksi menjadi lebih singkat. Meskipun demikian, kecepatan conveyor dapat disesuaikan menggunakan potensiometer, sehingga sistem dapat dioptimalkan dengan mengurangi kecepatan conveyor agar proses deteksi dan penyortiran berlangsung lebih stabil.

Tingkat keberhasilan implementasi sebesar 89% lebih rendah dibandingkan hasil evaluasi model BPNN yang memperoleh akurasi sekitar 97%. Perbedaan ini terjadi karena pengujian model dilakukan menggunakan data uji yang telah melalui proses preprocessing dengan kondisi yang lebih terkontrol, sedangkan pada implementasi secara realtime sistem harus bekerja pada kondisi nyata yang melibatkan kamera, YOLO, proses ekstraksi fitur, BPNN, sensor proximity, conveyor, dan motor servo secara bersamaan. Adanya variasi pencahayaan, posisi cabai di atas conveyor, serta sinkronisasi antar perangkat menyebabkan hasil ekstraksi fitur tidak selalu sama dengan data pelatihan, sehingga performa sistem secara keseluruhan mengalami sedikit penurunan. Namun, tingkat keberhasilan yang diperoleh menunjukkan bahwa sistem telah mampu melakukan deteksi, klasifikasi, dan penyortiran cabai secara otomatis dengan kinerja yang baik

4.6.3 Pengujian Variasi Jumlah Epoch

Pengujian dilakukan untuk mengetahui pengaruh jumlah epoch terhadap performa model BPNN. Model dilatih dengan empat variasi epoch yaitu 100, 250, 500, dan 1000 epoch menggunakan konfigurasi arsitektur yang sama yaitu 10 input neurons, 15 hidden neuron, 3 output neurons, learning rate 0.05 dan momentum 0.9

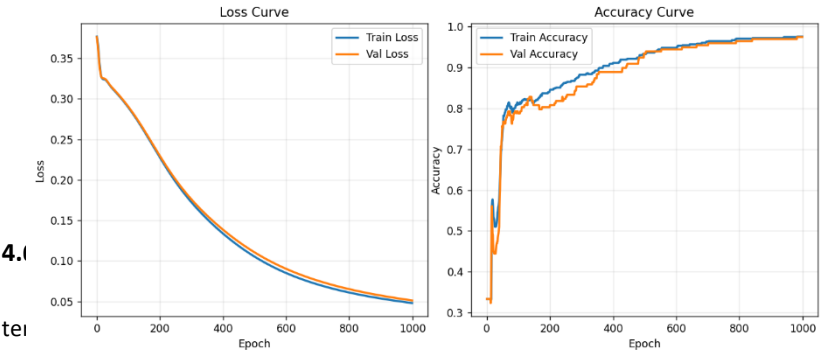
Tabel 12. Hasil Pengujian Variasi Jumlah Epoch

Epoch	Loss Value	Akurasi	Precision	Recall	F-1 Score
100	0,2907	0,78	0,8363	0,7878	0,7749
250	0,2006	0,82	0,8418	0,8232	0,8166
500	0,1107	0,93	0,9379	0,9343	0,9341
1000	0,0512	0,97	0,9753	0,9747	0,9747
1100	0,0586	0,96	0,9722	0,9696	0,9698
1200	0,0601	0,96	0,9722	0,9696	0,9698

Hasil pengujian menunjukkan bahwa peningkatan jumlah epoch berpengaruh sangat signifikan terhadap performa model. Pada epoch 100, model baru mencapai akurasi 78% dengan lose value 0.2907, menunjukkan bahwa model belum cukup belajar dari training.

Peningkatan yang signifikan terlihat pada epoch 500 dimana akurasi melonjak menjadi 93% dengan loss value turun drastis ke 0.1107. precision dan F-1 Score juga meningkat ke 0.937 dan 0.9341. Konfigurasi terbaik didapatkan pada epoch 1000 dengan akurasi mencapai 97%, loss 0.0512, precision 0.9753, dan F-1 Score 0.9747. Loss value yang terus menurun secara konsisten tanpa adanya peningkatan membuktikan tidak ada overfitting. Model terus belajar dan meningkatkan performa hingga 1000 epoch.

Berdasarkan hasil ini, konfigurasi optimal untuk sistem klasifikasi cabai adalah 1000 epoch yang memberikan keseimbangan terbaik dengan akurasi tertinggi dan loss terendah.



Gambar 20. Grafik Kurva Loss dan Accuracy

Tabel 13. Hasil Pengujian Variasi Jumlah Hidden Layer

No	Hidden Layer	Akurasi	Precision	Recall	F-1 Score
1	10	0,7474	0,8042	0,7474	0,7271
2	12	0,8939	0,9049	0,8939	0,8939
3	15	0,9747	0,9752	0,9747	0,9747

Berdasarkan hasil pengujian, terlihat bahwa model dengan 10 hidden neurons menghasilkan performa yang masih relatif rendah. Hal ini menunjukkan bahwa jumlah neuron yang terbatas belum mampu merepresentasikan pola kompleks pada data citra cabai secara optimal. Ketika jumlah *hidden neurons* ditingkatkan menjadi 12 neuron, terjadi peningkatan performa yang cukup signifikan, yang ditunjukkan oleh kenaikan nilai akurasi, presisi, recall, dan F1-score. Peningkatan ini menunjukkan bahwa penambahan neuron memberikan kapasitas yang lebih baik bagi jaringan dalam mempelajari karakteristik fitur warna dan tekstur cabai.

Konfigurasi 15 *hidden neurons* memberikan performa terbaik dengan nilai akurasi yang mencapai 97%. Selain itu, nilai presisi, recall, dan F1-score juga menunjukkan keseimbangan yang baik, yang menandakan bahwa model juga konsisten pada setiap kelas. Hasil ini menunjukkan bahwa jumlah *hidden neurons* yang lebih besar mampu meningkatkan kemampuan model dalam menangkap pola data yang lebih kompleks tanpa menyebabkan penurunan performa akibat overfitting. Oleh karena itu, konfigurasi dengan 15 *hidden neurons* dipilih sebagai arsitektur jaringan yang paling optimal pada penelitian ini

4.6.5 Evaluasi Metode Cross Validation

Pengujian menggunakan metode K-Fold Cross Validation dilakukan untuk mengevaluasi kemampuan generalisasi dan kestabilan performa model BPNN terhadap data yang belum pernah digunakan pada proses pelatihan. Metode ini dipilih karena mampu memberikan hasil evaluasi yang lebih representatif dibandingkan dengan satu kali pembagian data latih dan data uji. Pada penelitian ini digunakan 5 Fold Cross Validation, sehingga seluruh dataset dibagi menjadi lima bagian (fold) dengan jumlah data yang relatif sama. Pada setiap iterasi, satu Fold digunakan sebagai data uji, sedangkan empat fold lainnya digunakan sebagai data latih. Proses tersebut dilakukan sebanyak lima kali hingga setiap fold memperoleh kesempatan menjadi data uji tepat satu kali.

Tabel 14. Hasil Pengujian Menggunakan Cross Validation

No	Fold	Akurasi	Precision	Recall	F-1 Score
1	1	0,9848	0,9851	0,9848	0,9848
2	2	0,9495	0,9522	0,9495	0,9494
3	3	0,9798	0,9799	0,9798	0,9797
4	4	0,9798	0,9801	0,9798	0,9798
5	5	0,9545	0,9546	0,9545	0,9543
	rata-rata	0,96968	0,97038	0,96968	0,9696

Berdasarkan hasil pengujian, nilai akurasi pada masing-masing fold berada pada kisaran yang relatif tinggi dan tidak menunjukkan perbedaan yang signifikan antar fold. Fold 1 memberikan akurasi tertinggi sebesar 98%, sedangkan fold 2 memberikan akurasi terendah sebesar 94%. rentang perbedaan akurasi antar fold hanya sekitar 3.5% menunjukkan konsistensi performa yang sangat baik.

Nilai rata-rata akurasi yang diperoleh dari seluruh fold mencapai sekitar 96%, dengan nilai presisi, dan f1-score yang juga berada pada kisaran yang sama. Hasil ini menunjukkan bahwa model memiliki kestabilan performa yang baik serta mampu melakukan generalisasi dengan efektif terhadap data yang belum pernah dilihat sebelumnya. Dengan demikian, penggunaan metode cross validation membuktikan bahwa model BPNN yang dikembangkan tidak hanya memberikan performa tinggi pada satu pembagian data, tetapi juga konsisten ketika diuji pada berbagai skenario pembagian data

BAB 5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil penelitian mengenai implementasi metode Backpropagation Neural Network (BPNN) untuk klasifikasi kondisi cabai berbasis citra digital, dapat disimpulkan bahwa sistem yang dikembangkan berhasil mengklasifikasikan cabai ke dalam tiga kelas, yaitu mentah, matang, dan busuk. Sistem bekerja melalui beberapa tahapan, yaitu akuisisi citra, deteksi objek menggunakan YOLO, preprocessing yang meliputi proses resizing, segmentasi, penghilangan noise menggunakan median filter, serta konversi citra ke ruang warna HSV dan grayscale. Selanjutnya dilakukan ekstraksi fitur warna dan tekstur, normalisasi menggunakan Min-Max Scaling, kemudian proses klasifikasi menggunakan metode Backpropagation Neural Network.

Hasil pengujian model menggunakan confusion matrix menunjukkan bahwa model BPNN memperoleh akurasi sebesar 97%, dengan nilai precision sebesar 97,53%, recall sebesar 97,47%, F1-score sebesar 97,47%, dan loss value sebesar 0,0512. Hasil tersebut menunjukkan bahwa model mampu mengenali pola fitur warna dan tekstur cabai dengan baik sehingga menghasilkan performa klasifikasi yang tinggi. Selain itu, pengujian menggunakan metode 5-Fold Cross Validation menunjukkan bahwa model memiliki performa yang stabil pada setiap iterasi, sehingga kemampuan generalisasi model terhadap data yang belum pernah dilatih dapat dikatakan baik.

Pengujian variasi parameter menunjukkan bahwa konfigurasi model berpengaruh terhadap performa klasifikasi. Semakin besar jumlah epoch, nilai loss cenderung menurun dan akurasi model meningkat hingga mencapai kondisi yang optimal. Selain itu, hasil pengujian menunjukkan bahwa penggunaan 15 neuron pada hidden layer memberikan performa terbaik dibandingkan konfigurasi lainnya, sehingga konfigurasi tersebut dipilih sebagai arsitektur akhir model BPNN pada penelitian ini.

Pada tahap implementasi, sistem berhasil diintegrasikan dengan kamera, ESP32, sensor proximity, conveyor, motor DC, dan dua motor servo sehingga mampu melakukan proses deteksi, klasifikasi, dan penyortiran cabai secara otomatis. Pengujian Upload Mode menunjukkan bahwa 9 dari 10 sampel berhasil diprediksi dengan akurasi 100%, sedangkan satu sampel cabai busuk mengalami kesalahan klasifikasi karena karakteristik visualnya masih didominasi warna hijau. Sementara itu, pada pengujian Realtime Mode, sistem berhasil menyortir 106 dari 118 cabai dengan tingkat keberhasilan sebesar 89,83%. Perbedaan hasil tersebut dibandingkan akurasi model disebabkan oleh kondisi implementasi nyata yang dipengaruhi oleh perubahan pencahayaan, posisi objek pada conveyor, serta sinkronisasi proses deteksi dan aktuasi servo. Meskipun demikian, hasil tersebut menunjukkan bahwa sistem telah mampu bekerja dengan baik dan memiliki potensi untuk diterapkan sebagai alat bantu penyortiran cabai secara otomatis.

5.2 Saran

Untuk pengembangan penelitian selanjutnya, disarankan agar jumlah dataset citra cabai yang digunakan dapat diperbanyak dan dibuat lebih bervariasi agar model memiliki kemampuan generalisasi yang lebih baik terhadap kondisi cabai di lingkungan nyata. Selain itu, proses pengambilan citra juga dapat dilakukan dengan variasi pencahayaan dan latar belakang yang berbeda untuk meningkatkan ketahanan sistem terhadap kondisi lingkungan yang beragam.

Pengembangan sistem juga dapat diarahkan pada implementasi yang lebih praktis, seperti integrasi dengan perangkat kamera atau sistem berbasis aplikasi sehingga proses deteksi dan klasifikasi cabai dapat dilakukan secara lebih mudah dan realtime. Dengan pengembangan tersebut, sistem klasifikasi berbasis pengolahan citra ini diharapkan dapat memberikan kontribusi dalam membantu proses sortasi cabai secara lebih objektif, cepat, dan efisien.

Daftar Pustaka

- [1] R. Melani, D. Azmi, I. Muslem, and M. Amin, "Klasifikasi Kematangan Cabai Berdasarkan Citra Digital Menggunakan Algoritma K-Nearest Neighbor".
- [2] A. Baso Kaswar, F. Adiba, and D. Darma Andayani, "Sistem Klasifikasi Tingkat Kematangan Buah Cabai Katokkon Berdasarkan Fitur Warna LAB Menggunakan Artificial Neural Network Backpropagation," 2023.
- [3] Z. Duan *et al.*, "A CNN model for early detection of pepper Phytophthora blight using multispectral imaging, integrating spectral and textural information," *Plant Methods*, vol. 20, no. 1, Dec. 2024, doi: 10.1186/s13007-024-01239-7.
- [4] A. Herdiansah, R. I. Borman, D. Nurnaningsih, A. A. J. Sinlae, and R. R. Al Hakim, "Klasifikasi Citra Daun Herbal Dengan Menggunakan Backpropagation Neural Networks Berdasarkan Ekstraksi Ciri Bentuk," *JURIKOM (Jurnal Riset Komputer)*, vol. 9, no. 2, p. 388, Apr. 2022, doi: 10.30865/jurikom.v9i2.4066.
- [5] A. Susanto and I. U. Wahyu Mulyono, "A Good Accuracy in Apple Fruits Quality Based on Back Propagation Neural Network and Feature Extraction," *JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING*, vol. 6, no. 1, pp. 38–48, Jul. 2022, doi: 10.31289/jite.v6i1.6938.
- [6] M. A. Aldiansyah, "PEMROSESAN CITRA DIGITAL UNTUK KLASIFIKASI TANAMAN CABAI MENGGUNAKAN JARINGAN SARAF TIRUAN BACKPROPOGATION," *Jurnal Teknologi Informasi dan Terapan*, vol. 5, no. 1, pp. 31–36, Apr. 2019, doi: 10.25047/jtit.v5i1.76.
- [7] "Journal of Engineering - 2024 - Khan - Automated Tomato Leaf Disease Detection Using Image Processing An SVM-Based".
- [8] U. Mahdiah, "KLASIFIKASI KUALITAS CITRA CABAI DENGAN MENGGUNAKAN ALGORITMA GRADIEN BOOSTING," *JAMI: Jurnal Ahli Muda Indonesia*, vol. 4, no. 1, pp. 61–69, Jun. 2023, doi: 10.46510/jami.v4i1.137.
- [9] G. Dwi Cahyo, N. Ahmadi Thobroni, T. Pinarung Gusti, and R. Wulanningrum, "Penggunaan OpenCV untuk Segmentasi Cabai menggunakan Konversi RGB ke HSV," 2025.
- [10] Siti Saniah and Mhd. Furqan, "Classification Of Rice Plant Diseases Using K-Nearest Neighbor Algorithm Based On Hue Saturation Value Color Extraction And Gray Level Co-Occurrence

- Matrix Features,” *JURNAL TEKNOLOGI DAN OPEN SOURCE*, vol. 7, no. 2, pp. 212–223, Dec. 2024, doi: 10.36378/jtos.v7i2.3972.
- [11] S. S. Zuain, H. Fitriyah, and R. Maulana, “Deteksi Penyakit pada Daun Cabai berdasarkan Fitur HSV dan GLCM menggunakan Algoritma C4.5 berbasis Raspberry Pi,” 2021. [Online]. Available: <http://j-ptiik.ub.ac.id>
 - [12] A. Fauzi, “Pengurangan Derau (Noise) Pada Citra Paper Dokumen menggunakan Metode Gaussian Filter dan Median Filter Oleh : Achmad Fauzi 7 Pengurangan Derau (Noise) pada Citra Paper Dokumen menggunakan Metode Gaussian Filter dan Median Filter,” 2022.
 - [13] S. Chairunnissa, D. Akbar, 李, N. Yasmin, and A. Ramadhanu, “Pengelolaan Citra Cabai Keriting: Kombinasi Median Filtering dan Algoritma K-Means untuk Pengelompokan Berbasis Fitur,” *Journal of Education Research*, vol. 5, no. 4.
 - [14] R. Ramli and A. A. Riadi, “Classification of Rice Leaf Diseases Using Support Vector Machine with HSV and GLCM-Based Feature Extraction,” 2025. [Online]. Available: <http://jurnal.polibatam.ac.id/index.php/JAIC>
 - [15] P. Atribut Mutu dan Umur Simpan Beberapa Jenis Cabai Pada Berbagai Kemasan dan Suhu Penyimpanan *et al.*, “Darkam Musaddad et al Perubahan Atribut Mutu dan Umur Simpan Beberapa Jenis Cabai Pada Berbagai Kemasan dan Suhu Penyimpanan (The Quality Attribute Change and Shelf Life of Several Types of Chili on Various Packaging and Storage Temperature).”
 - [16] Subiyanto, “SISTEM KOMPUTASI JARINGAN SYARAF TIRUAN BACKPROPAGATION,” 2010.
 - [17] G. Firgiawan, N. Lintang Seina, and P. Rosyani, “Implementasi Metode You Only Look Once (YOLO) untuk Pendeteksi Objek dengan Tools OpenCV,” 2024. [Online]. Available: <https://jurnalmahasiswa.com/index.php/aidanspk>
 - [18] C. M. Badgujar, A. Poullose, and H. Gan, “Agricultural Object Detection with You Look Only Once (YOLO) Algorithm: A Bibliometric and Systematic Literature Review.”
 - [19] A. H. Azzam Ash’shobir, K. G. Putri Harli, A. P. Putri Rudi, I. G. Syah Putro, and O. D. Putra Cahyono, “Sistem Deteksi Kualitas Cabai Rawit Menggunakan Metode YOLO,” *Modem : Jurnal Informatika dan Sains Teknologi.*, vol. 3, no. 1, pp. 114–132, Jan. 2025, doi: 10.62951/modem.v3i1.363.
 - [20] M. Prediksi Stock Barang, N. Adhi Santoso, N. Fadilah, R. Dwi Kurniawan, A. Supratman, and S. YMI TEGAL Kota Tegal, “Penerapan Neural Network Method dengan Struktur Backpropagation dalam,” *Remik: Riset dan E-Jurnal Manajemen*

- Informatika Komputer*, vol. 7, no. 3, 2023, doi: 10.33395/remik.v7i3.12828.
- [21] M. R. Lubis, "ANALISIS JARINGAN SARAF TIRUAN BACKPROPAGATION UNTUK PENINGKATAN AKURASI PREDIKSI HASIL PERTANDINGAN SEPAKBOLA," *TECHSI - Jurnal Teknik Informatika*, vol. 10, no. 1, pp. 50–62, May 2018, doi: 10.29103/techsi.v10i1.560.
 - [22] N. Nurmila and A. Sugiharto, "ALGORITMA BACK PROPAGATION NEURAL NETWORK UNTUK PENGENALAN POLA KARAKTER HURUF JAWA."
 - [23] D. Andini, "Bulletin of Artificial Intelligence Jaringan Syaraf Tiruan Untuk Klasifikasi Penyakit Demam Menggunakan Algoritma Backpropagation," vol. 2, no. 1, 2023, [Online]. Available: <https://journal.grahamitra.id/index.php/buai>
 - [24] M. Azhima, I. Afrianty, E. Budianita, and S. Kurnia Gusti, "KLIK: Kajian Ilmiah Informatika dan Komputer Penerapan Metode Backpropagation Neural Network untuk Klasifikasi Penyakit Stroke," *Media Online*, vol. 4, no. 6, pp. 3013–3021, 2024, doi: 10.30865/klik.v4i6.1956.
 - [25] I. Saluza, L. Widya Astuti, and E. Yulianti, "Ensemble Backpropagation Neural Network Dalam Memprediksi Inflasi."

Biodata



Nama : Christoper Modji
TTL : Batam, 29 Mei 2004
Agama : Kristen Protestan
Alamat : Graha Nusa Batam Blok D1 no 27

Email : cmodji@gmail.com
Riwayat Pendidikan SMA/SMK : SMA Negeri 17 Batam
SMP : SMP Kristen Basic

Lampiran

Link GDRIVE HASIL PENELITIAN