

RANCANG BANGUN SORTASI BUAH TOMAT MENGUNAKAN METODE *FUZZY LOGIC* DENGAN MEMANFAATKAN ENERGI MATAHARI BERBASIS MIKROKONTROLER

Shadiq Taqwa Afrianto¹, Muh. Aldiansyah², Abdul Hafid³, Adriani⁴

^{1,2,3,4}Program Studi Teknik Elektro, Fakultas Teknik, Universitas Muhammadiyah Makassar

E-mail: shadiqtaqwaafriantol1@gmail.com¹, muhaldiansyah183@gmail.com²,
abdulhafid@unismuh.ac.id³, adriani@unismuh.ac.id⁴

Abstract - As is commonly known, tomatoes have a variety of ripe color variations. Therefore, for a farmer, the process of sorting tomatoes is important. Through this sorting step, farmers are able to determine suitable destinations for marketing their tomatoes, such as traditional markets, supermarkets and other places. Sorting tomatoes manually not only takes a long time, but also requires a high level of concentration to avoid mistakes when sorting. Sometimes, this level of concentration can be disturbed by factors such as fatigue, and this can result in sorting being carried out without proper attention to the condition of the tomatoes. To overcome these problems, the authors offer an automatic tool for sorting tomatoes. This tool applies the Sugeno fuzzy logic principle, a method that relies on a vague approach to decision making, in the sense that a value can be true or false depending on the degree of membership. The design of this tool uses an Arduino microcontroller as a control and a TCS3200 sensor to determine the RGB value by utilizing solar energy (solar panels) as a power source. By implementing this tool, it is hoped that problems in the tomato sorting process can be resolved, while reducing the use of time and costs. The fuzzy logic approach is used in processing tomato fruit values which include the range of RGB values generated by the TCS3200 color sensor. The result of this research is a tool that can automatically separate ripe, semi-ripe and unripe fruit according to the condition of the tomatoes.

*Intisari - Seperti yang umum diketahui, buah tomat memiliki variasi warna kematangan yang beragam. Oleh karena itu, bagi seorang petani, proses penyortiran buah tomat menjadi penting. Melalui langkah penyortiran ini, petani mampu menentukan destinasi yang sesuai untuk pemasaran buah tomatnya, seperti pasar tradisional, supermarket, dan tempat lainnya. Melakukan penyortiran buah tomat secara manual tidak hanya menghabiskan waktu yang cukup lama, tetapi juga memerlukan tingkat konsentrasi yang tinggi guna menghindari kesalahan saat melakukan penyortiran. Terkadang, tingkat konsentrasi ini dapat terganggu oleh faktor-faktor seperti kelelahan, dan hal ini dapat mengakibatkan penyortiran dilakukan tanpa atensi yang tepat terhadap keadaan buah tomat. Untuk mengatasi permasalahan berikut, penulis menawarkan sebuah alat otomatis untuk menyortir buah tomat. Alat ini menerapkan prinsip *fuzzy logic sugeno*, sebuah metode yang mengandalkan pendekatan samar untuk pengambilan keputusan, dalam artian suatu nilai bisa saja bernilai benar dan salah tergantung derajat keanggotaannya. Pada perancangan alat ini menggunakan mikrokontroler arduino sebagai control dan sensor TCS3200 untuk menentukan nilai RGB dengan memanfaatkan energi matahari (Panel surya) sebagai sumber tenaga. Dengan*

penerapan alat ini, diharapkan permasalahan dalam proses penyortiran buah tomat dapat teratasi, sekaligus mengurangi penggunaan waktu dan biaya. Pendekatan fuzzy logic digunakan dalam pemrosesan nilai buah tomat yang mencakup rentang nilai RGB yang dihasilkan oleh sensor warna TCS3200. Hasil dari penelitian ini adalah alat yang dapat memisahkan secara otomatis antara buah yang matang, setengah matang, dan mentah sesuai dengan kondisi buah tomat.

Kata Kunci: Logika Fuzzy Sugeno, Sensor Warna TCS3200, Buah Tomat, Arduino Uno, Panel Surya.

I. PENDAHULUAN

Tomat atau Rangam (*Solanum lycopersium syn. Lycopersicon esculentum*) adalah tumbuhan dari keluarga *Solanaceae*, tumbuhan asli Amerika Tengah dan Selatan dari Meksiko sampai Peru. Tomat merupakan tumbuhan siklus hidup singkat, dapat tumbuh setinggi 1 sampai 3 meter. Tumbuhan ini memiliki buah berwarna hijau, merah, dan kuning yang biasa dimasak sebagai sayur atau dimakan secara langsung tanpa diproses. [1]

Seperti yang telah diketahui, buah tomat memiliki variasi dalam hal warna kematangan dan ukuran yang sangat bervariasi. Oleh karena itu, bagi seorang petani, penting untuk melakukan proses penyortiran pada buah tomat. Dengan melakukan ini, petani dapat mengarahkan pemasaran buah tomat mereka ke berbagai pasar, termasuk supermarket dan lainnya.

Pada penyortiran buah tomat yang saat ini masih dilakukan secara manual, penulis berkeinginan untuk memanfaatkan kemajuan teknologi untuk menciptakan sebuah perangkat penyortir buah tomat.

Proses penyortiran buah tomat saat ini masih terdapat beberapa masalah yang penting untuk dilakukan penelitian, terutama untuk perancangan alat. Oleh karena itu, perancangan alat penyortir buah tomat ini diharapkan dapat mengatasi hal tersebut. Alat ini sengaja dibuat untuk memudahkan dalam penyortiran buah tomat berdasarkan warna, dengan begitu petani dapat dengan mudah dalam menentukan strategi pemasaran produk berdasarkan jarak antara pasar dan kebun.

II. TINJAUAN PUSTAKA

A. Buah Tomat

Tomat atau Rancam (*Solanum lycopersium syn. Lycopersicon esculentum*) adalah sayuran buah yang tergolong tanaman semusim berbentuk perdu dan termasuk kedalam family *solanaceae*. Tomat merupakan tumbuhan siklus hidup singkat, dapat tumbuh setinggi 1 sampai 3 meter. [1]

B. Representasi Warna

Representasi warna umumnya terdiri dari tiga komponen warna utama yang ada dalam sebuah citra, yaitu merah (red), hijau (green), dan biru (blue). Kombinasi intensitas ketiga warna ini membentuk berbagai macam warna lain tergantung pada seberapa kuat masing-masing warna ini dalam cahaya yang diterimanya. Ketika ketiga warna RGB ini digabungkan dengan intensitas minimum, hasilnya akan membentuk warna hitam. [2], [3]

C. Mikrokontroler

Mikrokontroler merupakan sebuah sistem mikroprosesor dimana didalamnya sudah terdapat CPU, ROM, RAM, I/O clock dan peripheral lainnya, mikrokontroler juga dapat digunakan sebagai pengatur kecil untuk perangkat sistem internal. [4]

D. Software dan Hardware Pendukung

1. Panel Surya

Panel surya merupakan salah satu sumber energi yang terbaru dengan teknologi fotovoltaik, yang menggunakan perangkat semikonduktor untuk mengubah energi sinar matahari menjadi energi listrik. [5][6]



Gbr. 1 panel surya

2. Solar Charger Controller

Solar *charger controller* rangkaian elektronika yang berfungsi untuk mengatur aliran arus listrik DC yang mengalir ke baterai dan sebaliknya, yaitu mengambil daya dari baterai untuk digunakan oleh beban. [7]



Gbr. 2 Solar Charger Controller

3. Inverter

Inverter Inverter adalah jenis perangkat elektronika daya yang mengubah tegangan DC menjadi tegangan AC. [7]



Gbr. 3 Inverter

4. Akumulator (Accu)

Akumulator (*Accu*) adalah tegangan searah (DC) mengubah energi kimia menjadi energi listrik dalam suatu akumulasi (*Accu*), yang sering disebut dengan baterai. [7]



Gbr. 4 Akumulator (Accu)

5. Arduino Uno

Arduino Uno adalah papan mikrokontroler berdasarkan ATmega328. Arduino Uno memiliki 14 pin informasi hasil komputerisasi dimana 6 pin informasi tersebut dapat digunakan sebagai hasil PWM dan 6 pin input analog, 16 MHz osilator kristal, koneksi USB, jack power, ICSP header, dan tombol reset. [8][9]



Gbr. 5 Arduino Uno

6. Sensor Warna TCS3200

Sensor warna TCS3200 adalah suatu DT-sense colour sensor. TCS3200 adalah sebuah Integrated Circuit (IC) yang telah diprogram untuk mengubah informasi warna menjadi sinyal berfrekuensi yang diekspresikan dalam bentuk sinyal kotak. Sensor warna ini dilengkapi dengan sejumlah filter cahaya yang berfungsi untuk mengidentifikasi warna dasar seperti merah (R), hijau (G), dan biru (B). [2][10]



Gbr. 6 Sensor TCS3200

7. Motor DC

Motor DC adalah jenis motor listrik yang cara kerjanya memanfaatkan sumber tegangan DC. Motor DC memerlukan suplai tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik. [11]



Gbr. 7 Motor DC

8. Motor Servo

Motor servo merupakan suatu alat putar (aktuator) yang diatur dengan kontrol umpan balik atau loop tertutup untuk memastikan dan menentukan posisi sudut poros keluaran motor. [5]



Gbr. 8 Motor Servo MG996R

9. Driver Motor L298N

Agar arduino dapat melakukan *controlling* terhadap motor DC digunakan driver motor. Pada dasarnya rangkaian penggerak driver motor L298N hanya mengontrol tegangan dan momentum sehingga kecepatan dan arah motor dapat diatur sesuai keinginan. [12]



Gbr. 9 Driver Motor L298N

E. Fuzzy Logic

Logika *Fuzzy* merupakan suatu logika yang memiliki nilai kekaburan atau kesamaran antara benar atau salah. Dalam

teori logika fuzzy, suatu nilai dapat memiliki kedekatan dengan kategori benar atau salah secara bersamaan. Namun, sebagian besar penentuan kebenaran atau kesalahan tergantung pada tingkat keanggotaan yang diberikan. Logika fuzzy menggunakan derajat keanggotaan yang berkisar antara 0 hingga 1. Derajat keanggotaan 0 (FALSE) mengindikasikan bahwa nilai tersebut bukanlah anggota himpunan yang relevan, sedangkan nilai 1 (TRUE) menunjukkan bahwa nilai tersebut adalah anggota penuh dari himpunan yang dimaksud. [13]

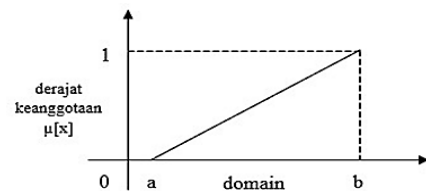
1. Fungsi Keanggotaan *Fuzzy*

Fungsi keanggotaan yang dapat digunakan untuk mendapatkan nilai keanggotaan dideskripsikan dalam berbagai bentuk kurva, seperti linier, segitiga, trapesium. [14]

a. Representasi Linear

1) Representasi Linear Naik

Dikatakan naik yaitu saat mengalami kenaikan dari derajat keanggotaan nol dan bergerak ke kanan menuju keanggotaan yang lebih tinggi (menuju satu). [14]



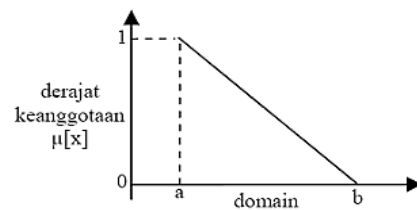
Gbr. 10 Linear Naik

Fungsi keanggotaan antara lain:

$$\mu[x] = \begin{cases} 0; & x \leq a \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ 1; & x \geq b \end{cases} \quad (1)$$

2) Representasi Linear Turun

Dikatakan turun saat mengalami penurunan dari derajat keanggotaan satu dan bergerak ke kanan menuju keanggotaan yang lebih rendah (menuju nol). [14]



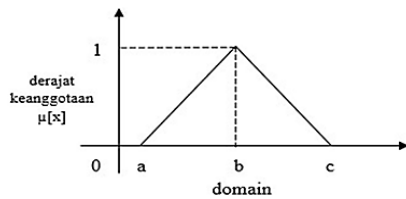
Gbr. 11 Linear Turun

Fungsi keanggotaan antara lain:

$$\mu[x] = \begin{cases} \frac{b-x}{b-a}; & a \leq x \leq b \\ 0; & x \geq b \end{cases} \quad (2)$$

3) Fungsi keanggotaan segitiga

Fungsi ini merupakan gabungan dari representasi linear dan mempunyai tiga parameter yaitu a, b dan c. [7]



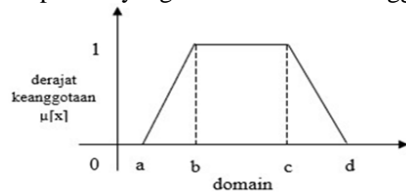
Gbr. 12 Fungsi Keanggotaan Segitiga

Fungsi keanggotaan segitiga yaitu:

$$\mu[x] \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ \frac{b-x}{c-b}; & b \leq x \leq c \end{cases} \quad (3)$$

4) Fungsi Keanggotaan Trapesium

Fungsi ini pada dasarnya berbentuk segitiga tetapi terdapat beberapa titik yang memiliki nilai keanggotaan 1. [7]



Gbr. 13 Fungsi Keanggotaan Trapesium

Fungsi keanggotaan trapesium antara lain:

$$\mu[x] \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}; & c \leq x \leq d \end{cases} \quad (4)$$

Logika *Fuzzy* memiliki beberapa metode dalam perhitungannya, salah satunya adalah metode Sugeno. Menurut model fuzzy tersebut, terdapat langkah-langkah yang harus diikuti dalam menghitung menggunakan metode fuzzy Sugeno, yang meliputi: [13]

1. Tahapan *Fuzzyfikasi*

Fuzzyfikasi adalah langkah dalam proses yang mengubah variabel *non-fuzzy* menjadi variabel linguistik dan kemudian mengelompokkannya berdasarkan himpunan *fuzzy*. Tujuan dari langkah ini adalah untuk memperoleh nilai-nilai derajat keanggotaan. [13]

2. Implikasi

Implikasi adalah proses mencari nilai minimum (atau nilai terkecil) dari aturan yang telah dibentuk adalah bagaimana

mendapatkan nilai keluaran dari aturan IF-THEN. Hal ini dikarenakan pada tahap penyusunan *rulebase* Sugeno menggunakan administrator AND. Berikutnya adalah rumus yang digunakan untuk menemukan nilai dasar pada tahap implikasi. [13]

$$\mu_{A \cap B} = \min(\mu_A(x), \min(\mu_B(x))) \quad (5)$$

Keterangan:

$\mu_{A \cap B}$ = Nilai minimum himpunan *fuzzy* A dan B pada aturan ke-i

$\mu_A(x)$ = Derajat keanggotaan *fuzzy* A pada aturan ke-i

$\mu_B(x)$ = Derajat keanggotaan *fuzzy* B pada aturan ke-i

3. Tahapan *Defuzzyfikasi*

Defuzzyfikasi merupakan kebalikan dari proses *fuzzyfikasi*. Dalam metode perhitungan Sugeno, proses *defuzzyfikasi* menggunakan *Weight Average* (WA) dengan rumus sebagai berikut. [13]

$$Z = \frac{Q_1 Z_1 + Q_2 Z_2 + \dots + Q_n Z_n}{Q^1 + Q^2 + \dots + Q_n} \quad (6)$$

Keterangan:

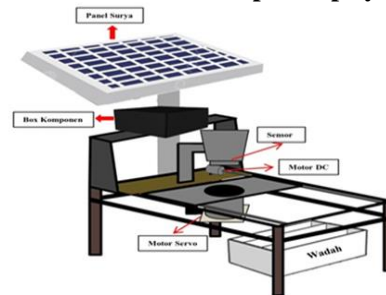
Z_n = nilai yang telah ditetapkan berdasarkan variabel output.
 α -predikat = nilai yang dihasilkan dari proses implikasi.

III. METODE PENELITIAN

Alat yang digunakan dalam penelitian ini: penggaris, tang, multimeter, solder, timah, gergaji, obeng.

Bahan penelitian yang digunakan adalah: arduino uno, sensor TCS3200, motor dc 12volt, motor servo, adaptor 12v/10A, kabel jumper, *solar cell*, *charger controller*, *inverter*, aki[15].

A. Gambaran umum dari desain prototipe yang diusulkan



Gbr. 14 Ilustrasi Perancangan Alat

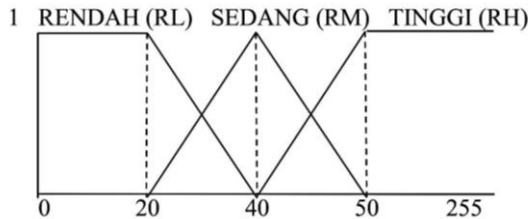
B. Diagram Blok Rangkaian Penyortir Buah Tomat

Berikut adalah penjelasan dari diagram blok rangkaian penyortir buah tomat yang dapat dilihat pada gambar 15:

Proses perhitungan fuzzyfikasi dalam penelitian ini, peneliti menggunakan rumus kurva segitiga.

Terdapat 3 himpunan fuzzy sebagai berikut:

a. Himpunan Nilai R (Red)



Gbr. 18 Grafik Fungsi Merah (R)

Dari ilustrasi grafik pada gambar 18, dapat di rumuskan derajat keanggotaan sebagai berikut::

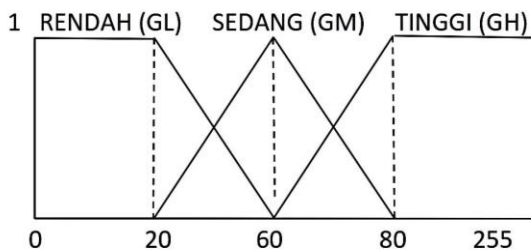
$$\mu_{RL} = \begin{cases} 1 & ; x \leq 20 \\ \frac{40-x}{40-20} & ; 20 \leq x \leq 40 \\ 0 & ; x \geq 40 \end{cases}$$

$$\mu_{RM} = \begin{cases} 0 & ; x \leq 20 \text{ or } x \geq 60 \\ \frac{x-20}{40-20} & ; 20 \leq x \leq 40 \\ \frac{60-x}{60-40} & ; 40 \leq x \leq 60 \end{cases}$$

$$\mu_{RH} = \begin{cases} 1 & ; x \geq 60 \\ \frac{x-40}{60-40} & ; 40 \leq x \leq 60 \\ 0 & ; x \leq 40 \end{cases}$$

RL, RM, dan RH merupakan tiga kondisi keanggotaan dalam grafik pada rumus. Bagian RL antara 0 dan 20, sedangkan bagian RM antara 20 dan 50, dan bagian RH antara 50 dan 255.

b. Himpunan Nilai G (Green)



Gbr. 19 Grafik Fungsi Green (G)

Dari ilustrasi grafik ke anggotaaan pada gambar 19 dapat di rumuskan derajat ke anggotaaannya sebagai berikut;

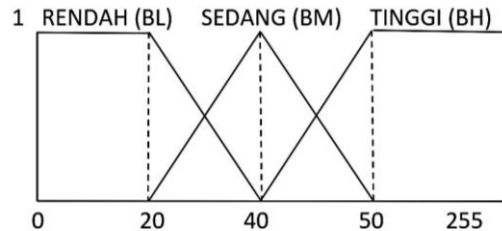
$$\mu_{GL} = \begin{cases} 1 & ; x \leq 20 \\ \frac{20-x}{40-20} & ; 20 \leq x < 40 \\ 0 & ; x \geq 40 \end{cases}$$

$$\mu_{GM} = \begin{cases} 0 & ; x \leq 20 \text{ or } x \geq 60 \\ \frac{x-20}{60-20} & ; 20 \leq x \leq 40 \\ \frac{60-x}{80-60} & ; 40 \leq x \leq 60 \end{cases}$$

$$\mu_{GH} = \begin{cases} 1 & ; x \geq 60 \\ \frac{x-40}{70-60} & ; 60 \leq x \leq 80 \\ 0 & ; x \leq 60 \end{cases}$$

Dalam rumus ini, ada tiga kondisi yang dapat diidentifikasi, yaitu GL, GM, dan GH. Bagian GL mencakup nilai-nilai antara 0 hingga 40, sementara GM mencakup rentang dari 40 hingga 70, dan bagian GH melibatkan nilai-nilai yang berada di antara 80 hingga 255.

c. Himpunan Nilai B (Blue)



Gbr. 20 Grafik Fungsi Blue (B)

Dari grafik ke anggotaaan pada gambar 20 dapat dirumuskan derajat ke anggotaaannya sebagai berikut;

$$\mu_{BL} = \begin{cases} 1 & ; x \leq 20 \\ \frac{40-x}{40-20} & ; 20 \leq x \leq 40 \\ 0 & ; x \geq 40 \end{cases}$$

$$\mu_{BM} = \begin{cases} 0 & ; x \leq 20 \text{ or } x \geq 60 \\ \frac{x-20}{40-20} & ; 20 \leq x \leq 40 \\ \frac{60-x}{60-40} & ; 40 \leq x \leq 60 \end{cases}$$

$$\mu_{BH} \begin{cases} 1 & ; & x \geq 60 \\ \frac{x - 40}{60 - 40} & ; & 40 \leq x \leq 60 \\ 0 & ; & x \leq 40 \end{cases}$$

Dari rumus tersebut ada tiga bagian kondisi dalam grafik fungsi keanggotaan: BL, BM, dan BH. Bagian BL berkisar dari 0 hingga 20, sedangkan bagian BM berkisar dari 20 hingga 50, dan bagian BH berkisar dari lebih dari 50 hingga 255.

2. Rule Evaluation

Tabel 2 Rule Base Sistem

RULE	RED	GREEN	BLUE	KEMATANGAN TOMAT
R0	Rendah	Rendah	Rendah	Mentah
R1	Rendah	Rendah	Sedang	Mentah
R2	Rendah	Rendah	Tinggi	Mentah
R3	Rendah	Sedang	Rendah	Setengah Matang
R4	Rendah	Sedang	Sedang	Setengah Matang
R5	Rendah	Sedang	Tinggi	Setengah Matang
R6	Rendah	Tinggi	Rendah	Setengah Matang
R7	Rendah	Tinggi	Sedang	Matang
R8	Rendah	Tinggi	Tinggi	Setengah Matang
R9	Sedang	Rendah	Rendah	Matang
R10	Sedang	Rendah	Sedang	Matang
R11	Sedang	Rendah	Tinggi	Tidak Terdeteksi
R12	Sedang	Sedang	Rendah	Tidak Terdeteksi
R13	Sedang	Sedang	Sedang	Tidak Terdeteksi
R14	Sedang	Sedang	Tinggi	Mentah
R15	Sedang	Tinggi	Rendah	Mentah
R16	Sedang	Tinggi	Sedang	Matang
R17	Sedang	Tinggi	Tinggi	Matang
R18	Sedang	Rendah	Rendah	Matang
R19	Tinggi	Rendah	Sedang	Setengah Matang
R20	Tinggi	Rendah	Tinggi	Setengah Matang
R21	Tinggi	Sedang	Rendah	Mentah
R22	Tinggi	Sedang	Sedang	Mentah
R23	Tinggi	Sedang	Tinggi	Matang
R24	Tinggi	Tinggi	Rendah	Tidak Terdeteksi
R25	Tinggi	Tinggi	Sedang	Tidak Terdeteksi
R26	Tinggi	Tinggi	Tinggi	Tidak Terdeteksi

Pada tabel 2 terdapat rule dimana R0 – R26 yang akan menyajikan kondisi rendah, sedang atau tinggi pada tiap himpunan *Red*, *Gree*, and *Blue* untuk menentukan suatu tindakan apakah termasuk kategori mentah, setengah matang dan matang pada sebuah tomat.

3. Fungsi Implikasi

Fungsi implikasi yang digunakan pada proses ini yaitu fungsi (MIN) minimum, dimana derajat keanggotaan yang diambil adalah nilai minimum dari input maupun output. Berikut ini rumus untuk mencari nilai implikasi lihat rumus (2.8).

$$\begin{aligned} \text{Tidak Terdeteksi} &= 0 \\ \text{Mentah} &= 1 \\ \text{Setengah Matang} &= 2 \\ \text{Matang} &= 3 \end{aligned}$$

[R0] Jika nilai merah rendah (RL) dan nilai hijau rendah (GL) dan nilai biru rendah (BL), maka mentah.

$$\begin{aligned} \alpha\text{-predikat}_1 &= \min(\mu \text{ Merah Rendah [1]}, \mu \text{ Hijau Rendah [1]}, \\ &\quad \mu \text{ Biru Rendah [1]}) \\ &= \min(1 \ 1 \ 1) \\ &= 1 \end{aligned}$$

$$Z_1 = 1$$

[R26] Jika nilai merah tinggi (RH) dan nilai hijau tinggi (GH) dan nilai biru tinggi (BH), maka tidak terdeteksi.

$$\begin{aligned} \alpha\text{-predikat}_2 &= \min(\mu \text{ Merah Tinggi [0]}, \mu \text{ Hijau Tinggi [0]}, \mu \\ &\quad \text{Biru Tinggi [0]}) \\ &= \min(0 \ 0 \ 0) \\ &= 0 \end{aligned}$$

$$Z_2 = 0$$

[R23] Jika nilai merah tinggi (RH) dan nilai hijau sedang (GM) dan nilai biru tinggi (BH), maka matang.

$$\begin{aligned} \alpha\text{-predikat}_3 &= \min(\mu \text{ Merah tinggi [0]}, \mu \text{ Hijau Sedang [0]}, \mu \\ &\quad \text{Biru Tinggi [0]}) \\ &= \min(0 \ 0 \ 0) \\ &= 0 \end{aligned}$$

$$Z_3 = 3$$

[R1] Jika nilai merah rendah (RL) dan nilai hijau rendah (GL) dan nilai biru sedang (BM), maka mentah.

$$\begin{aligned} \alpha\text{-predikat}_4 &= \min(\mu \text{ Merah Rendah [1]}, \mu \text{ Hijau Rendah [1]}, \\ &\quad \mu \text{ Biru Sedang [0]}) \\ &= \min(1 \ 1 \ 0) \\ &= 0 \end{aligned}$$

$$Z_4 = 1$$

[R8] Jika nilai merah rendah (RL) dan nilai hijau tinggi (GH) dan nilai biru tinggi (BH), maka setengah matang.

$$\begin{aligned} \alpha\text{-predikat}_5 &= \min(\mu \text{ Merah Rendah [0]}, \mu \text{ Hijau tinggi [0]}, \mu \\ &\quad \text{Biru tinggi [0]}) \\ &= \min(1 \ 0 \ 0) \\ &= 0 \end{aligned}$$

$$Z_5 = 2$$

[R9] Jika nilai merah sedang (RM) dan nilai hijau rendah (GL) dan nilai biru rendah (BL), maka tidak terdeteksi.

$$\begin{aligned} \alpha\text{-predikat}_6 &= \min(\mu \text{ Merah sedang [0]}, \mu \text{ Hijau Rendah [1]}, \\ &\quad \mu \text{ Biru Sedang [0]}) \\ &= \min(0 \ 1 \ 1) \\ &= 0 \end{aligned}$$

$$Z_6 = 2$$

4. Proses Defuzzifikasi

Setelah mendapatkan nilai MIN dari masing-masing rule yang terpenuhi, maka langkah berikutnya yaitu mencari

bilai rata-ratanya menggunakan rumus (weighted average) WA. Pada proses defuzzyfikasi menggunakan rumus (2.9) sebagai berikut:

Rumus:

$$z: \frac{(\alpha - \text{predikat}_1 * z_1) + (\alpha - \text{predikat}_2 * z_2) + \dots + (\alpha - \text{predikat}_n * z_n)}{\alpha - \text{predikat}_1 + \alpha - \text{predikat}_2 + \dots + \alpha - \text{predikat}_n}$$

a. Perhitungan mencari tidak terdeteksi

$$\text{Output} = (\text{rule0} \times \text{mentah}) + (\text{rule26} \times \text{tidak}) + (\text{rule3} \times \text{matang}) + (\text{rule1} \times \text{mentah}) + (\text{rule8} \times \text{setengah}) + (\text{rule9} \times \text{setengah}) / (\text{rule0} + \text{rule26} + \text{rule3} + \text{rule1} + \text{rule8} + \text{rule9})$$

$$\text{Output} = \frac{(0 \times 1) + (1 \times 0) + (0 \times 3) + (0 \times 1) + (0 \times 3) + (0 \times 2)}{0 + 1 + 0 + 0 + 0 + 0}$$

$$\text{Output} = \frac{0 + 0 + 0 + 0 + 0 + 0}{1}$$

$$\text{Output} = \frac{0}{1}$$

$$\text{Output} = 0$$

b. Proses perhitungan mencari nilai mentah

$$\text{Output} = (\text{rule0} \times \text{mentah}) + (\text{rule26} \times \text{tidak}) + (\text{rule23} \times \text{matang}) + (\text{rule1} \times \text{mentah}) + (\text{rule8} \times \text{setengah}) + (\text{rule9} \times \text{setengah}) / (\text{rule0} + \text{rule26} + \text{rule3} + \text{rule1} + \text{rule8} + \text{rule9})$$

$$\text{Output} = \frac{(1 \times 1) + (0 \times 0) + (0 \times 3) + (0 \times 1) + (0 \times 2) + (0 \times 2)}{1 + 0 + 0 + 0 + 0 + 0}$$

$$\text{Output} = \frac{1 + 0 + 0 + 0 + 0 + 0}{1}$$

$$\text{Output} = \frac{1}{1}$$

$$\text{Output} = 1$$

c. Mencar nilai setengah matang

$$\text{Output} = (\text{rule0} \times \text{mentah}) + (\text{rule26} \times \text{tidak}) + (\text{rule23} \times \text{matang}) + (\text{rule1} \times \text{mentah}) + (\text{rule8} \times \text{setengah}) + (\text{rule9} \times \text{setengah}) / (\text{rule0} + \text{rule26} + \text{rule3} + \text{rule1} + \text{rule8} + \text{rule9})$$

$$\text{Output} = \frac{(0 \times 1) + (0 \times 0) + (0 \times 3) + (0 \times 1) + (0 \times 3) + (1 \times 2)}{0 + 0 + 0 + 0 + 0 + 1}$$

$$\text{Output} = \frac{0 + 0 + 0 + 0 + 0 + 2}{1}$$

$$\text{Output} = \frac{2}{1}$$

$$\text{Output} = 2$$

d. Proses mencari nilai matang

$$\text{Output} = (\text{rule0} \times \text{mentah}) + (\text{rule26} \times \text{tidak}) + (\text{rule23} \times \text{matang}) + (\text{rule1} \times \text{mentah}) + (\text{rule8} \times \text{setengah}) + (\text{rule9} \times \text{setengah}) / (\text{rule0} + \text{rule26} + \text{rule3} + \text{rule1} + \text{rule8} + \text{rule9})$$

$$\text{Output} = \frac{\text{rule8} + \text{rule9}}{(0 \times 1) + (0 \times 0) + (1 \times 3) + (0 \times 1) + (0 \times 3) + (0 \times 2)}$$

$$\text{Output} = \frac{0 + 0 + 1 + 0 + 0 + 0}{0 + 0 + 3 + 0 + 0 + 0}$$

$$\text{Output} = \frac{1}{3}$$

$$\text{Output} = \frac{3}{1}$$

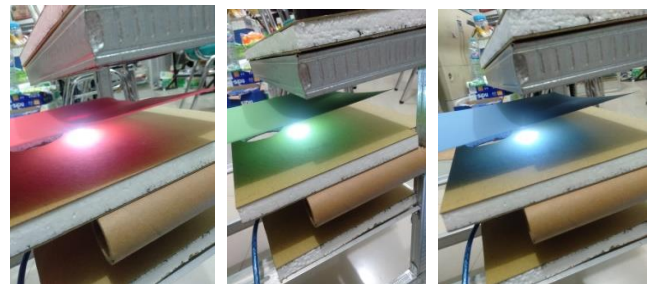
$$\text{Output} = 3$$

B. Pengujian

Setelah semua bagian telah disatukan dan sistem siap untuk dijalankan, pengujian sistem dilakukan. Tujuan pengujian sistem adalah untuk memberi tahu penulis apakah sistem berfungsi sebagaimana dimaksud.

1. Menguji sensor warna TCS3200

Pada pengujian ini, digunakan tiga warna dasar yang sudah ditentukan, yaitu Merah, Hijau, dan Biru, yang ditempatkan di sekitar sensor. Selanjutnya, nilai citra warna R, G, dan B yang dibaca oleh sensor akan dipantau melalui layar terminal serial di Arduino. Pengujian ini bertujuan untuk menilai sejauh mana kemampuan sensor dalam menjalankan tugasnya secara efisien.



Gbr 21. Pengujian Sensor TCS3200

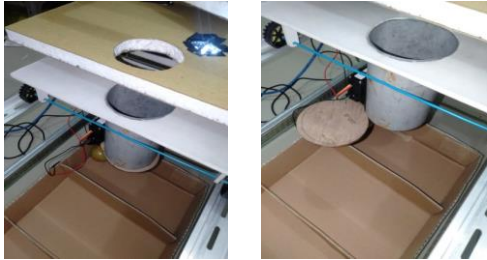
2. Menguji Motor Servo

Tujuan dari pengujian ini adalah untuk mendapatkan posisi rotasi yang akurat untuk servo. Hal ini dicapai dengan memodifikasi nilai sudut rotasi pada program yang mengatur arah rotasi motor servo. Hasil dari penyesuaian sudut rotasi pada motor servo dapat dilihat dalam ilustrasi pada Gambar 22 dan Gambar 23.



Gbr.22 Hasil Pengujian Motor Servo Untuk Mengarahkan Buah ke Penampung

Pada gambar 23 merupakan hasil pengujian motor servo dimana dalam keadaan ini, fungsi dari motor servo adalah untuk mengalokasikan arah buah tomat menuju sensor dan kemudian ke wadah penampung..



Gbr. 23 Hasil Pengujian Motor Servo Untuk

3. Proses Pengujian Fuzzifikasi

Untuk menguji kemampuan sistem dalam memproses sensor TCS3200 menggunakan metode fuzzifikasi yang efisien, tomat ditempatkan pada sensor dan hasil proses fuzzifikasi ditampilkan pada layar terminal Arduino. Hasil dari pengujian ketika sensor mengenali buah tomat yang belum matang terlihat dalam Gambar 24.

```
R: 20||G: 30||B: 18
RL: 1||RM: 0||RH: 0
GL: 1||GM: 0||GH: 0
BL: 1||BM: 0||BH: 0
```

Gbr 24. Hasil Fuzzifikasi Ketika Mendeteksi Tomat Mentah

Selain itu, pada saat buah tomat setengah matang terdeteksi sensor hasil fuzzifikasi dapat dilihat pada gambar 25.

```
R: 23||G: 28||B: 17
RL: 0||RM: 1||RH: 0
GL: 1||GM: 0||GH: 0
BL: 1||BM: 0||BH: 0
```

Gbr 25. Hasil Fuzzifikasi Ketika Mendeteksi Tomat Setengah Matang

Ketika buah tomat matang terdeteksi sensor maka fuzzifikasi yang dihasilkan pada terminal arduino dapat dilihat pada gambar 26

```
R: 56||G: 61||B: 56
RL: 0||RM: 0||RH: 1
GL: 0||GM: 1||GH: 0
BL: 0||BM: 0||BH: 1
```

Gbr 26. Hasil Fuzzifikasi Saat Mendeteksi Tomat Matang

4. Proses Pengujian Defuzzifikasi

Dengan mengidentifikasi buah tomat melalui sensor, langkah selanjutnya adalah memperhatikan hasil dari proses defuzzifikasi melalui perhitungan menggunakan metode *Weighted Average (Wa)*. Ini bertujuan untuk menghasilkan keluaran yang diinginkan, yaitu nilai 1 yang menunjukkan buah belum matang, nilai 2 yang mengindikasikan buah setengah matang, dan nilai 3 yang menggambarkan buah yang sudah matang. Detail hasil pengujian pada serial terminal Arduino dapat ditemukan dalam Tabel 3.

Tabel 3 Hasil Pengujian Defuzzifikasi

Pengujian	Nilai RGB	Rule	Defuzzifikasi
Mentah	R = 20 G = 30 B = 18	RL, GL, BL	R: 20 G: 30 B: 18 RL: 1 RM: 0 RH: 0 GL: 1 GM: 0 GH: 0 BL: 1 BM: 0 BH: 0 Nilai Deffuzi adalah: 1
Setengah Matang	R = 23 G = 28 B = 17	RM, GL,BL	R: 23 G: 28 B: 17 RL: 0 RM: 1 RH: 0 GL: 1 GM: 0 GH: 0 BL: 1 BM: 0 BH: 0 Nilai Deffuzi adalah: 2
Matang	R = 56 G = 61 B = 56	RH,GM,BH	R: 56 G: 61 B: 56 RL: 0 RM: 0 RH: 1 GL: 0 GM: 1 GH: 0 BL: 0 BM: 0 BH: 1 Nilai Deffuzi adalah: 3

5. Proses Pengujian Penyortiran Buah Tomat

a. Uji penyortiran pada tomat mentah

Untuk menguji penyortiran tomat mentah, hal pertama yang perlu dilakukan adalah mengarahkan tomat ke sensor. Setelah tomat dimasukkan ke dalam wadah, servo akan membuka katup reservoir dan memasukkan tomat ke dalam keranjang penyimpanan tomat mentah.



Gbr. 27 Deteksi warna pada tomat.

Berdasarkan hasil deteksi oleh sensor dan penilaian data yang dievaluasi sesuai aturan, motor DC akan memindahkan tomat mentah ke keranjang kemudian membuka wadah sehingga tomat jatuh ke dalam keranjang tomat mentah.



Gbr.28 Hasil pengujian Buah Tomat Mentah

b. Pengujian Penyortiran Tomat Setengah Matang

Langkah pertama dalam proses pengujian penyortiran adalah mengarahkan tomat ke arah sensor sebelum dimasukkan ke dalam wadah. Wadah yang terhubung dengan motor DC kemudian akan mengarahkan tomat ke arah keranjang tomat setengah matang, dan servo akan membuka katup penampung sehingga tomat dapat dijatuhkan ke dalam keranjang.



Gbr 29. Deteksi Pada Buah Tomat

Berdasarkan langkah-langkah serta hasil keluaran dari proses defuzzifikasi yang terkait dengan warna yang terdeteksi sensor, tomat tersebut diidentifikasi sebagai setengah matang. Dibawah ini adalah hasil pengujian dari buah tomat yang masih setengah matang setelah sistem penampung mengarahkannya ke keranjang.



Gbr. 30 Hasil Pengujian Buah Tomat Setengah Matang

c. Pengujian Penyortiran Pada Buah Tomat Matang

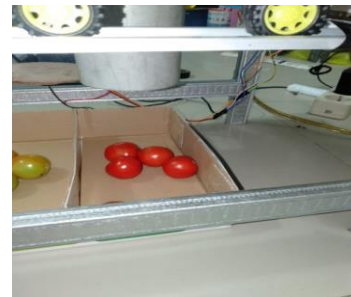
Pada uji sortasi tomat matang, langkah pertama adalah mengarahkan tomat ke sensor sebelum memasukkan tomat ke dalam penampung, keranjang untuk buah tomat matang

berada pada titik awal penampung, sehingga dalam pengujian ini motor DC yang terhubung dengan penampung tidak bergerak, dan servo segera membuka katub penampung untuk menjatuhkan tomat ke keranjang.



Gbr 31. Deteksi Pada Buah Tomat

Tomat berada dalam kategori matang sesuai dengan nilai output deteksi warna dan defuzzifikasi sistem; akibatnya, wadah yang terhubung ke motor DC tidak aktif, dan servo segera membuka katup reservoir untuk menjatuhkan tomat ke dalam keranjang tomat yang sudah matang.



Gbr 32. Hasil Pengujian Buah Tomat Matang

V. Kesimpulan

Berdasarkan hasil analisis, perancangan, dan pengujian yang dilakukan, maka dapat ditarik kesimpulan sebagai berikut:

1. Hasil pembuatan alat sortasi buah tomat diidentifikasi hanya berdasarkan pada warnanya, dengan mengukur nilai-nilai RGB melalui sensor TCS3200, yang mana berfungsi dengan efektif dalam membaca nilai RGB dan kemudian mengendalikan perangkat arduino sesuai hasil pembacaan tersebut.
2. Perancangan perangkat pemilah buah tomat ini telah berhasil sesuai rencana, dengan output yang mencakup nilai-nilai seperti berikut: 0 menunjukkan ketidakterdeteksian, 1 merepresentasikan tingkat kematangan mentah, 2 mengindikasikan tingkat kematangan setengah matang, dan 3 mewakili tingkat kematangan matang. Semua nilai ini dihasilkan melalui

perhitungan menggunakan rumus WA (*Weighted Average*).

3. Penerapan metode fuzzy logic dalam konteks menentukan warna pada sistem pengelolaan buah tomat menghasilkan output sesuai dengan yang telah direncanakan. Dalam hal ini, peneliti dapat berasumsi bahwa tomat yang matang sempurna tentunya akan berwarna merah cerah dan segar. Sedangkan tomat yang warna kekuningan merupakan tomat setengah matang dan yang berwarna hijau merupakan tomat yang masih mentah.

REFERENSI

- [1] C. Wasonowati, "Tomat atau Rangam (*Solanum lycopersium syn. Lycopersicum esculentum*)," *ISSN 1979 5777*, vol. 4, no. 1, pp. 21–27, 2011.
- [2] D. A. Nano, "Alat Pendeteksi Warna Menggunakan Sensor Warna Tcs3200. Alat Pendeteksi Warna. Menggunakan Sens. Warn. Tcs3200 Dan Arduino Nano," vol. 1, pp. 167–170, 2018.
- [3] R. Ridwang, I. Nurtanio, A. Ahmad Ilham, and S. Syafaruddin, "Deaf Sign Language Translation System with Pose and Hand Gesture Detection Under LSTM-Sequence Clasification Model," *ICIC Express Letters*, vol. 17, no. 7, pp. 809–816, 2023, doi: 10.24507/icicel.17.07.809.
- [4] I. M. N. Arijaya, "Rancang Bangun Alat Konveyor Untuk Sistem Soltir Barang Berbasis Mikrokontroler Arduino Uno," *Jurnal RESISTOR (Rekayasa Sistem Komputer)*, vol. 2, no. 2, pp. 126–135, 2019.
- [5] A. D. B. Tarigan, "Rancang Bangun Sistem Kendali Alat Penyortir Barang Berwarna Merah Dan Hijau Dengan Sensor TCS230 Berbasis PLC Schneider (Doctoral dissertation, undip)," 2018.
- [6] I. F. Purwoto, B. H., Jatmiko, J., Fadilah, M. A., & Huda, "Efisiensi Penggunaan Panel Surya Sebagai Sumber Energi Alternatif," *Emitor: Jurnal Teknik Elektro*, vol. 18, no. 1, pp. 10–14, 2018.
- [7] D. K. Iskandar, J., & Utami, "Penerapan Fuzzy Logic Untuk Meningkatkan Derajat Kebenaran Deteksi Pada Alat Bantu Buta Warna Berbasis Sensor Optik," *Komputasi: Jurnal Ilmiah Ilmu Komputer dan Matematika*, vol. 16, no. 1, pp. 195–202, 2019.
- [8] S. Fathurrahman, G., Nugraha, R., & Sumaryo, "Sistem Kendali Posisi Sel Surya Menggunakan Fuzzy Logic Controller," *e-Proceeding of Engineering*, vol. VI, no. 2, pp. 3061–3067, 2019.
- [9] A. A. Tyas, U. M., & Buckhari, "mplementasi Aplikasi Arduino Ide Pada Mata Kuliah Sistem Digital," *TEKNOS: Jurnal Pendidikan dan Teknologi*, vol. 1, no. 1, pp. 1–9, 2023.
- [10] B. Zulkarnain, I., Ramadhan, M., & Anwar, "Implementasi Alat Pendeteksi Warna Arduino. Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD Benda Menggunakan Fuzzy Logic dengan Sensor TCS3200 Berbasis," vol. 2, no. 2, pp. 106–117, 2019.
- [11] A. Arrahman, "TA: Rancang Bangun Pengontrol Pintu Air Menggunakan Metode Fuzzy Berdasarkan Perubahan Ketinggian Air (Doctoral dissertation, Institut Bisnis dan Informatika Stikom Surabaya)," 2019.
- [12] S. T. Pramudita, D., & Agus Ulinuha, "Protoype Sistem Buka Tutup Pintu Air Otomatis pada Persawahan Berbasis Arduino Uno (Doctoral dissertation, Universitas Muhammadiyah Surakarta)," 2017.
- [13] E. A. SARI, "Analisis Sistem Inferensi Fuzzy Metode Sugeno Orde Nol Untuk Data Skala Ordinal," 2016.
- [14] T. A. Oktavia, "Rancang Bangun Alat Pengering Pakaian Menggunakan Metode Fuzzy Logic," 2022.
- [15] R. Ridwang, A. Syafar, A. Indra Syahyadi, L. Anas, and T. Wahyuni, "SISTEM DETEKSI MASKER RUANGAN WAJIB MASKER DALAM MENGHADAPI ERA NEW NORMAL BERBASIS DEEP LEARNING," *Jurnal INSTEK*, vol. 7, no. 1, pp. 115–121, 2022.