

RANCANG BANGUN SISTEM KLASIFIKASI KESEGERAN DAGING SAPI MENGUNAKAN NEURAL NETWORK TERKOMPRESI PADA MIKROKONTROLER ESP32 DENGAN SENSOR GAS MQ DAN SENSOR WARNA RGB

Ricky Dwi Rudianto; Fajar Suryawan, S.T., M.Eng.Sc., Ph.D

Program Studi Teknik Elektro, Fakultas Teknik,

Universitas Muhammadiyah Surakarta

Abstrak

Kesegaran daging sapi merupakan faktor penting yang memengaruhi kualitas dan keamanan pangan. Penilaian kesegaran daging yang dilakukan secara manual melalui pengamatan warna dan aroma masih bersifat subjektif sehingga berpotensi menimbulkan kesalahan. Oleh karena itu, penelitian ini bertujuan untuk merancang dan membangun sistem klasifikasi kesegaran daging sapi menggunakan sensor gas MQ-135 dan sensor warna TCS3200 dengan metode *Neural Network* yang diimplementasikan pada mikrokontroler ESP32. Data sensor diproses melalui beberapa tahapan praproses, yaitu filter *Infinite Impulse Response* (IIR) untuk mengurangi noise, konversi ruang warna RGB ke HSV, serta normalisasi data sebelum digunakan sebagai masukan model. Fitur yang digunakan meliputi nilai R, G, B, *Saturation (S)*, *Value (V)*, dan MQ. Model *Neural Network* dibangun menggunakan *TensorFlow* dan dikonversi ke format *TensorFlow Lite* agar dapat dijalankan pada ESP32. Hasil evaluasi model menunjukkan akurasi sebesar 91,10% dengan nilai *weighted average F1-score* sebesar 0,91. Pengujian sistem secara langsung menghasilkan akurasi sebesar 83,3%, dengan tingkat keberhasilan deteksi daging busuk mencapai 100%. Hasil penelitian menunjukkan bahwa kombinasi sensor gas, sensor warna, dan *Neural Network* mampu mengklasifikasikan tingkat kesegaran daging sapi secara akurat serta dapat diimplementasikan secara *real-time* pada perangkat *embedded* tanpa memerlukan server eksternal.

Kata Kunci: *Neural Network*, *TensorFlow Lite*, Klasifikasi Kesegaran Daging, Sensor Gas MQ, Sensor Warna RGB, ESP32

Abstract

Beef segariness is an important factor that affects food quality and safety. Segariness assessment performed manually through visual observation and odor evaluation is still subjective and may lead to inaccurate judgments. Therefore, this study aims to design and develop a beef segariness classification system using an MQ-135 gas sensor and a TCS3200 color sensor combined with a Neural Network method implemented on an ESP32 microcontroller. Sensor data were processed through several preprocessing stages, including the application of an Infinite Impulse Response (IIR) filter to reduce noise, RGB-to-HSV color space conversion, and data normalization before being used as model inputs. The input features consisted of R, G, B, Saturation (S), Value (V), and MQ values. The Neural Network model was developed using TensorFlow and converted into TensorFlow Lite format to enable deployment on the ESP32 microcontroller. The evaluation results showed an accuracy of 91.10% with a weighted average F1-score of 0.91. Direct testing of the system achieved an accuracy of 83.3%, with a 100% success rate in identifying spoiled beef samples. The results indicate that the combination of gas sensing, color sensing, and Neural Network methods is capable of accurately classifying beef segariness levels. Furthermore, the implementation on ESP32 enables real-time classification without requiring an external server, making the system suitable for embedded applications.

Keywords: *Neural Network, TensorFlow Lite, Beef Segariness Classification, MQ Gas Sensor, RGB Color Sensor, ESP32*

1. PENDAHULUAN

Kualitas daging sapi merupakan faktor penting dalam menjamin keamanan pangan dan kesehatan masyarakat. Daging yang tidak segar berpotensi menyebabkan gangguan kesehatan, seperti diare dan keracunan makanan, akibat pertumbuhan mikroorganisme serta peningkatan kadar senyawa volatil seperti amonia. Namun, penilaian tingkat kesegaran daging di masyarakat masih banyak dilakukan secara manual melalui pengamatan warna dan penciuman aroma. Metode tersebut bersifat subjektif dan rentan terhadap kesalahan karena sangat bergantung pada kemampuan dan pengalaman masing-masing individu. Berbagai penelitian telah dilakukan untuk mengembangkan sistem pendeteksi kesegaran daging berbasis sensor. Salah satu penelitian yang berkaitan dengan deteksi kesegaran daging dilakukan oleh (Rahmana & Syauqy, 2019) melalui penelitian berjudul *Sistem Deteksi Lama Waktu Penyimpanan Daging Ayam Berdasarkan Warna dan Kadar Amonia Berbasis Sensor TCS3200 dan MQ135 dengan Metode Jaringan Syaraf Tiruan*. Penelitian tersebut menggunakan sensor warna TCS3200 dan sensor gas MQ135 dengan metode Jaringan Syaraf Tiruan untuk mendeteksi lama waktu penyimpanan daging ayam. Hasil penelitian menunjukkan bahwa kombinasi parameter warna dan kadar amonia dapat digunakan sebagai indikator tingkat kesegaran daging. Keberhasilan penelitian tersebut menunjukkan bahwa parameter warna dan kadar amonia memiliki potensi untuk diterapkan pada sistem klasifikasi kesegaran daging sapi menggunakan metode *Neural Network* berbasis *embedded system*.

Penelitian mengenai klasifikasi kesegaran daging telah banyak dilakukan dengan memanfaatkan parameter warna, kadar gas amonia, serta metode machine learning sebagai algoritma klasifikasi. Salah satu penelitian yang memanfaatkan teknologi sensor dilakukan oleh (Firmansyah et al., 2019) melalui penelitian berjudul *Implementasi Sistem Penentuan Kesegaran Daging Sapi Lokal Berdasarkan Warna dan Kadar Amonia Dengan Metode Jaringan Saraf Tiruan Berbasis Embedded System*. Sistem yang dikembangkan menggunakan Arduino sebagai pengolah data sensor warna dan kadar amonia, serta *Neural Network* sebagai metode klasifikasi. Hasil penelitian menunjukkan bahwa pendekatan berbasis *embedded system* mampu mengolah data sensor secara langsung untuk menentukan tingkat kesegaran daging sapi. Selain itu, pendekatan berbasis sensor gas juga telah dikembangkan oleh (R. Hidayat, 2024) melalui penelitian berjudul *Implementasi Sensor Gas untuk Mendeteksi Kesegaran Daging Menggunakan Metode Neural Network*. Penelitian tersebut memanfaatkan beberapa sensor gas

untuk membaca gas yang dihasilkan selama proses degradasi daging dan mengolahnya menggunakan Neural Network untuk membangun model prediksi tingkat kesegaran. Penelitian lain yang dilakukan oleh (Rahmawati et al., 2021.) mengembangkan sistem *electronic nose* berbasis *Neural Network* yang menunjukkan tingkat keberhasilan klasifikasi yang cukup tinggi. Hasil penelitian tersebut menunjukkan bahwa metode *Neural Network* memiliki kemampuan yang baik dalam mengenali pola data yang berasal dari sensor gas.

Dalam proses identifikasi kesegaran daging, sensor gas MQ135 dan sensor warna TCS3200 merupakan dua jenis sensor yang banyak digunakan dalam berbagai penelitian. Sensor MQ135 berfungsi untuk mendeteksi gas amonia yang dihasilkan selama proses pembusukan daging, sedangkan sensor TCS3200 digunakan untuk mendeteksi perubahan warna pada permukaan daging. Kemampuan sensor TCS3200 dalam membaca berbagai variasi warna telah dibuktikan pada penelitian yang dilakukan oleh (Firmansyah et al., 2019). Penelitian tersebut menjelaskan bahwa sensor TCS3200 mampu membaca beberapa jenis warna melalui proses pembacaan yang dilakukan secara bertahap sehingga dapat digunakan untuk mengidentifikasi perubahan karakteristik warna daging. Selain pendekatan berbasis sensor gas dan warna, metode pengolahan citra digital juga banyak digunakan dalam klasifikasi kualitas daging. Salah satu penelitian yang menerapkan pendekatan tersebut dilakukan oleh (Asmoro & Solichin, 2023) menggunakan metode *Convolutional Neural Network* (CNN) untuk klasifikasi kualitas daging sapi berbasis Android. Meskipun menghasilkan tingkat akurasi yang tinggi, metode CNN membutuhkan kapasitas penyimpanan dan komputasi yang relatif besar sehingga kurang sesuai untuk diterapkan secara langsung pada mikrokontroler dengan sumber daya terbatas.

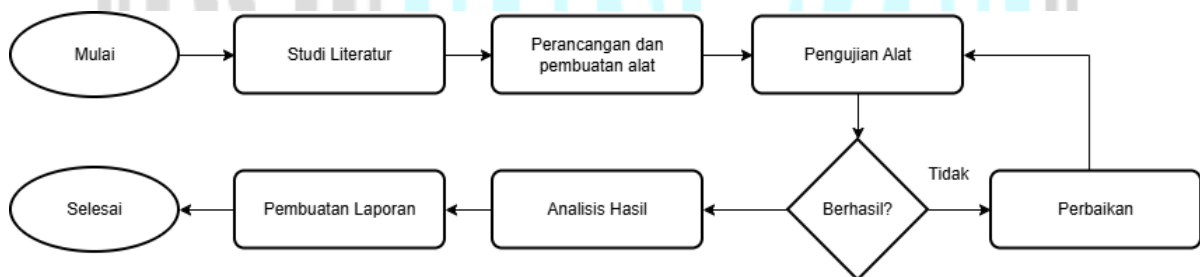
Berdasarkan penelitian-penelitian terdahulu, dapat diketahui bahwa parameter warna dan kadar amonia merupakan indikator yang efektif untuk menentukan tingkat kesegaran daging. Selain itu, metode *Neural Network* terbukti mampu melakukan klasifikasi data sensor dengan baik pada sistem embedded. Namun, implementasi model Neural Network yang ringan dan efisien pada mikrokontroler ESP32 untuk mengklasifikasikan kesegaran daging sapi ke dalam tiga kategori, yaitu segar, semi segar, dan busuk, masih memerlukan pengembangan lebih lanjut. Penderita gangguan penciuman memiliki keterbatasan dalam mendeteksi aroma pembusukan secara alami sehingga membutuhkan sistem pendukung yang mampu melakukan penilaian secara objektif. Berdasarkan permasalahan tersebut, penelitian ini mengusulkan perancangan dan pembangunan sistem klasifikasi kesegaran daging sapi menggunakan metode *Neural Network* pada mikrokontroler ESP32. Sistem memanfaatkan sensor gas MQ135 dan sensor warna TCS3200

untuk memperoleh parameter fisik dan kimia daging, kemudian mengklasifikasikannya ke dalam tiga kategori, yaitu segar, semi segar, dan busuk. Sistem ini diharapkan dapat menjadi solusi yang objektif, praktis, dan mampu bekerja secara *real-time* untuk membantu masyarakat dalam mengevaluasi tingkat kesegaran daging sapi.

2. METODE

Metode penelitian yang digunakan dalam penelitian ini adalah metode rancang bangun sistem, yang bertujuan untuk merancang dan mengimplementasikan sistem klasifikasi kesegaran daging sapi secara menyeluruh, mulai dari akuisisi data sensor, pelatihan model *Neural Network*, hingga *deployment* model pada mikrokontroler ESP32. Tahapan penelitian disusun secara sistematis sesuai dengan diagram alir yang ditunjukkan Gambar 1. Dan gambar 2 merupakan diagram alir tahapan *machine learning*

Berdasarkan Gambar 1, metode penelitian yang dilakukan terdiri dari beberapa tahapan, yaitu sebagai berikut :



Gambar 1. Diagram alir metode pembuatan alat

a. Studi Literatur

Tahap studi literatur dilakukan untuk memperoleh dasar teori dan referensi yang relevan dengan penelitian. Pada tahap ini, peneliti mengumpulkan dan mempelajari berbagai sumber pustaka seperti jurnal ilmiah, buku, dan artikel ilmiah terkait sistem deteksi kesegaran daging, sensor gas MQ, sensor warna RGB, metode *Neural Network*, serta implementasi *TensorFlow Lite* pada mikrokontroler. Hasil dari tahap ini digunakan sebagai landasan dalam perancangan dan pengembangan sistem.

b. Perancangan dan pembuatan alat

merupakan tahap inti dalam penelitian ini. Pada tahap perancangan dilakukan penyusunan arsitektur sistem secara keseluruhan, meliputi alur input, proses, dan output sistem. Input berupa data sensor gas MQ dan sensor warna RGB yang dibaca secara real-time dari sampel daging. Data tersebut kemudian diproses melalui filter IIR dan konversi HSV sebelum dimasukkan ke model *Neural Network* untuk menghasilkan hasil klasifikasi. Output dari sistem

berupa tampilan kategori kesegaran daging pada LCD 20x4 dan aktivasi buzzer sebagai indikator peringatan. Setelah tahap perancangan selesai, dilakukan pembuatan alat dengan merakit perangkat keras sesuai desain yang telah dibuat serta mengembangkan perangkat lunak pendukung. Perangkat lunak yang dikembangkan meliputi program pembacaan sensor, praproses data, serta inferensi model *Neural Network* yang dijalankan langsung pada ESP32 menggunakan library *TensorFlow Lite*

c. Pengujian Alat

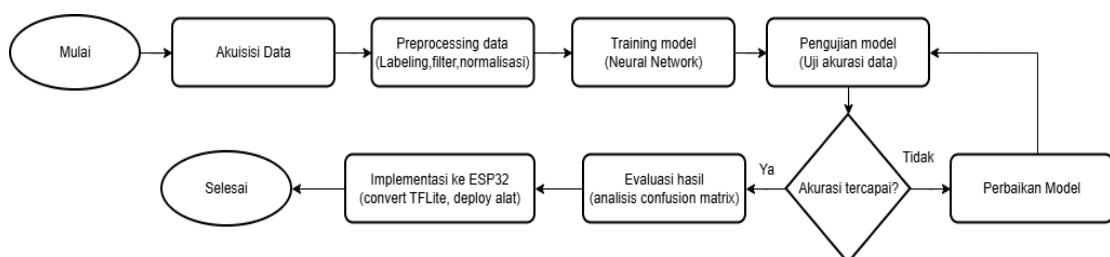
Tahap pengujian alat dilakukan untuk mengetahui apakah sistem yang telah dibuat dapat bekerja sesuai dengan perancangan. Pengujian dilakukan menggunakan sampel daging sapi asli pada tiga kondisi kesegaran : segar, semi, dan busuk. Pengujian meliputi kemampuan sistem dalam membaca data sensor, mengklasifikasikan kondisi daging secara *real-time*, serta ketepatan tampilan LCD dan respon *buzzer*. Pengujian juga dilakukan dengan variasi kondisi sampel untuk mengevaluasi ketahanan sistem.

d. Analisis hasil

Setelah proses pengujian selesai, dilakukan analisis terhadap data hasil pengujian. Analisis hasil bertujuan untuk mengevaluasi performa sistem yang telah dibuat. Parameter yang dianalisis meliputi akurasi klasifikasi, *precision*, *recall*, dan *F1-score* model *Neural Network*, serta waktu proses sistem dari proses pembacaan sensor hingga output ditampilkan. Hasil analisis digunakan untuk mengetahui kelebihan dan keterbatasan sistem serta sebagai dasar dalam penarikan kesimpulan penelitian.

e. Pembuatan laporan

Tahap pembuatan laporan merupakan tahap akhir dari penelitian. Pada tahap ini, seluruh proses penelitian mulai dari studi literatur, perancangan dan pembuatan alat, pengujian, hingga analisis hasil disusun dalam bentuk laporan Tugas Akhir, selain itu hasil penelitian juga dituangkan dalam bentuk naskah publikasi ilmiah sesuai dengan pedoman penulisan yang berlaku.



Gambar 2. Diagram alir tahapan *machine learning*

Gambar 2 menunjukkan tahapan machine learning yang diterapkan dalam penelitian ini terdiri dari beberapa tahap sebagai berikut :

a. Akuisisi Data

Tahap akuisisi data merupakan proses pengumpulan data mentah dari sensor secara langsung terhadap sampel daging sapi. Data diperoleh dari dua sensor utama, yaitu sensor gas MQ-135 yang membaca kadar senyawa volatil dan sensor warna TCS3200 yang membaca nilai RGB permukaan daging. Proses pengumpulan data menghasilkan total 953 sampel yang kemudian disimpan dalam format *spreadsheet* untuk diproses pada tahap selanjutnya.

b. *Preprocessing* Data

Tahap *preprocessing* bertujuan untuk mempersiapkan data mentah agar siap digunakan dalam pelatihan model. Tahap ini terdiri dari tiga proses utama. Pertama, penentuan label dilakukan melalui dua tahap, yaitu tahap kalibrasi dan tahap otomasi. Pada tahap kalibrasi, dilakukan observasi langsung terhadap kondisi fisik daging (warna, tekstur, dan aroma) pada setiap rentang waktu penyimpanan untuk mengidentifikasi pola nilai sensor R dan MQ yang berasosiasi dengan masing-masing kategori kesegaran. Pada tahap otomasi, ambang batas (*threshold*) nilai R dan MQ yang diperoleh dari hasil kalibrasi tersebut diterapkan sebagai aturan pelabelan pada keseluruhan dataset, sehingga setiap sampel dapat dikategorikan secara konsisten ke dalam kelas segar, semi, atau busuk tanpa perlu observasi manual pada setiap sampel. Kedua, filter IIR dengan koefisien alpha 0,05 diterapkan untuk menghaluskan *noise* pada sinyal sensor. Ketiga, nilai RGB hasil filter dikonversi ke ruang warna HSV untuk menghasilkan representasi warna yang lebih stabil, sehingga fitur akhir yang digunakan dalam pelatihan model terdiri dari enam variabel, yaitu R, G, B, S, V, dan MQ, yang kemudian dinormalisasi menggunakan *StandardScaler*. Perlu dicatat bahwa meskipun label diturunkan menggunakan ambang batas pada variabel R dan MQ, model *Neural Network* tetap dilatih menggunakan kombinasi keenam fitur tersebut, sehingga model perlu mempelajari batas keputusan yang lebih kompleks dan tidak sekadar mereplikasi aturan ambang batas tunggal, khususnya pada kondisi nilai sensor yang mendekati batas kategori akibat variasi dan *noise* pengukuran di lingkungan nyata. Pendekatan yang mengombinasikan metode berbasis ambang batas dengan model *Neural Network* pada fitur pengukuran yang sama juga telah diterapkan pada penelitian sejenis di bidang klasifikasi kesegaran pangan, di mana model *Neural Network* menunjukkan performa yang lebih baik dibandingkan model berbasis ambang batas semata (Beyza et al., 2024).

c. *Training Model*

Pelatihan model dilakukan menggunakan arsitektur *Neural Network* yang terdiri dari tiga *hidden layer*, yaitu 64 unit dengan aktivasi *tanh*, 32 unit dengan aktivasi *tanh*, dan 16 unit dengan aktivasi *relu*, serta satu *output layer* dengan tiga unit aktivasi *softmax* yang merepresentasikan tiga kelas kesegaran. Model dilatih menggunakan optimizer Adam dengan *learning rate* 0,01, *batch size* 32, dan maksimum 300 *epoch*. Mekanisme *Early Stopping* diterapkan untuk mencegah *overfitting* dan menjaga efisiensi pelatihan.

d. *Pengujian Model*

Pengujian model dilakukan menggunakan data uji sebesar 20% dari total dataset untuk mengukur akurasi model terhadap data yang belum pernah dilihat selama pelatihan. Apabila akurasi yang diperoleh belum memenuhi target, dilakukan perbaikan model melalui proses *hyperparameter tuning* menggunakan Keras Tuner *RandomSearch* hingga performa model mencapai hasil yang optimal.

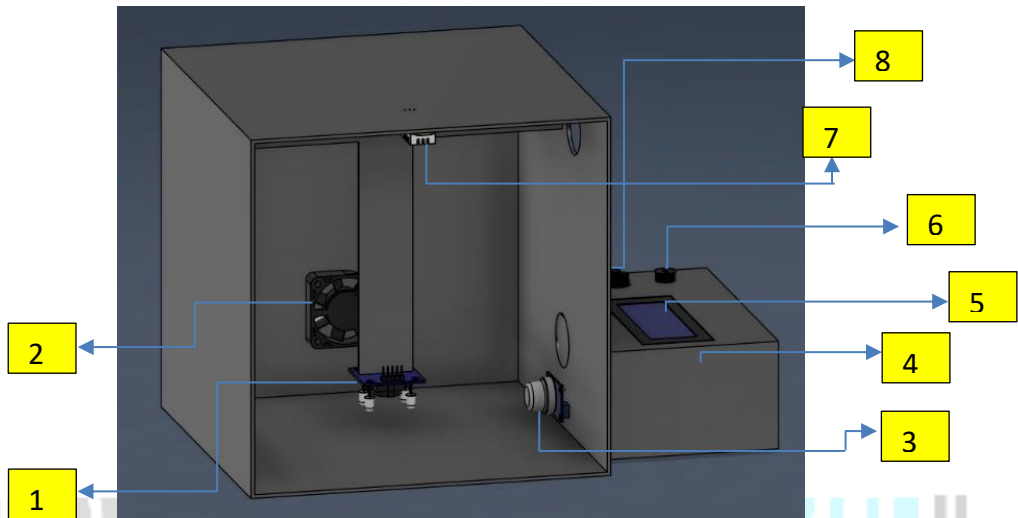
e. *Evaluasi Hasil*

Evaluasi hasil dilakukan menggunakan analisis *confusion matrix* untuk menilai performa model secara menyeluruh pada setiap kelas. Selain itu, dihitung pula metrik *precision*, *recall*, dan *F1-score* sebagai ukuran kuantitatif terhadap kemampuan model dalam mengklasifikasikan setiap kategori kesegaran daging secara tepat.

f. *Implementasi ke ESP32*

Tahap akhir adalah implementasi model ke mikrokontroler ESP32. Model yang telah dilatih dikonversi dari format Keras ke format *TensorFlow Lite* untuk mereduksi ukuran model agar sesuai dengan keterbatasan memori ESP32. Model yang telah dikonversi kemudian diekspor sebagai array C dan ditanamkan langsung ke dalam program ESP32 sehingga inferensi dapat dilakukan secara lokal tanpa ketergantungan pada server eksternal.

2.1. Perancangan *Hardware*

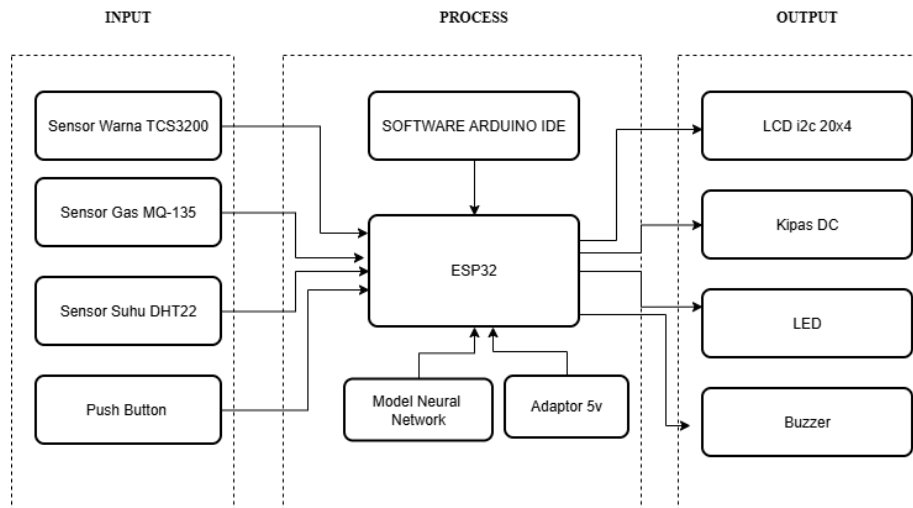


Gambar 3. Desain sketsa 3d alat

Gambar 3 menunjukkan desain sketsa 3d alat pada penelitian ini untuk komponen di antaranya yaitu :

Tabel 1. Keterangan komponen pada desain alat

NO	Keterangan Komponen
1	Sensor warna TCS3200
2	Kipas angin dc 5v
3	Sensor gas MQ-135
4	Box elektronika
5	LCD I2C
6	Tombol mulai klasifikasi
7	Sensor suhu DHT22
8	Tombol kipas dc



Gambar 4. Blok diagram sistem

Gambar 4 memperlihatkan blok diagram sistem, terdapat tiga bagian utama yaitu *input*, *proses*, dan *output*

1. Bagian input:
 - a. Sensor Warna TCS3200 berfungsi untuk membaca parameter warna permukaan daging sapi secara *real-time*. Data warna yang diperoleh kemudian di konversi ke ruang warna HSV untuk menghasilkan representasi warna yang lebih stabil terhadap variasi intensitas cahaya.
 - b. Sensor Gas MQ-135 berfungsi sebagai perangkat input untuk mendeteksi senyawa volatil seperti amonia yang dihasilkan selama proses degradasi daging sapi.
 - c. Sensor Suhu DHT22 berfungsi untuk mengukur kondisi lingkungan sekitar. Suhu dan kelembapan merupakan faktor pendukung yang mempengaruhi laju pembusukan daging
 - d. Terdapat 2 push button yang masing-masing berfungsi sebagai :
 1. Button pertama mulai klasifikasi digunakan untuk memicu proses akuisisi data sensor dan inferensi model Neural Network ketika tombol ini ditekan sistem akan secara otomatis membaca nilai dari sensor gas MQ-135, sensor warna TCS3200 dan sensor suhu DHT22, kemudian melakukan praproses data serta menjalankan inferensi model TensorFlow Lite untuk menghasilkan klasifikasi kesegaran daging.
 2. Button kedua berfungsi untuk mengaktifkan kipas DC secara manual setelah proses klasifikasi selesai. Kipas ini berguna untuk membersihkan sisa gas volatil dari ruang pengujian sebelum dilakukan klasifikasi berikutnya, sehingga pembacaan sensor pada pengujian selanjutnya tidak terkontaminasi oleh residu gas dari sampel sebelumnya.

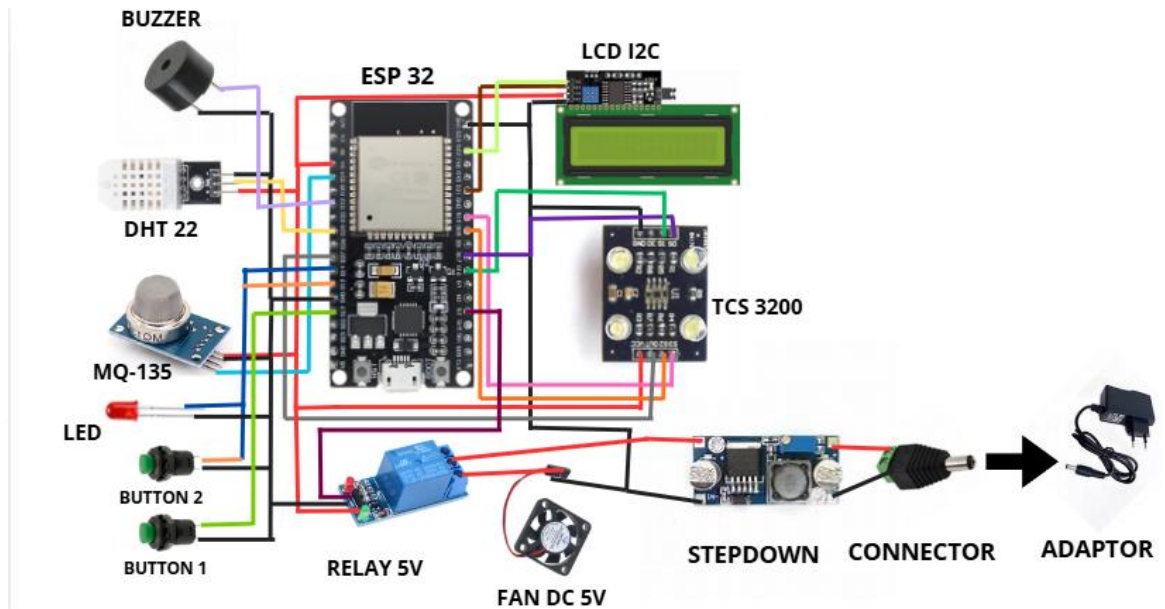
2. Bagian proses:

- a. *Software* Arduino Ide berfungsi sebagai menulis dan mengunggah program ke ESP32 program ini bertugas untuk mengatur logika pengendalian perangkat keras, membaca *input push button*, serta memproses model *neural network*
- b. ESP32 bertindak sebagai unit pemrosesan utama pada sistem ini ESP32 dipilih karena memiliki prosesor dual-core dengan kecepatan clock hingga 240 MHz memori SRAM sebesar 520 KB, serta dukungan flash eksternal hingga 4 MB yang cukup untuk menyimpan model TensorFlow Lite, ESP32 bertugas membaca data dari seluruh sensor dan menjalankan inferensi model *Neural Network* secara lokal tanpa ketergantungan pada server eksternal serta mengendalikan semua perangkat output seperti LCD , buzzer, LED, dan Kipas DC.
- c. Model *Neural Network* yang digunakan telah dilatih menggunakan google colab dan dikonversi ke format *TensorFlow Lite* agar dapat dijalankan secara efisien pada ESP32 dengan sumber daya komputasi yang terbatas. Model menerima input berupa vektor fitur gabungan dari data sensor gas dan data warna yang telah dikonversi ke ruang warna HSV kemudian menghasilkan prediksi terhadap tiga kelas kesegaran daging, kelas dengan nilai probabilitas tertinggi dipilih sebagai hasil klasifikasi akhir yang ditampilkan.

3. Bagian *output*:

- a. LCD I2C 20x4 berfungsi untuk menampilkan informasi teks mengenai hasil klasifikasi daging sapi secara real-time. LCD menampilkan 4 baris teks sehingga dapat menampilkan beberapa informasi sekaligus meliputi hasil klasifikasi kesegaran, nilai pembacaan sensor gas, suhu dan juga warna antarmuka I2C yang digunakan hanya memerlukan 2 jalur komunikasi (SDA dan SCL) sehingga menghemat penggunaan pin GPIO pada ESP32.
- b. Kipas DC berfungsi sebagai sistem ventilasi aktif untuk membersihkan ruang pengujian dari akumulasi gas volatil setelah setiap sesi klasifikasi. Keberadaan kipas sangat penting untuk menjaga keakuratan pembacaan sensor MQ-135 pada pengujian berikutnya karena gas amonia dan senyawa volatil lainnya yang tertinggal dapat mempengaruhi *baseline* sensor dan menyebabkan hasil klasifikasi yang tidak akurat, kipas DC dikendalikan oleh ESP32 melalui modul relay dan diaktifkan melalui push button kedua setelah proses klasifikasi selesai.
- c. LED berfungsi sebagai indikator visual yang menyala secara bersamaan ketika kipas DC diaktifkan, memberikan umpan balik langsung kepada pengguna bahwa sistem sedang dalam mode ventilasi. Penggunaan LED sebagai indikator membantu pengguna memantau status operasi sistem.

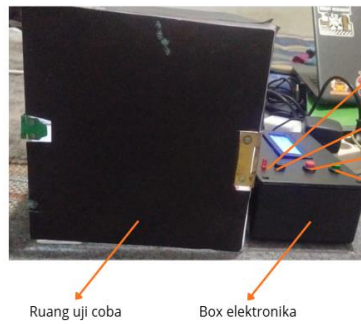
- d. *Buzzer* berfungsi sebagai indikator peringatan berbasis suara ketika sistem mendeteksi daging dalam kondisi busuk, bunyi buzzer memberikan notifikasi yang dapat didengar dari jarak tertentu tanpa harus secara aktif memantau layar LCD, sehingga memperkuat efektivitas sistem sebagai alat peringatan dini.



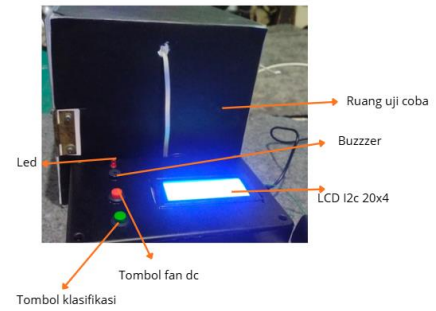
Gambar 5. Wiring diagram

Gambar 5 *Wiring* diagram tersebut menggambarkan hubungan antar komponen pada alat klasifikasi kesegaran daging sapi, ESP32 berfungsi sebagai pusat pengendali utama yang terhubung dengan sensor warna TCS3200, sensor gas MQ-135, sensor suhu DHT22, push button, LCD I2C, buzzer, LED, dan kipas DC, Sensor TCS3200 digunakan untuk membaca warna permukaan daging sapi sedangkan sensor MQ-135 digunakan untuk membaca gas amonia yang dihasilkan selama proses pembusukan daging, sensor DHT22 digunakan untuk mengukur suhu lingkungan sekitar.

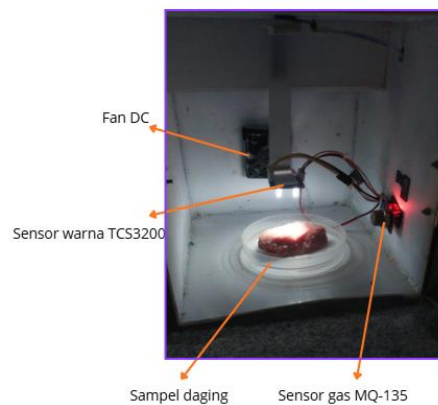
Data dari seluruh sensor diproses oleh ESP32 menggunakan model *Neural Network* yang telah dikonversi ke format *TensorFlow Lite*, hasil klasifikasi kemudian ditampilkan pada LCD berupa informasi tingkat kesegaran daging, selain itu buzzer akan aktif sebagai indikator peringatan ketika sistem mendeteksi daging dalam kondisi busuk. Push button pertama digunakan untuk memulai proses klasifikasi, sedangkan push button kedua digunakan untuk mematikan kipas DC guna membersihkan sisa gas di dalam ruang pengujian. LED digunakan sebagai indikator visual ketika kipas sedang aktif. Seluruh komponen memperoleh daya sesuai dengan kebutuhan tegangan masing-masing perangkat sehingga sistem dapat bekerja secara terintegrasi.



(a) Tampak depan



(b) Tampak samping



(c) Tampak dalam ruang uji

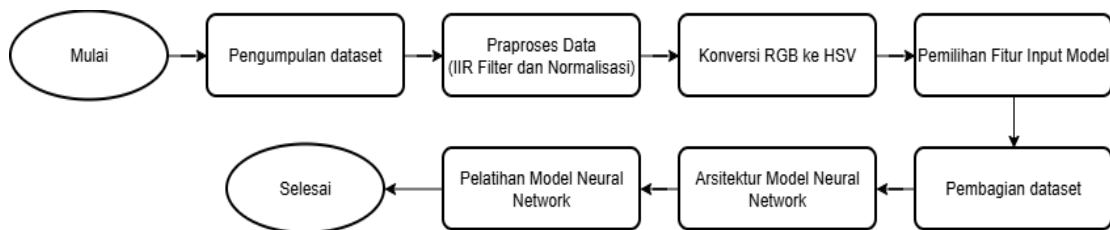
Gambar 6. Pembuatan *Hardware*

Gambar 6 merupakan tahap pembuatan *hardware* yang terdiri dari ruang uji coba dan *box* elektronika. Pada gambar (a) tampak samping dan gambar (b) tampak depan dari keseluruhan alat, di mana ruang uji coba dan *box* elektronika terlihat secara keseluruhan. Pada *box* elektronika terdapat komponen *output* berupa LED, buzzer, dan LCD I2C 20x4, serta komponen *input* berupa tombol klasifikasi dan tombol kipas DC. Pada gambar (c) merupakan tampak dalam ruang uji coba, di mana terdapat sensor warna TCS3200 yang dipasang di bagian atas ruang uji, sensor gas MQ-135 yang diletakkan di sisi kanan, kipas DC yang berfungsi sebagai ventilasi udara untuk membersihkan sisa gas setelah proses klasifikasi, serta alas untuk meletakkan sampel daging sapi yang akan diklasifikasikan.

2.2 Perancangan Software

Perancangan *software* dalam penelitian ini meliputi seluruh tahapan pengolahan data hingga menghasilkan model *Neural Network* yang siap diimplementasikan pada mikrokontroler

ESP32. Tahapan perancangan *software* disusun secara sistematis sebagaimana ditunjukkan pada Gambar 7.



Gambar 7. Diagram alir perancangan *software*

2.2.1 Pengolahan Neural Network

1. Pengumpulan *dataset*

Pada pengumpulan dataset ini dimulai dengan pengambilan data sensor secara langsung menggunakan perangkat yang telah dirancang, yaitu sensor gas MQ-135, sensor warna TCS3200, dan sensor suhu DHT22 yang terhubung ke mikrokontroler ESP32. Pengambilan data dilakukan secara *real-time* dengan meletakkan sampel daging sapi di dalam ruang pengujian yang telah disiapkan, Pengambilan dataset dilakukan dalam ruang tertutup untuk meminimalkan pengaruh angin dan kontaminasi gas dari luar terhadap pembacaan sensor MQ-135.

Tabel 2. Komposisi dataset daging

No	Kategori	Persentase	Rentang Waktu	Jumlah Sampel
1	Segar	37,46%	07.00 – 12.56	357
2	Semi	30,75%	12.57 – 17.49	293
3	Busuk	31,79%	17.50 – 22.52	303
Total				953

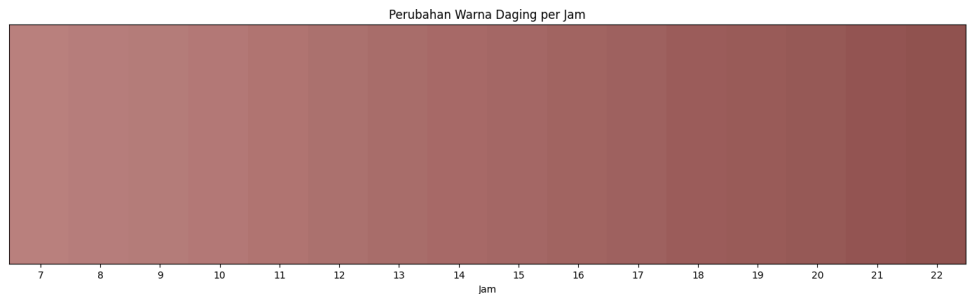
Tabel 2 merupakan *dataset* yang digunakan dalam penelitian ini berjumlah 953 sampel, yang terbagi ke dalam tiga kategori kesegaran daging sapi, yaitu Segar, Semi, dan Busuk. Setiap kategori mencerminkan kondisi daging pada rentang waktu pengambilan data yang berbeda dalam satu hari penuh. Kategori Segar terdiri dari 357 sampel yang diambil pada rentang waktu 07.00 hingga 12.56, kategori Semi terdiri dari 293 sampel pada rentang waktu 12.57 hingga 17.49, dan kategori Busuk terdiri dari 303 sampel pada rentang waktu 17.50 hingga 22.52. Dengan demikian, persentase masing-masing kategori adalah 37,46% untuk Segar, 30,75% untuk Semi, dan 31,79%

untuk Busuk, sehingga keseluruhan dataset yang digunakan dalam penelitian ini berjumlah 953 sampel.

Tabel 3. Komposisi sensor warna RGB

Kategori	R Min	G Min	B Min
Segar	167	107	106
Semi	150	91	89
Busuk	140	79	78

Tabel 3 merupakan komposisi nilai minimum sensor warna RGB untuk setiap kategori kesegaran daging sapi. Berdasarkan tabel tersebut, terlihat bahwa nilai minimum setiap kanal warna menurun seiring dengan penurunan kesegaran daging. Kategori Segar memiliki nilai minimum tertinggi dengan R = 167, G = 107, dan B = 106, yang mencerminkan warna permukaan daging yang masih merah segar. Kategori Semi menunjukkan penurunan nilai dengan R = 150, G = 91, dan B = 89, yang mengindikasikan perubahan warna permukaan daging akibat proses degradasi yang mulai berlangsung. Sementara itu, kategori Busuk memiliki nilai minimum terendah dengan R = 140, G = 79, dan B = 78, yang menggambarkan perubahan warna permukaan daging yang signifikan akibat proses pembusukan. Pola penurunan nilai RGB ini menunjukkan bahwa sensor warna TCS3200 mampu menangkap perubahan warna permukaan daging secara konsisten sebagai indikator tingkat kesegaran.



Gambar 8. Perubahan warna daging per jam

Gambar 8 menunjukkan visualisasi perubahan warna permukaan daging sapi secara rata-rata per jam selama proses pengambilan data berlangsung dari pukul 07.00 hingga 22.00. Berdasarkan gambar 6, terlihat bahwa warna daging

mengalami perubahan secara gradual dari warna merah muda yang relatif cerah pada jam-jam awal pengambilan data menuju warna merah kecokelatan yang lebih gelap dan redup seiring bertambahnya waktu. Perubahan warna ini mencerminkan proses degradasi daging yang berlangsung secara bertahap, di mana penurunan intensitas warna merah dan kecerahan permukaan daging merupakan indikator visual dari penurunan tingkat kesegaran. Pola perubahan warna yang konsisten dan searah ini mengonfirmasi bahwa sensor warna TCS3200 mampu menangkap perubahan karakteristik visual daging sapi secara representatif sebagai salah satu parameter dalam sistem klasifikasi kesegaran.

Tabel 4. Komposisi gas MQ-135 & suhu (DHT22)

Kategori	MQ Min	MQ Max	Suhu Min (°C)
Segar	0,70	1,89	29,5
Semi	0,94	2,67	32,8
Busuk	2,01	3,88	32,8

Tabel 4 merupakan komposisi nilai sensor gas MQ-135 dan sensor suhu DHT22 untuk setiap kategori kesegaran daging sapi. Berdasarkan tabel tersebut, terlihat bahwa nilai pembacaan sensor gas MQ-135 meningkat secara signifikan seiring dengan penurunan kesegaran daging. Kategori Segar memiliki nilai MQ terendah dengan rentang 0,70 hingga 1,89 volt dan suhu minimum 29,5°C, yang menunjukkan bahwa kadar gas amonia yang dihasilkan masih sangat rendah pada kondisi daging yang masih segar. Kategori Semi menunjukkan peningkatan nilai MQ dengan rentang 0,94 hingga 2,67 volt dan suhu minimum 32,8°C, yang mengindikasikan mulai meningkatnya kadar senyawa volatil akibat proses degradasi daging yang berlangsung. Sementara itu, kategori Busuk memiliki nilai MQ tertinggi dengan rentang 2,01 hingga 3,88 volt dan suhu minimum 32,8°C, yang menggambarkan tingginya konsentrasi gas amonia sebagai hasil dari proses pembusukan daging yang telah berlangsung secara penuh. Pola kenaikan nilai MQ-135 ini menunjukkan bahwa sensor gas mampu mendeteksi perubahan

kadar senyawa volatil secara konsisten sebagai indikator tingkat kesegaran daging sapi.

2. Praproses Data

Setelah proses pengumpulan data sensor selesai, seluruh data yang diperoleh terlebih dahulu melalui tahap praproses sebelum digunakan untuk pelatihan model. Praproses dilakukan untuk menyeragamkan dan membersihkan data agar model dapat mempelajari pola secara lebih optimal. Tahap praproses yang diterapkan meliputi dua proses utama, yaitu IIR filter dan konversi ruang warna HSV.

a. IIR Filter

IIR (*Infinite Impulse Response*) filter diterapkan pada seluruh data sensor, yaitu MQ, R, G, dan B, untuk meredam noise pada sinyal pembacaan sensor. Filter IIR bekerja dengan mempertimbangkan nilai sebelumnya secara rekursif, sehingga menghasilkan sinyal yang lebih halus tanpa menghilangkan pola utama data. Persamaan IIR filter satu arah yang digunakan ditunjukkan pada persamaan berikut :

$$y[n] = \alpha \cdot x[n] + (1 - \alpha) \cdot y[n-1] \quad (1)$$

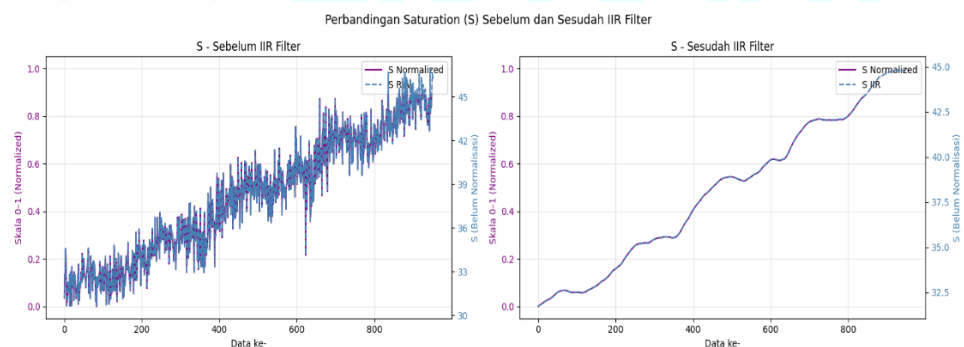
Keterangan:

$y[n]$ = nilai output filter pada data ke-n

$x[n]$ = nilai input (data sensor asli) pada data ke-n

$y[n-1]$ = nilai output filter pada data sebelumnya

α = koefisien filter = 0,05 (nilai smoothing)



Gambar 9. Perbedaan sebelum dan sesudah penapisan tapis IIR

Gambar 9 menunjukkan perbandingan nilai *Saturation* (S) sebelum dan sesudah penerapan filter IIR, dapat dilihat bahwa data Saturation sebelum proses filtering masih mengandung banyak fluktuasi atau noise yang

ditunjukkan oleh naik-turunnya nilai data secara acak. Meskipun demikian, tren kenaikan nilai *Saturation* tetap terlihat seiring bertambahnya waktu pengamatan. Fluktuasi tersebut kemungkinan disebabkan oleh gangguan pembacaan sensor warna, perubahan intensitas cahaya, maupun faktor lingkungan lainnya yang dapat memengaruhi hasil pengukuran, setelah diterapkan filter IIR, grafik menunjukkan perubahan yang signifikan pada kualitas data. Nilai *Saturation* menjadi lebih halus dan stabil karena noise yang sebelumnya muncul berhasil direduksi. Tren kenaikan data tetap dipertahankan sehingga informasi utama dari hasil pengukuran tidak hilang. Dengan berkurangnya fluktuasi yang tidak diinginkan, pola perubahan warna daging menjadi lebih mudah diamati dan dianalisis. Hasil ini menunjukkan bahwa filter IIR bekerja dengan baik dalam meningkatkan kualitas data sensor warna dengan cara menyaring gangguan berfrekuensi tinggi tanpa mengubah karakteristik utama sinyal. Data yang lebih stabil tersebut sangat bermanfaat sebagai masukan bagi jaringan saraf tiruan, karena dapat membantu model dalam mengenali pola kesegaran daging dengan lebih akurat dan mengurangi kemungkinan kesalahan klasifikasi yang disebabkan oleh noise pada data sensor. Hasil filtering disimpan sebagai fitur baru dengan label MQ_IIR, R_IIR, G_IIR, dan B_IIR.

b. Normalisasi Min – Max

Setelah proses IIR filter, seluruh fitur dinormalisasi ke dalam skala 0 hingga 1 menggunakan metode normalisasi Min-Max. Normalisasi dilakukan untuk menyeragamkan rentang nilai antar fitur sehingga tidak ada fitur yang mendominasi proses pembelajaran model akibat perbedaan skala. Persamaan normalisasi Min-Max ditunjukkan pada Persamaan berikut :

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (2)$$

Keterangan :

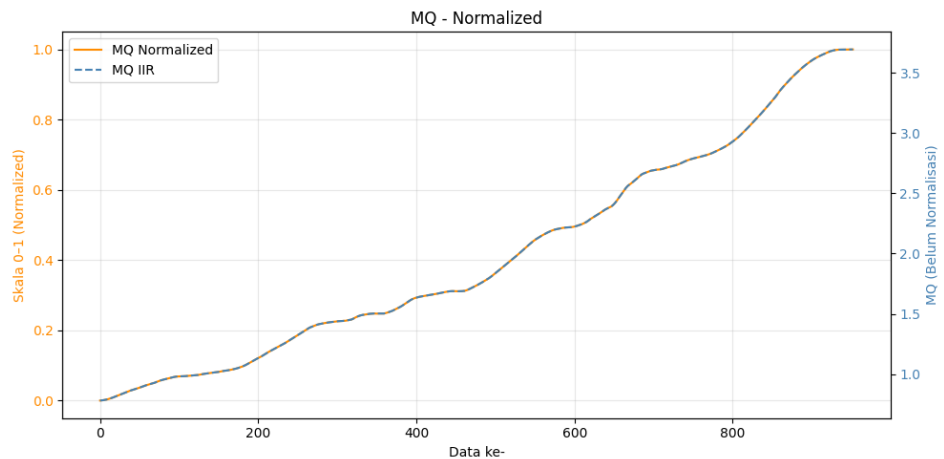
x' = nilai fitur setelah dinormalisasi (skala 0–1)

x = nilai fitur asli

$x_{\min}$ = nilai minimum fitur pada seluruh dataset

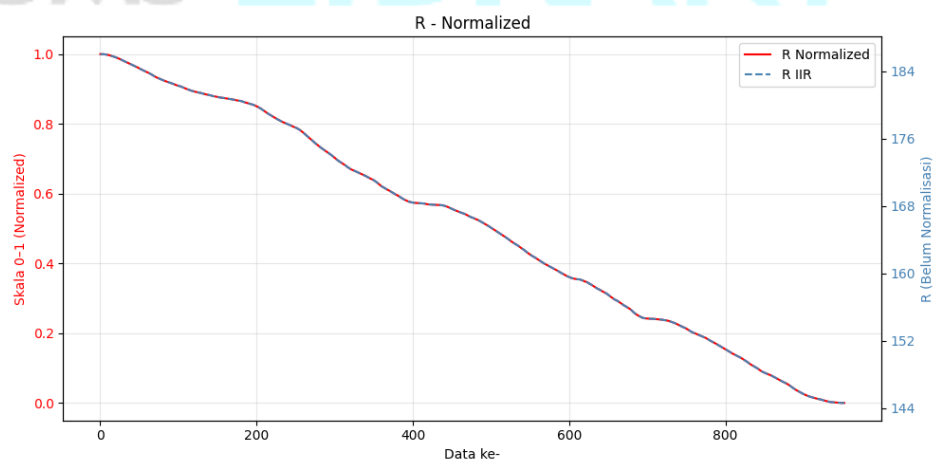
$x_{\max}$ = nilai maksimum fitur pada seluruh dataset

Hasil normalisasi masing-masing fitur sensor ditunjukkan pada gambar 10 hingga gambar 13.



Gambar 10. Normalisasi nilai MQ-135

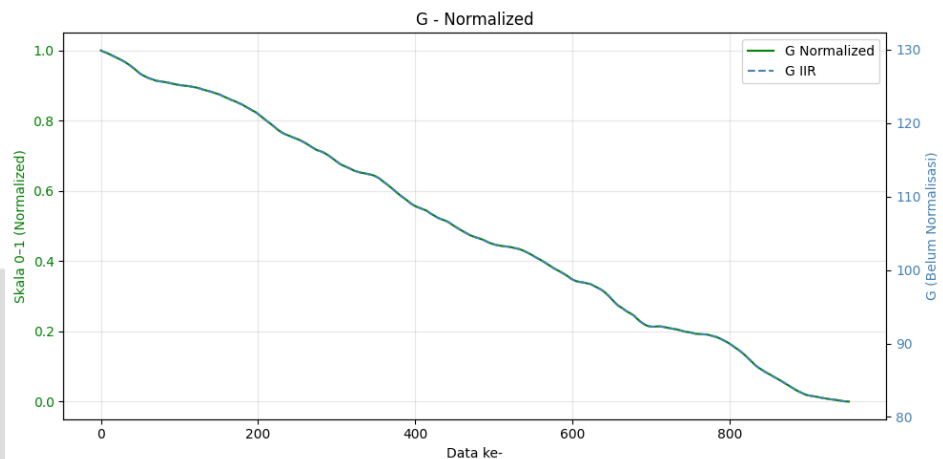
Gambar 10 menunjukkan perbandingan nilai MQ sebelum dan sesudah normalisasi setelah melalui proses IIR filter dalam satu grafik dual axis. Sumbu Y kiri menampilkan nilai MQ yang telah dinormalisasi pada skala 0–1, sedangkan sumbu Y kanan menampilkan nilai MQ asli hasil IIR filter yang berkisar antara 0,81 hingga 3,70. Terlihat bahwa kedua kurva memiliki bentuk yang identik dan bergerak searah, yang mengkonfirmasi bahwa proses normalisasi min-max tidak mengubah pola distribusi data, melainkan hanya menyesuaikan skalanya. Nilai MQ meningkat secara konsisten dari mendekati 0 pada data awal hingga mencapai 1,0 pada data akhir, mencerminkan peningkatan kadar gas amonia seiring berjalannya waktu dari kondisi Segar menuju Busuk.



Gambar 11. Normalisasi nilai r sensor warna TCS3200

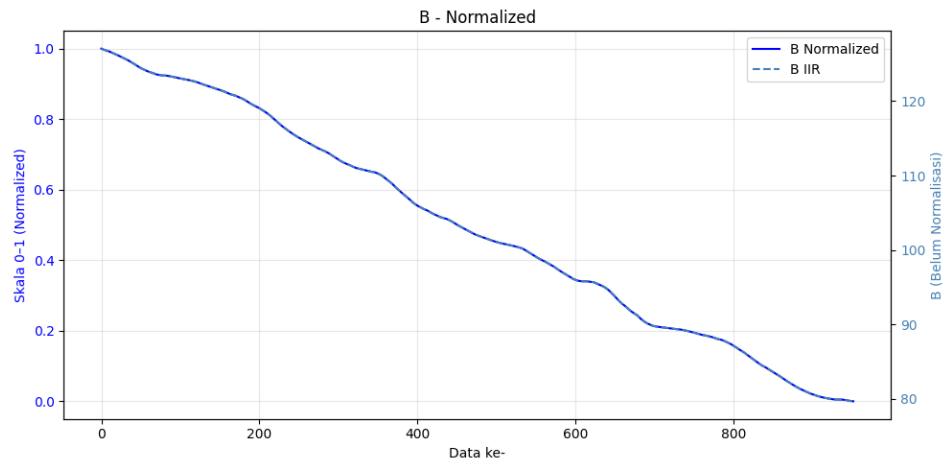
Gambar 11 menunjukkan perbandingan nilai R sebelum dan sesudah normalisasi setelah melalui proses IIR filter dalam satu grafik dual axis.

Sumbu Y kiri menampilkan nilai R yang telah dinormalisasi pada skala 0–1, sedangkan sumbu Y kanan menampilkan nilai R asli hasil IIR filter yang berkisar antara 144 hingga 184. Terlihat bahwa kedua kurva memiliki bentuk yang identik dan bergerak searah, yang mengkonfirmasi bahwa proses normalisasi min-max tidak mengubah pola distribusi data, melainkan hanya menyesuaikan skalanya. Berbeda dengan MQ, nilai R mengalami penurunan secara konsisten dari mendekati 1,0 pada data awal hingga mencapai 0 pada data akhir, mencerminkan penurunan intensitas warna merah pada sampel seiring berjalannya waktu dari kondisi Segar menuju Busuk.



Gambar 12. Normalisasi nilai g sensor warna TCS3200

Gambar 12 menunjukkan perbandingan nilai G sebelum dan sesudah normalisasi setelah melalui proses IIR filter dalam satu grafik dual axis. Sumbu Y kiri menampilkan nilai G yang telah dinormalisasi pada skala 0–1, sedangkan sumbu Y kanan menampilkan nilai G asli hasil IIR filter yang berkisar antara 80 hingga 130. Terlihat bahwa kedua kurva memiliki bentuk yang identik dan bergerak searah, yang mengkonfirmasi bahwa proses normalisasi min-max tidak mengubah pola distribusi data, melainkan hanya menyesuaikan skalanya. Serupa dengan nilai R, nilai G mengalami penurunan secara konsisten dari mendekati 1,0 pada data awal hingga mencapai 0 pada data akhir, mencerminkan penurunan intensitas warna hijau pada sampel seiring berjalannya waktu dari kondisi Segar menuju Busuk.



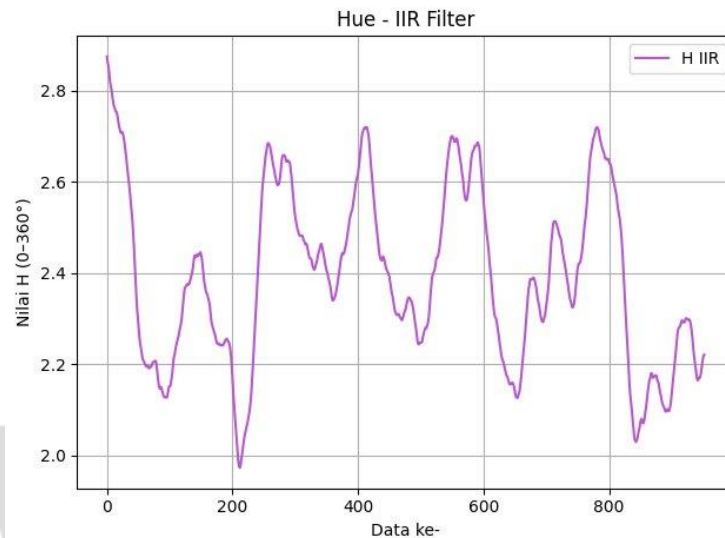
Gambar 13. Normalisasi nilai b sensor warna TCS3200

Gambar 13 menunjukkan perbandingan nilai B sebelum dan sesudah normalisasi setelah melalui proses IIR filter dalam satu grafik dual axis. Sumbu Y kiri menampilkan nilai B yang telah dinormalisasi pada skala 0–1, sedangkan sumbu Y kanan menampilkan nilai B asli hasil IIR filter yang berkisar antara 80 hingga 122. Terlihat bahwa kedua kurva memiliki bentuk yang identik dan bergerak searah, yang mengkonfirmasi bahwa proses normalisasi min-max tidak mengubah pola distribusi data, melainkan hanya menyesuaikan skalanya. Serupa dengan nilai R dan G, nilai B mengalami penurunan secara konsisten dari mendekati 1,0 pada data awal hingga mencapai 0 pada data akhir, mencerminkan penurunan intensitas warna biru pada sampel seiring berjalannya waktu dari kondisi Segar menuju Busuk. Pola penurunan ketiga komponen warna RGB yang searah ini mengindikasikan bahwa sampel mengalami perubahan warna menjadi lebih gelap secara keseluruhan seiring proses pembusukan.

3. Konversi RGB ke HSV

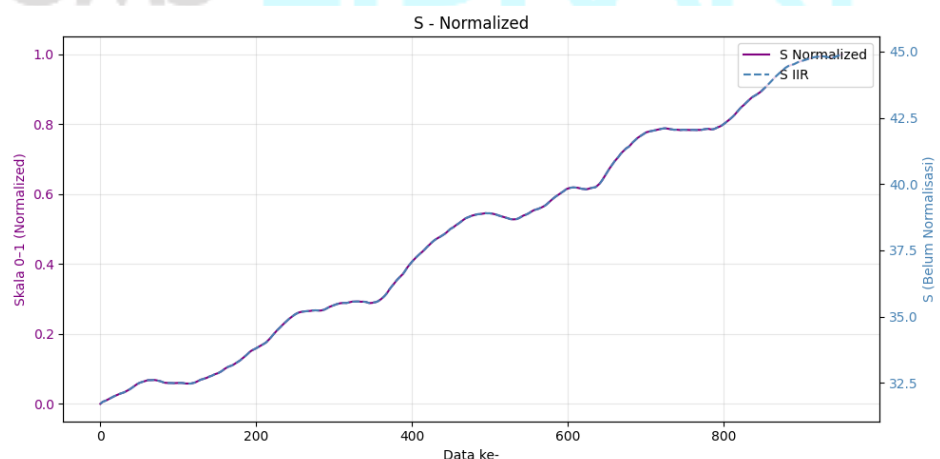
Setelah proses IIR filter, data warna hasil filtering kemudian dikonversi dari ruang warna RGB ke ruang warna HSV. Konversi ini bertujuan untuk menghasilkan representasi warna yang lebih stabil terhadap variasi intensitas cahaya dibandingkan RGB mentah, sehingga fitur warna yang dimasukkan ke model lebih konsisten antar sampel. Proses konversi RGB ke HSV dilakukan melalui tahapan, Pertama, nilai R, G, dan B dinormalisasi ke rentang $[0, 1]$ dengan membagi masing-masing nilai dengan 255. Konversi menghasilkan tiga komponen baru yaitu H_IIR , S_IIR , dan V_IIR . Hasil

konversi dan normalisasi komponen HSV ditunjukkan pada Gambar 14 hingga Gambar 16.



Gambar 14. Nilai hue (h) setelah IIR filter

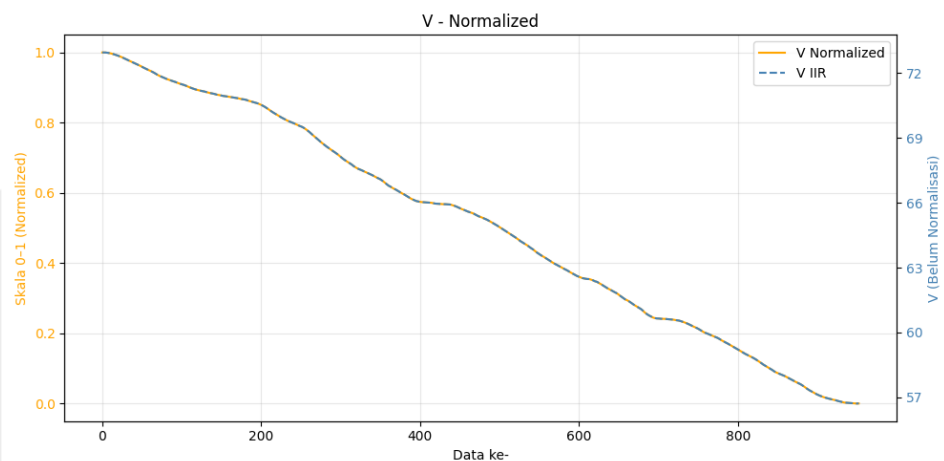
Gambar 14 menunjukkan nilai komponen Hue setelah melalui IIR filter. Terlihat bahwa nilai H berfluktuasi tidak beraturan di sepanjang data dengan rentang sekitar 2,0 hingga 2,9, sehingga komponen Hue tidak diikutsertakan sebagai fitur input model karena nilainya kurang stabil dan tidak menunjukkan tren yang konsisten terhadap perubahan kategori kesegaran daging.



Gambar 15. Normalisasi nilai saturation (S)

Gambar 15 menunjukkan perbandingan nilai S (Saturation) sebelum dan sesudah normalisasi setelah melalui proses IIR filter dalam satu grafik dual axis. Sumbu Y kiri menampilkan nilai S yang telah dinormalisasi pada skala 0–1, sedangkan sumbu Y kanan menampilkan nilai S asli hasil IIR filter yang

berkisar antara 32,5 hingga 45,0. Terlihat bahwa kedua kurva memiliki bentuk yang identik dan bergerak searah, yang mengkonfirmasi bahwa proses normalisasi min-max tidak mengubah pola distribusi data, melainkan hanya menyesuaikan skalanya. Berbeda dengan komponen RGB yang mengalami penurunan, nilai S justru menunjukkan tren peningkatan secara konsisten dari mendekati 0 pada data awal hingga mencapai 1,0 pada data akhir, mencerminkan peningkatan saturasi warna pada sampel seiring berjalannya waktu dari kondisi Segar menuju Busuk. Hal ini mengindikasikan bahwa meskipun sampel menjadi lebih gelap, warnanya justru semakin jenuh atau pekat selama proses pembusukan berlangsung.



Gambar 16. Normalisasi nilai value (V)

Gambar 16 menunjukkan perbandingan nilai V (Value) sebelum dan sesudah normalisasi setelah melalui proses IIR filter dalam satu grafik dual axis. Sumbu Y kiri menampilkan nilai V yang telah dinormalisasi pada skala 0–1, sedangkan sumbu Y kanan menampilkan nilai V asli hasil IIR filter yang berkisar antara 57 hingga 72. Terlihat bahwa kedua kurva memiliki bentuk yang identik dan bergerak searah, yang mengkonfirmasi bahwa proses normalisasi min-max tidak mengubah pola distribusi data, melainkan hanya menyesuaikan skalanya. Serupa dengan komponen RGB, nilai V mengalami penurunan secara konsisten dari mendekati 1,0 pada data awal hingga mencapai 0 pada data akhir, mencerminkan penurunan kecerahan pada sampel seiring berjalannya waktu dari kondisi Segar menuju Busuk. Penurunan nilai V yang searah dengan penurunan RGB ini semakin memperkuat kesimpulan bahwa sampel mengalami perubahan warna menjadi semakin gelap dan redup selama proses pembusukan, yang sejalan dengan perubahan visual yang dapat diamati secara langsung pada sampel.

4. Pemilihan Fitur Input Model

Berdasarkan hasil praproses yang telah dilakukan, fitur yang digunakan sebagai vektor masukan (*input vector*) model Neural Network berjumlah 6

fitur, yaitu: R_IIR, G_IIR, B_IIR, S_IIR, V_IIR, dan MQ_IIR. Pemilihan keenam fitur ini didasarkan pada analisis korelasi dan tren sinyal sensor terhadap perubahan tingkat kesegaran daging, sebagaimana telah ditunjukkan pada Gambar 10 hingga Gambar 16.

Komponen Hue (H) tidak diikutsertakan sebagai fitur meskipun dihitung dalam proses konversi HSV. Berdasarkan Gambar 14, nilai H menunjukkan fluktuasi yang tidak beraturan dan tidak berkorelasi dengan perubahan kesegaran daging, sehingga tidak memberikan informasi diskriminatif yang berarti bagi model. Sebaliknya, komponen S dan V menunjukkan tren yang konsisten dan berkorelasi langsung dengan tingkat pembusukan, sehingga keduanya dipertahankan sebagai fitur.

5. **Pembagian Dataset**

Dataset yang telah melalui tahap praproses kemudian dilakukan normalisasi menggunakan *StandardScaler* untuk menyeragamkan skala setiap fitur sebelum dimasukkan ke model. Normalisasi *StandardScaler* bekerja dengan mengurangi nilai rata-rata (mean) kemudian membagi dengan standar deviasi pada setiap fitur, sesuai Persamaan berikut :

$$Z = \frac{x - \mu}{\sigma} \quad (3)$$

Keterangan :

z = nilai fitur setelah dinormalisasi

x = nilai fitur asli

μ = nilai rata-rata (mean) fitur pada data latih

σ = standar deviasi fitur pada data latih

Setelah normalisasi, dataset dibagi ke dalam dua subset menggunakan metode train-test split dengan proporsi 80% untuk data latih (training set) dan 20% untuk data uji (test set). Pembagian dilakukan secara stratified berdasarkan label kategori untuk memastikan distribusi kelas yang proporsional pada kedua subset

Tabel 5. Pembagian dataset training dan testing

No	Subtet	Proporsi	Keterangan	Jumlah Sampel
1	Data Latih	80%	Melatih bobot model	762
2	Data Uji	20%	Evaluasi akhir model	191
Total				953

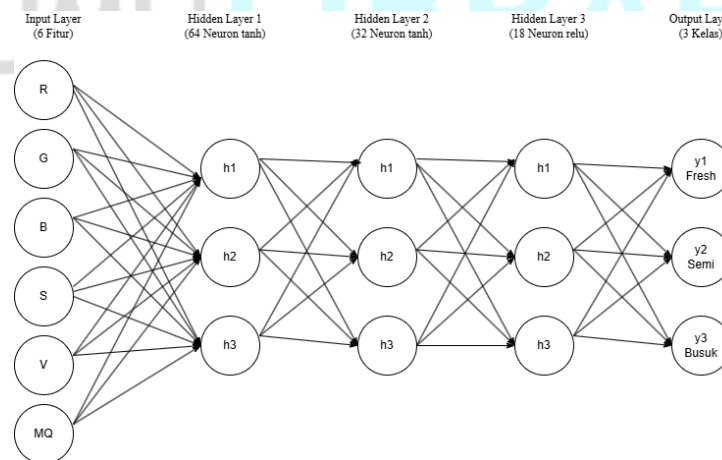
Dari total 953 sampel, diperoleh 762 sampel data latih dan 191 sampel data uji. Pembagian secara stratified memastikan setiap kelas (Segar, Semi, Busuk) terwakili secara proporsional pada kedua subset, sehingga hasil evaluasi dapat mencerminkan kemampuan generalisasi model secara objektif.

6. **Arsitektur Model *Neural Network***

Penentuan arsitektur model *Neural Network* dalam penelitian ini dilakukan melalui proses *hyperparameter* tuning menggunakan Keras Tuner dengan metode *RandomSearch* sebanyak 30 trial. Sebagaimana dikemukakan oleh (Bergstra & Bengio, 2012), metode *RandomSearch* terbukti secara empiris lebih efisien dibandingkan *gridsearch* dalam menemukan konfigurasi *hyperparameter* yang optimal dengan komputasi yang lebih singkat, sehingga metode ini dipilih sebagai strategi pencarian dalam penelitian ini. Pada setiap trial, Keras Tuner secara otomatis mencoba kombinasi jumlah neuron, fungsi aktivasi, dan learning rate yang berbeda, kemudian mengevaluasi performa masing-masing konfigurasi berdasarkan nilai `val_accuracy` pada data validasi. Dari 30 trial yang dijalankan, konfigurasi terbaik yang diperoleh adalah arsitektur dengan tiga hidden layer menggunakan susunan 64 neuron (tanh) → 32 neuron (tanh) → 18 neuron (relu) dengan learning rate 0,01, yang menghasilkan `val_accuracy` tertinggi sebesar 91,10%. Pola jumlah neuron yang semakin mengecil (64 → 32 → 18) mengikuti struktur *pyramidal*, di mana jumlah neuron dikurangi secara bertahap pada setiap layer berikutnya. Hal ini sejalan dengan temuan (Deng, 2023) yang menyatakan bahwa terdapat jumlah neuron optimal pada hidden layer, di mana penambahan neuron yang melampaui batas tersebut tidak meningkatkan akurasi secara signifikan bahkan berisiko menyebabkan *overfitting*. Dengan demikian, struktur

pyramidal yang diperoleh melalui proses tuning ini memungkinkan model untuk membangun representasi internal yang lebih ringkas dan bermakna.

Model *Neural Network* pada penelitian ini menggunakan metode *feedforward neural network*. Sebagaimana dijelaskan oleh (Sazli, 2006) *feedforward neural network* merupakan arsitektur jaringan saraf tiruan di mana informasi bergerak hanya dalam satu arah, dari input layer melewati hidden layer hingga output layer tanpa adanya siklus atau umpan balik (*feedback*) antar layer. Pada proses ini, setiap neuron menerima data dari layer sebelumnya, kemudian menghitung bobot dan fungsi aktivasi untuk menghasilkan output yang diteruskan ke layer berikutnya. Proses *feedforward* digunakan pada tahap inferensi untuk menentukan kelas kesegaran daging berdasarkan fitur input berupa R_IIR, G_IIR, B_IIR, S_IIR, V_IIR, dan MQ_IIR.



Gambar 17. Arsitektur model *neural network* library yang dipakai adalah TensorFlow

Tabel 6. Detail arsitektur model *neural network*

Layer	Tipe	Neuron	Aktivasi	Keterangan
Input	Input Layer	6	-	R_IIR, G_IIR, B_IIR, S_IIR, V_IIR, MQ_IIR
H1	Hidden	64	Tanh	Ekstraksi fitur utama, output terpusat di nol
H2	Hidden	32	Tanh	Kompresi fitur, representasi semakin abstrak

H3	Hidden	18	Relu	Kompersi akhir, mempercepat konvergensi
Output	Output	3	Softmax	Probabilitas kelas Segar/Semi/Busuk

Pemilihan fungsi aktivasi pada setiap layer dalam penelitian ini didasarkan pada karakteristik dan peran masing-masing fungsi aktivasi. Sebagaimana dikemukakan oleh (Feng & Lu, 2019) fungsi aktivasi tanh menghasilkan output yang terpusat di sekitar nol (zero-centered) dengan rentang nilai antara -1 hingga 1, sehingga lebih sesuai untuk data yang telah dinormalisasi dan membantu proses konvergensi pada tahap pelatihan. Oleh karena itu, tanh dipilih pada hidden layer pertama dan kedua yang menerima input fitur sensor yang telah melalui proses normalisasi. Pada hidden layer ketiga, digunakan fungsi aktivasi ReLU yang terbukti efektif dalam menghindari masalah *vanishing gradient* serta memiliki efisiensi komputasi yang lebih tinggi dibandingkan fungsi aktivasi berbasis eksponensial adapun pada output layer, digunakan fungsi aktivasi softmax yang mengubah output jaringan menjadi distribusi probabilitas untuk setiap kelas, sehingga sesuai untuk permasalahan klasifikasi multikelas. Fungsi aktivasi tanh didefinisikan pada persamaan berikut

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (4)$$

Keterangan :

x = nilai input neuron

$\tanh(x)$ = nilai output dalam rentang $(-1, 1)$

Fungsi aktivasi relu digunakan pada hidden layer ketiga untuk mempercepat konvergensi dan mengurangi risiko *vanishing gradient* pada layer yang lebih dalam. Fungsi relu didefinisikan pada persamaan berikut

$$\text{relu}(x) = \max(0, x) \quad (5)$$

Keterangan :

x = nilai input neuron

$\text{relu}(x)$ = nilai output: 0 jika $x < 0$, dan x jika $x \geq 0$

Pada layer output, fungsi aktivasi softmax digunakan untuk mengubah nilai logit menjadi distribusi probabilitas yang berjumlah 1 di antara ketiga kelas sesuai persamaan berikut

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \quad (6)$$

Keterangan :

z_i = nilai logit neuron output ke-i

$\sum_j e^{z_j}$ = jumlah eksponen seluruh neuron output (normalisasi)

$\text{softmax}(z_i)$ = probabilitas prediksi untuk kelas ke-i, bernilai 0–1

Kelas dengan nilai probabilitas softmax tertinggi dipilih sebagai hasil klasifikasi akhir yang ditampilkan pada LCD 20×4 dan memicu respons buzzer apabila kelas yang terdeteksi adalah Busuk.

7. Pelatihan Model *Neural Network*

Proses pelatihan model *Neural Network* dilakukan menggunakan platform *Google Colaboratory (Google Colab)* dengan dukungan GPU berbasis cloud. Model dikompilasi menggunakan optimizer Adam dengan learning rate 0,01.

```
# CALLBACK
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau
early_stop = EarlyStopping(
    monitor='val_loss',
    patience=25,
    restore_best_weights=True
)

reduce_lr = ReduceLROnPlateau(
    monitor='val_loss',
    factor=0.5,
    patience=10,
    min_lr=1e-5
)

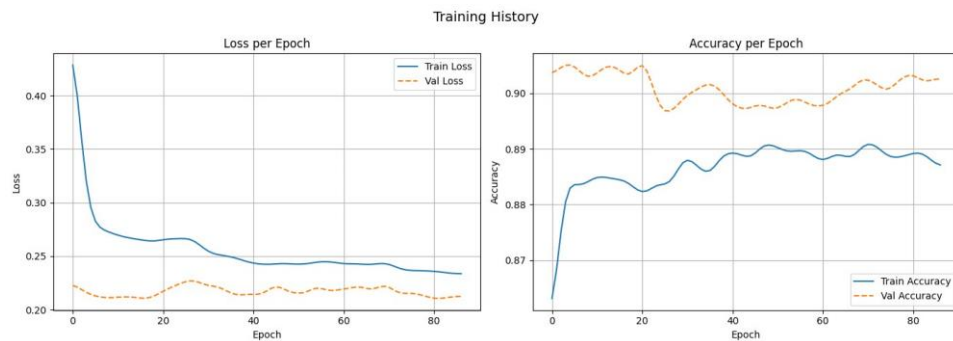
# Training Model

history = model.fit(
    x_train, y_train,
    epochs=200,
    batch_size=32,
    validation_split=0.2,
    class_weight=class_weight,
    callbacks=[early_stop, reduce_lr]
)
```

Gambar 18. Pelatihan model neural network

Pelatihan dilakukan dengan maksimum 200 epoch menggunakan dua callback utama. Pertama, EarlyStopping dengan patience 25 epoch yang menghentikan pelatihan secara otomatis apabila val_loss tidak membaik dan mengembalikan bobot terbaik (restore_best_weights = True). Kedua, ReduceLROnPlateau yang mengurangi learning rate sebesar setengahnya

apabila `val_loss` tidak membaik selama 10 epoch berturut-turut, dengan batas minimum learning rate 1×10^{-5} . Selain itu diterapkan class weight dengan bobot 2,0 pada kelas Semi untuk memberikan penekanan lebih pada kelas yang paling kritis secara praktis.



Gambar 19. Grafik Training History (Loss dan Accuracy per Epoch)

Berdasarkan Gambar 19, grafik *Loss per Epoch* menunjukkan bahwa nilai *Train Loss* mengalami penurunan signifikan dari sekitar 0,42 pada epoch awal menuju 0,23 pada akhir pelatihan, sedangkan *Val Loss* stabil di kisaran 0,21 hingga 0,23 sepanjang proses pelatihan. Kondisi ini menunjukkan bahwa model berhasil belajar dari data latih tanpa mengalami *overfitting* yang signifikan karena nilai *Val Loss* tidak meningkat meskipun *Train Loss* terus menurun.

Grafik *Accuracy per Epoch* menunjukkan bahwa *Val Accuracy* konsisten lebih tinggi dibandingkan *Train Accuracy* sepanjang pelatihan, dengan *Train Accuracy* berada di kisaran 0,88 dan *Val Accuracy* di kisaran 0,90 hingga 0,91. Hal ini mengindikasikan bahwa model memiliki kemampuan generalisasi yang baik terhadap data yang belum pernah dilihat sebelumnya. Proses pelatihan berhenti sebelum mencapai 200 *epoch* akibat mekanisme *EarlyStopping*, dengan bobot terbaik dikembalikan secara otomatis.

Setelah proses pelatihan selesai, model disimpan dalam format `.h5` kemudian dikonversi ke format *TensorFlow Lite* (`.tflite`) agar dapat dijalankan secara efisien pada mikrokontroler ESP32 dengan sumber daya komputasi terbatas. Model yang telah dikonversi selanjutnya diekspor ke dalam bentuk C array (`model_data.h`) untuk diunggah langsung ke ESP32, disertai dengan parameter normalisasi (mean dan scale) yang diperlukan untuk praproses data sensor secara *real-time* di dalam program ESP32.

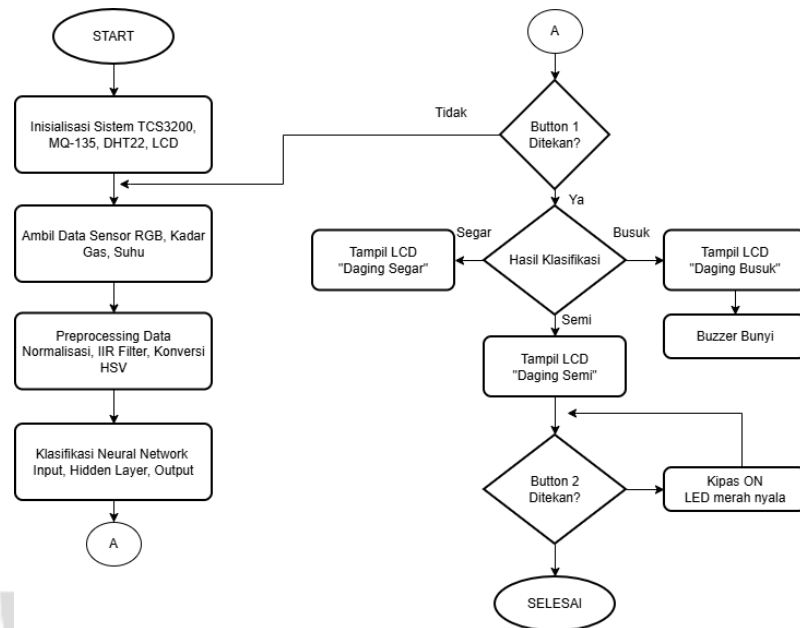
2.2.2 ESP32

ESP32 pada penelitian ini digunakan sebagai bagian dari sistem perangkat lunak tertanam (*embedded software*) yang berfungsi untuk memproses data sensor secara langsung dan mengendalikan keluaran sistem tanpa ketergantungan pada komputer eksternal. ESP32 dipilih karena memiliki prosesor *dual-core* Xtensa LX6 dengan kecepatan hingga 240 MHz serta kapasitas memori *flash* yang cukup untuk menyimpan model Neural Network dalam format TensorFlow Lite secara *on-device*.

Proses inferensi model dilakukan sepenuhnya di dalam ESP32 menggunakan library *TensorFlow Lite for Microcontrollers*. Data sensor dari MQ-135 dan TCS3200 dibaca secara *real-time*, kemudian melalui tahap praproses berupa IIR filter, konversi RGB ke HSV, dan normalisasi menggunakan parameter *StandardScaler* yang telah di-*hardcode* ke dalam program. Hasil praproses tersebut diumpankan ke model Neural Network untuk menghasilkan prediksi kelas kesegaran daging berupa *Segar*, *Semi*, atau *Busuk*.

Berdasarkan hasil inferensi yang diperoleh, perangkat lunak pada ESP32 mengendalikan berbagai keluaran sistem, yaitu tampilan LCD I2C 20×4 untuk menampilkan hasil klasifikasi secara teks, LED sebagai indikator visual, *buzzer* sebagai peringatan suara apabila daging terdeteksi dalam kondisi busuk, serta kipas DC untuk membersihkan ruang pengujian dari akumulasi gas volatil setelah proses klasifikasi selesai. Dengan pendekatan ini, ESP32 berperan sebagai unit pemrosesan terpadu yang mengintegrasikan akuisisi data sensor, inferensi model, dan pengendalian keluaran dalam satu perangkat, sehingga sistem dapat beroperasi secara mandiri dan *real-time* tanpa memerlukan koneksi ke komputer.

2.3 Perancangan Sistem



Gambar 20. *Flowchart* cara kerja alat

Flowchart gambar 20 menunjukkan alur kerja sistem pendeteksi kesegaran daging sapi yang dimulai dari proses inisialisasi seluruh komponen. Pada tahap ini, sensor TCS3200, sensor MQ-135, sensor DHT22, LCD 20×4, dan buzzer diaktifkan terlebih dahulu agar sistem siap digunakan. Setelah semua perangkat berhasil terhubung, alat mulai membaca data dari sensor berupa nilai warna RGB, kadar gas, dan suhu di sekitar daging.

Data yang didapat dari sensor kemudian diproses terlebih dahulu sebelum masuk ke tahap klasifikasi. Proses ini meliputi normalisasi data agar nilai sensor lebih stabil, penggunaan filter IIR untuk mengurangi noise pada pembacaan sensor, serta konversi warna dari RGB ke HSV supaya warna daging dapat dianalisis dengan lebih baik. Setelah proses preprocessing selesai, data akan masuk ke model *Neural Network* berbasis TensorFlow Lite untuk menentukan kondisi kesegaran daging.

Sistem kemudian menunggu input dari tombol pertama. Jika tombol belum ditekan, maka alat akan terus melakukan pembacaan sensor secara berulang. Namun jika tombol ditekan, sistem akan menampilkan hasil klasifikasi kesegaran daging pada LCD. Hasil klasifikasi dibagi menjadi tiga kondisi, yaitu segar, semi, dan busuk. Jika daging terdeteksi segar, LCD akan menampilkan tulisan “Daging Segar”. Jika kondisi daging semi segar, LCD akan menampilkan tulisan “Daging Semi”. Sedangkan jika daging

terdeteksi busuk, LCD akan menampilkan tulisan “Daging Busuk” dan buzzer akan berbunyi sebagai tanda peringatan.

Setelah hasil klasifikasi ditampilkan, sistem akan memeriksa tombol kedua. Jika tombol kedua ditekan, maka kipas akan menyala dan LED merah aktif sebagai indikator tambahan. Namun jika tombol kedua tidak ditekan, proses akan selesai. Setelah seluruh tahapan selesai, sistem dapat kembali melakukan pembacaan sensor sehingga alat mampu bekerja secara otomatis dan terus menerus dalam memantau kondisi kesegaran daging sapi.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Pengujian *Hardware*

Pengujian *hardware* dilakukan untuk memverifikasi bahwa seluruh komponen sistem bekerja sesuai dengan perancangan. Pengujian dilakukan dengan menyalakan sistem dan memeriksa satu per satu fungsi dari setiap komponen. Hasil pengujian hardware secara keseluruhan ditunjukkan pada tabel 7.

Tabel 7. Hasil pengujian komponen hardware

No	Komponen	Status	Keterangan
1	Sensor MQ-135	Normal	Membaca kadar gas amonia dari sampel daging
2	Sensor Warna TCS3200	Normal	Membaca nilai RGB permukaan daging secara <i>real-time</i>
3	Sensor Suhu DHT22	Normal	Membaca suhu dan kelembapan lingkungan pengujian
4	LCD I2C 20×4	Normal	Menampilkan hasil klasifikasi dan nilai sensor
5	Buzzer	Normal	Aktif dan berbunyi saat kategori Busuk terdeteksi
6	Kipas DC	Normal	Aktif membersihkan ruang saat tombol kedua ditekan
7	LED Indikator	Normal	Menyala bersamaan dengan aktivasi kipas DC
8	Push Button 1	Normal	Memicu proses akuisisi dan inferensi model
9	Push Button 2	Normal	Mengaktifkan kipas DC secara manual

Tabel 7, seluruh komponen hardware yang digunakan dalam sistem berfungsi dengan baik sesuai perancangan. Sensor MQ-135 berhasil mendeteksi kadar gas amonia yang meningkat seiring penurunan kesegaran daging. Sensor warna TCS3200 mampu membaca nilai RGB permukaan daging secara *real-time*. LCD I2C 20×4 menampilkan hasil klasifikasi dengan jelas, buzzer aktif saat kondisi Busuk terdeteksi, dan kipas DC berfungsi membersihkan ruang pengujian setelah setiap sesi klasifikasi.

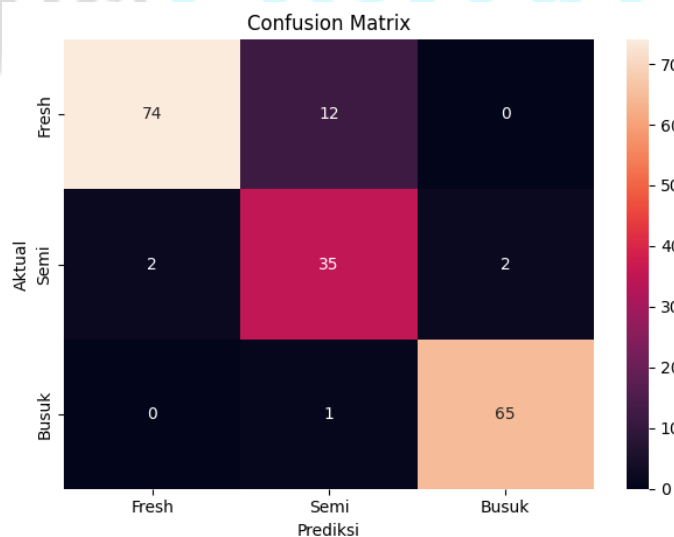
3.2 Hasil Pelatihan Model *Neural Network*

Pelatihan model *Neural Network* dilakukan menggunakan *Google Colaboratory* dengan dukungan GPU. Model dilatih dengan maksimum 200 epoch menggunakan mekanisme *EarlyStopping* (*patience* = 25) dan *ReduceLROnPlateau* (*patience* = 10, *factor* = 0,5). Proses pelatihan berhenti secara otomatis sebelum mencapai epoch maksimum karena *val_loss* tidak mengalami perbaikan selama 25 epoch berturut-turut. Berdasarkan Gambar 13 yang telah ditampilkan pada bagian Metode, nilai Train Loss mengalami penurunan signifikan dari sekitar 0,42 pada epoch awal menuju 0,23 pada akhir pelatihan, sementara Val Loss stabil di kisaran 0,21 hingga 0,23. Kondisi ini menunjukkan bahwa model berhasil belajar dari data latih tanpa mengalami *overfitting* yang signifikan. Val Accuracy konsisten berada di kisaran 0,90 hingga 0,91, yang mengonfirmasi bahwa model memiliki kemampuan generalisasi yang baik pada data yang belum pernah dilihat sebelumnya.

3.3 Evaluasi Model

3.3.1 *Confusion Matrix*

Evaluasi model dilakukan menggunakan 191 data uji (20% dari total *dataset*) yang tidak digunakan selama proses pelatihan. Hasil *confusion matrix* ditunjukkan pada Gambar 21 dan dirangkum dalam Tabel 8.

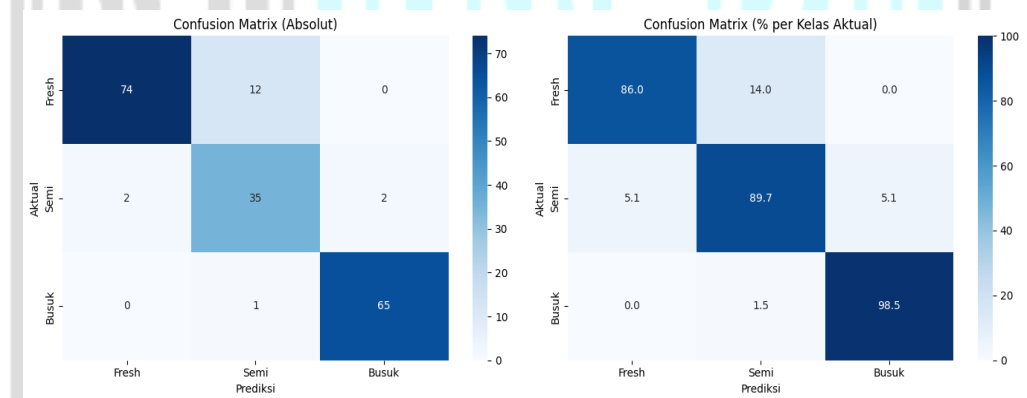


Gambar 21. *Confusion matrix* model *neural network*

Tabel 8. *Confusion matrix* model *neural network* (n = 191 sampel uji)

Aktual/Prediksi	Busuk	Segar	Semi
Segar (n=86)	0	74	12
Semi (n=39)	2	2	35
Busuk (n=66)	65	0	1

Gambar 21 dan Tabel 8, dari 66 sampel kelas Busuk, sebanyak 65 sampel diklasifikasikan dengan benar dan hanya 1 sampel salah diklasifikasikan sebagai Semi. Dari 86 sampel kelas Segar, sebanyak 74 sampel diklasifikasikan dengan benar, sementara 12 sampel salah diklasifikasikan sebagai Semi. Dari 39 sampel kelas Semi, sebanyak 35 sampel diklasifikasikan dengan benar, 2 sampel salah diklasifikasikan sebagai Busuk, dan 2 sampel salah diklasifikasikan sebagai Segar. Kelas Segar menunjukkan tingkat kesalahan terbesar dengan 12 sampel salah diklasifikasikan sebagai Semi. Hal ini disebabkan oleh karakteristik nilai sensor pada kondisi segar yang mendekati akhir masa kesegaran memiliki kemiripan dengan kondisi semi-segar, sehingga batas keputusan model sulit ditentukan secara tegas. Kelas Busuk memiliki tingkat klasifikasi terbaik karena perbedaan nilai MQ-135 antara kondisi busuk dengan kondisi lainnya sangat signifikan.



Gambar 22. Confusion matrix absolut dan ternormalisasi (% per kelas aktual)

Gambar 22 menampilkan confusion matrix dalam dua format yaitu nilai absolut dan persentase per kelas aktual. Berdasarkan confusion matrix ternormalisasi, kelas Busuk memiliki tingkat klasifikasi benar tertinggi sebesar 98,5%, diikuti kelas Semi sebesar 89,7%, dan kelas Segar sebesar 86,0%. Tingkat recall kelas Segar sebesar 86,0% menjadi perhatian utama karena 14,0% sampel Segar salah diklasifikasikan sebagai Semi.

3.3.2 Precision, Recall, dan F1-Score

Hasil evaluasi lengkap model *Neural Network* menggunakan metrik *precision*, *recall*, dan *F1-score*, *Precision* mengukur seberapa tepat model dalam memprediksi suatu kelas, yaitu proporsi prediksi benar dari seluruh data yang diprediksi sebagai kelas tertentu. *Recall* mengukur seberapa lengkap model dalam mendeteksi seluruh sampel yang sebenarnya termasuk dalam suatu kelas. Adapun *F1-score* merupakan rata-rata harmonik dari *precision* dan *recall* yang digunakan untuk memberikan penilaian tunggal terhadap keseimbangan antara keduanya,

sehingga lebih representatif dalam mengevaluasi performa model pada kondisi distribusi kelas yang tidak seimbang. untuk setiap kelas ditunjukkan pada Tabel 9.

Tabel 9. Classification report model neural network

Kelas	Precision	Recall	F1-Score	Support
Busuk	0,97	0,98	0,98	66
Segar	0,97	0,86	0,91	86
Semi	0,73	0,90	0,80	39
<i>Accuracy</i>			0,91	191

Tabel 9 menunjukkan model memperoleh akurasi keseluruhan sebesar 91,10% pada data uji. Kelas Busuk memperoleh nilai *precision* dan *recall* tertinggi masing-masing sebesar 0,97 dan 0,98 dengan *F1-score* 0,98, mengindikasikan bahwa model sangat handal dalam mendeteksi daging yang tidak layak konsumsi. Hal ini sangat penting secara praktis karena kesalahan dalam mengklasifikasikan daging busuk sebagai segar dapat membahayakan kesehatan konsumen. Kelas Semi memiliki nilai *precision* terendah sebesar 0,73, artinya dari seluruh sampel yang diprediksi sebagai Semi, sekitar 27% sebenarnya bukan Semi. Nilai *recall* Semi sebesar 0,90 cukup baik, menunjukkan bahwa model mampu mendeteksi sebagian besar sampel Semi yang ada. *Weighted average F1-score* sebesar 0,91 menunjukkan bahwa model secara keseluruhan memiliki performa klasifikasi yang baik dengan mempertimbangkan distribusi jumlah sampel per kelas.

3.4 Hasil Pengujian Sistem Pada Sampel Daging Nyata

Pengujian sistem dilakukan secara langsung menggunakan sampel daging sapi asli pada tiga kondisi kesegaran yang berbeda. Setiap sampel diuji beberapa kali untuk mengevaluasi konsistensi dan keandalan sistem dalam kondisi nyata. Hasil pengujian ditampilkan pada Tabel 10,11,12.

Tabel 10. Hasil pengujian sistem pada daging segar

No	Kondisi Aktual	MQ (V)	R	G	B	Prediksi Sistem	Hasil
1	Segar	0,83	163	120	113	Segar	Benar
2	Segar	0,81	164	134	124	Segar	Benar
3	Segar	0,92	172	150	142	Segar	Benar
4	Segar	0,84	185	167	156	Segar	Benar
5	Segar	1,04	156	137	130	Semi	Salah
6	Segar	1,20	166	152	146	Segar	Benar
7	Segar	0,97	167	157	150	Segar	Benar
8	Segar	0,90	160	132	126	Semi	Salah
9	Segar	0,84	170	148	140	Segar	Benar
10	Segar	0,80	179	143	133	Segar	Benar

Tabel 11. Hasil pengujian pada daging semi

No	Kondisi Aktual	MQ (V)	R	G	B	Prediksi Sistem	Hasil
1	Semi	1.07	149	126	115	Semi	Benar
2	Semi	1,37	147	125	117	Semi	Benar
3	Semi	1,31	154	131	124	Semi	Benar
4	Semi	1,54	153	130	120	Semi	Benar
5	Semi	1,38	160	140	129	Segar	Salah
6	Semi	1,37	150	143	136	Semi	Benar
7	Semi	1,70	140	107	98	Semi	Benar
8	Semi	1,87	132	105	98	Busuk	Salah
9	Semi	1,30	150	129	120	Semi	Benar
10	Semi	1,25	162	118	112	Segar	Salah

Tabel 12. Hasil pengujian pada daging busuk

No	Kondisi Aktual	MQ (V)	R	G	B	Prediksi Sistem	Hasil
1	Busuk	5,28	88	46	40	Busuk	Benar
2	Busuk	3,09	134	92	84	Busuk	Benar
3	Busuk	2,55	115	85	78	Busuk	Benar
4	Busuk	2,67	96	54	46	Busuk	Benar
5	Busuk	3,07	114	84	78	Busuk	Benar
6	Busuk	2,72	127	92	83	Busuk	Benar
7	Busuk	2,65	111	83	75	Busuk	Benar
8	Busuk	2,87	130	117	110	Busuk	Benar
9	Busuk	2,68	126	101	93	Busuk	Benar
10	Busuk	2,88	121	105	98	Busuk	Benar

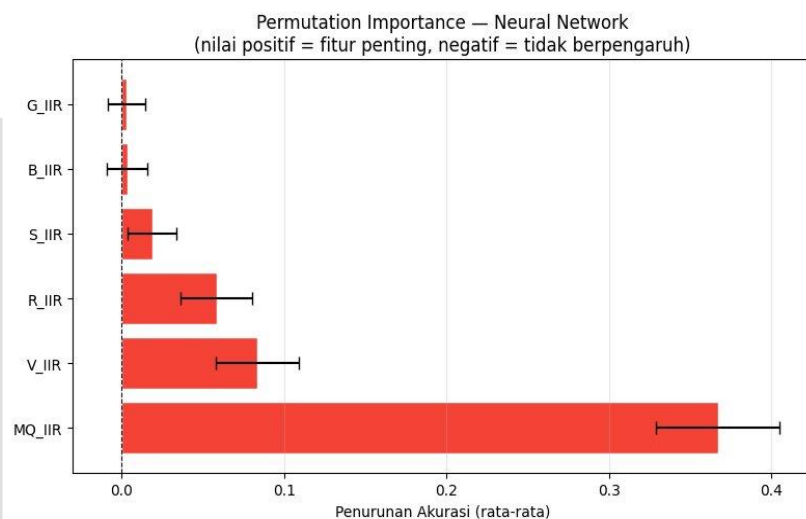
Tabel 10, 11, dan 12, pengujian sistem dilakukan secara langsung menggunakan 30 sampel daging sapi pada tiga kondisi kesegaran yang berbeda. Pada kondisi *segar*, sistem berhasil mengklasifikasikan 8 dari 10 sampel secara benar (80%), dengan 2 sampel salah diprediksi sebagai Semi akibat nilai MQ yang relatif lebih tinggi dan nilai RGB yang lebih rendah dibandingkan sampel *segar* lainnya, sehingga profil sensornya mendekati batas kelas Semi. Pada kondisi semi, sistem mengklasifikasikan 7 dari 10 sampel secara benar (70%), di mana 2 sampel diprediksi sebagai *segar* karena nilai MQ yang masih rendah dan RGB yang masih cukup tinggi, serta 1 sampel diprediksi sebagai busuk karena nilai MQ-nya sudah cukup tinggi. Pada kondisi busuk, sistem berhasil mengklasifikasikan seluruh 10 sampel dengan benar (100%), karena nilai MQ-135 pada kondisi busuk jauh lebih tinggi dibandingkan dua kelas lainnya sehingga model dapat membedakannya dengan sangat baik. Secara keseluruhan, sistem memperoleh akurasi pengujian nyata sebesar 83,3% (25

dari 30 sampel benar), dengan kesalahan klasifikasi yang sebagian besar terjadi pada zona transisi antara kelas *Segar* dan *Semi* yang memang tidak memiliki batas nilai sensor yang tegas.

3.5 Analisis Kepentingan Fitur

3.5.1 *Permutation Importance*

Analisis *permutation importance* dilakukan untuk mengukur kontribusi setiap fitur terhadap performa model *Neural Network*. Metode ini bekerja dengan mengacak nilai satu fitur secara acak sebanyak 30 kali, kemudian mengukur seberapa besar penurunan akurasi akibat pengacakan tersebut. Hasil analisis ditunjukkan pada Gambar 23.

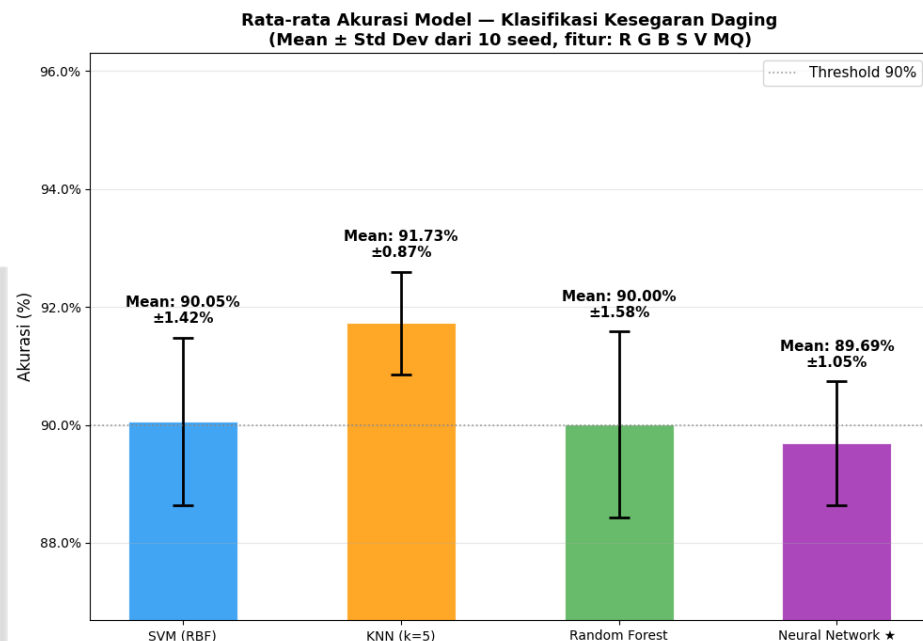


Gambar 23. *Permutation importance neural network*

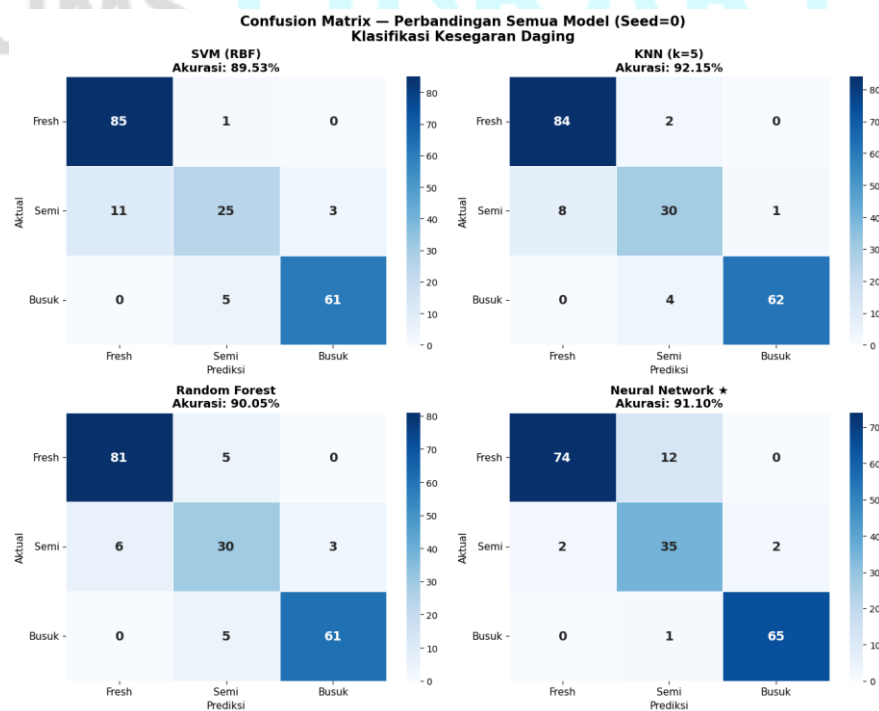
Gambar 23, fitur MQ_IIR merupakan fitur dengan kontribusi terbesar terhadap performa model dengan rata-rata penurunan akurasi mencapai 0,35 ketika fitur tersebut diacak. Hal ini mengonfirmasi bahwa sensor gas MQ-135 adalah komponen paling kritis dalam sistem, karena kadar gas amonia memiliki tren perubahan yang paling konsisten dan signifikan seiring penurunan kesegaran daging. Fitur V_IIR menempati posisi kedua, diikuti oleh R_IIR dan S_IIR. Fitur G_IIR dan B_IIR menunjukkan nilai mendekati nol bahkan sedikit negatif, mengindikasikan bahwa kedua fitur tersebut memberikan kontribusi yang sangat kecil secara individual terhadap model.

3.6 Perbandingan Dengan Model Lain

Untuk mengevaluasi performa model *Neural Network* secara objektif, dilakukan perbandingan dengan tiga model baseline yaitu *SVM* dengan kernel *RBF*, *KNN* ($k=5$), dan *Random Forest*. Semua model dilatih menggunakan dataset dan split yang identik (80:20) menggunakan 10 seed acak. Hasil perbandingan akurasi ditunjukkan pada Gambar 24, sedangkan confusion matrix masing-masing model ditunjukkan pada Gambar 25.



Gambar 24. Perbandingan akurasi semua model klasifikasi



Gambar 25. Confusion matrix perbandingan semua model

Gambar 24, dan Gambar 25, KNN ($k=5$) memperoleh rata-rata akurasi tertinggi sebesar 91,73% ($\pm 0,87\%$), diikuti SVM (RBF) sebesar 90,05% ($\pm 1,42\%$) dan *Random Forest* sebesar 90,00% ($\pm 1,58\%$), namun perlu dicatat bahwa ketiga model tersebut tidak dapat diimplementasikan secara langsung pada mikrokontroler ESP32 karena memerlukan memori dan komputasi yang jauh melebihi kapasitasnya. KNN bersifat *lazy learner* yang mengharuskan seluruh data latih disimpan saat inferensi, sehingga mustahil dijalankan pada mikrokontroler dengan memori terbatas. Model *Neural Network* memperoleh rata-rata akurasi sebesar 89,69% ($\pm 1,05\%$) dalam format *TensorFlow Lite* yang telah dioptimalkan untuk berjalan langsung pada ESP32 tanpa server eksternal. Meskipun rata-rata akurasi *Neural Network* tidak menjadi yang tertinggi dalam perbandingan ini, model ini memiliki standar deviasi yang relatif kecil ($\pm 1,05\%$) yang menunjukkan stabilitas performa antar *seed* yang cukup baik. Selain itu, berdasarkan *confusion matrix* pada Gambar 21, model *Neural Network* menunjukkan performa terbaik pada kelas Busuk dengan hanya 1 sampel salah klasifikasi, jauh lebih baik dibandingkan SVM (5 salah), KNN (4 salah), dan *Random Forest* (5 salah). Kemampuan mendeteksi kelas Busuk secara akurat sangat krusial dari sisi keamanan pangan, karena kesalahan mengklasifikasikan daging busuk sebagai segar berpotensi membahayakan kesehatan konsumen. Keunggulan utama *Neural Network* terletak pada kemampuan *on-device inference* pada mikrokontroler ESP32. Pendekatan ini memungkinkan sistem beroperasi secara mandiri, *real-time*, dan portabel tanpa ketergantungan pada server atau koneksi internet, yang menjadikannya pilihan paling tepat untuk tujuan penelitian ini.

3.7 Analisis Keterbatasan Sistem

Meskipun sistem menunjukkan performa klasifikasi yang baik, terdapat beberapa keterbatasan yang perlu diperhatikan untuk pengembangan lebih lanjut:

1. Pengaruh kondisi lingkungan. Suhu dan kelembapan ruangan yang tinggi dapat mempengaruhi baseline pembacaan sensor MQ-135, sehingga sistem perlu dikalibrasi ulang apabila digunakan pada lingkungan dengan kondisi yang berbeda signifikan dari kondisi pengujian.
2. Keterbatasan variasi dataset. Dataset hanya mencakup daging sapi lokal dalam satu sesi pengambilan data, sehingga diperlukan pengujian lebih lanjut dengan variasi jenis daging dan kondisi penyimpanan yang berbeda untuk meningkatkan robustness model.
3. Pengaruh variasi cahaya. Meskipun telah menggunakan konversi ruang warna HSV, intensitas cahaya eksternal yang berubah-ubah secara ekstrem masih berpotensi mempengaruhi akurasi pembacaan sensor warna TCS3200.
4. Zona ambiguitas kelas Segar dan Semi. Batas antara kategori Segar dan Semi bersifat gradual dan tidak memiliki batas yang tegas secara fisik, sebagaimana ditunjukkan oleh 12 sampel Segar yang salah diklasifikasikan sebagai Semi pada *confusion matrix*.

4.PENUTUP

4.1 Kesimpulan

Penelitian ini berhasil merancang dan membangun sistem klasifikasi kesegaran daging sapi menggunakan metode *Neural Network* yang diimplementasikan langsung pada mikrokontroler ESP32 dengan memanfaatkan sensor gas MQ-135 dan sensor warna TCS3200. Model yang dikembangkan menggunakan arsitektur tiga *hidden layer* dengan susunan 64–32–18 neuron dalam format *TensorFlow Lite*, dan dilatih menggunakan 953 sampel data sensor yang dibagi dengan proporsi 80:20. Sistem yang dibangun berhasil beroperasi secara mandiri dan *real-time* tanpa memerlukan server eksternal maupun koneksi internet. Evaluasi model pada data uji menghasilkan akurasi sebesar 91,10% dengan *weighted average F1-score* sebesar 0,91, di mana kelas Busuk memperoleh performa terbaik dengan *F1-score* dan *recall* masing-masing sebesar 0,98, yang mengonfirmasi keandalan sistem dalam mendeteksi daging yang tidak layak konsumsi. Pengujian langsung pada 30 sampel daging sapi nyata menghasilkan akurasi sebesar 83,3%, dengan seluruh 10 sampel daging busuk berhasil diklasifikasikan secara benar (100%). Kesalahan yang terjadi sebagian besar berada pada zona transisi antara kategori *Segar* dan *Semi* yang secara fisik tidak memiliki batas nilai sensor yang tegas. Dibandingkan dengan penelitian terdahulu yang menggunakan metode serupa, sistem ini mencapai akurasi tertinggi sebesar 91,10% dan merupakan satu-satunya implementasi yang mampu menjalankan inferensi secara langsung pada mikrokontroler ESP32 tanpa server eksternal, menjadikannya lebih praktis, portabel, dan hemat biaya untuk digunakan secara mandiri.

4.2 Saran

Berdasarkan keterbatasan yang telah diidentifikasi, pengembangan lebih lanjut disarankan mencakup perluasan dataset dengan variasi jenis daging sapi, kondisi penyimpanan yang berbeda seperti daging beku atau daging dalam kemasan vakum, serta sesi pengambilan data pada hari yang berbeda guna meningkatkan kemampuan generalisasi model. Selain itu, penambahan mekanisme kalibrasi otomatis pada sensor MQ-135 perlu dipertimbangkan untuk mengakomodasi perubahan kondisi lingkungan seperti suhu dan kelembapan, sehingga akurasi pembacaan sensor tetap terjaga pada berbagai kondisi penggunaan. Perlu diakui pula bahwa label kesegaran pada penelitian ini ditentukan menggunakan ambang batas pada nilai R dan MQ yang dikalibrasi dari pengamatan fisik daging, sehingga penelitian selanjutnya sebaiknya memvalidasi label

dengan metode yang independen dari sensor yang dipakai, misalnya uji laboratorium seperti TVB-N atau pH, agar hasil klasifikasi lebih dapat dipertanggungjawabkan secara ilmiah. Ke depannya, sistem juga dapat dikembangkan dengan antarmuka yang lebih mudah diakses melalui integrasi aplikasi *smartphone* berbasis *Bluetooth* atau Wi-Fi, agar sistem dapat dimanfaatkan secara lebih luas oleh masyarakat umum, termasuk penderita gangguan penciuman.

PERSANTUNAN

Alhamdulillah rabbil'alam, segala puji dan syukur penulis panjatkan kehadirat Allah SWT atas limpahan rahmat, hidayah, dan kemudahan-Nya sehingga penulisan artikel ilmiah ini dapat terselesaikan dengan baik. Penulis juga menyampaikan terima kasih dan penghargaan yang setinggi-tingginya kepada berbagai pihak yang telah memberikan dukungan baik teknis maupun non teknis selama proses penyusunan dan penyelesaian penelitian ini. Ucapan terima kasih secara khusus penulis sampaikan kepada :

1. Orang tua dan keluarga tercinta yang senantiasa memberikan doa, dukungan, dan semangat dalam menyelesaikan studi dan penelitian ini.
2. Bapak Fajar Suryawan, S.T.,M.Eng.Sc.,Ph.D selaku dosen pembimbing yang telah memberikan arahan, motivasi, dan bimbingan secara intensif selama proses penelitian dan penyusunan artikel ilmiah.
3. Seluruh dosen dan staf pengajar di Program Studi Teknik Elektro Universitas Muhammadiyah Surakarta yang telah memberikan ilmu dan pengalaman berharga selama masa perkuliahan.
4. Teman-teman Robot Research UMS yang telah memberikan fasilitas, pengetahuan, serta dukungan yang sangat berarti bagi penulis dalam menyelesaikan penelitian ini.
5. Seluruh rekan mahasiswa Teknik Elektro angkatan 2022 yang telah menemani dan memberikan bantuan selama proses perkuliahan hingga tahap akhir penyusunan karya ilmiah ini.
6. Perempuan (F.F) yang dengan penuh kasih sayang senantiasa memberikan doa, semangat dan dukungan moral yang tiada henti kepada penulis, sehingga penulis mampu menyelesaikan penelitian dan penyusunan artikel ilmiah ini dengan baik hingga akhir.

Semoga segala bantuan dan kebaikan yang telah diberikan menjadi amal jariyah dan mendapat balasan yang terbaik dari Allah SWT.

DAFTAR PUSTAKA

- Asmoro, P. B., & Solichin, A. (2023). *Penerapan Metode Convolutional Neural Network untuk Klasifikasi Kualitas Daging Sapi pada Aplikasi Berbasis Android*. 16(4), 286–298. <https://doi.org/10.30998/faktorexacta.v16i4.19564>
- Bergstra, J., & Bengio, Y. (2012). *Random Search for Hyper-Parameter Optimization*. 13, 281–305.
- Beyza, M., Elham, Y., Yasin, T., & Koklu, M. (2024). Fisheye freshness detection using common deep learning algorithms and machine learning methods with a developed mobile application. *European Food Research and Technology*, 1919–1932. <https://doi.org/10.1007/s00217-024-04493-0>
- Deng, T. (2023). *Effect of the Number of Hidden Layer Neurons on the Accuracy of the Back Propagation Neural Network*. 74, 462–468.
- Feng, J., & Lu, S. (2019). *Performance Analysis of Various Activation Functions in Artificial Neural Networks*. <https://doi.org/10.1088/1742-6596/1237/2/022030>
- Firmansyah, H. B., Syauqy, D., Hannats, M., & Ichsan, H. (2019). *Implementasi Sistem Penentuan Kesegaran Daging Sapi Lokal Berdasarkan Warna dan Kadar Amonia Dengan Metode Jaringan Saraf Tiruan Berbasis Embedded System*. 3(4).
- R. Hidayat, K. (2024). *Implementasi Sensor Gas untuk Mendeteksi Kesegaran Daging Menggunakan Metode Neural Network*. 10(2), 539–545.
- Rahmana, M. A. F., & Syauqy, D. (2019). *Sistem Deteksi Lama Waktu Penyimpanan Daging Ayam Berdasarkan Warna Dan Kadar Amonia Berbasis Sensor TCS3200 dan MQ135 Dengan Metode Jaringan Syaraf Tiruan*. 3(2), 1908–1916.
- Rahmawati, L., Hakimuddin, A. M., & Umami, I. (n.d.). *JURNAL INTAKE Implementasi Sensor Gas MQ-136 Dan MQ-137 Untuk Mendeteksi Kesegaran Daging Sapi Menggunakan Metode Neural Network*. 26–36.
- Sazli. (2006). *A BRIEF REVIEW OF FEED-FORWARD NEURAL NETWORK*.