

SYSTEMIC: Information System and Informatics Journal

ISSN: 2460-8092, 2548-6551 (e)

Vol 6 No 2 – Desember 2020

Pemanfaatan Image Mining Untuk Klasifikasi Dan Prediksi Kematangan Tomat Menggunakan Metode Jaringan Saraf Tiruan BackpropagationFirdaus¹, Nori Sahrin²¹ AMIK Boekittinggi² Sekolah Tinggi Ilmu Ekonomi Riaufirdaus6ta@gmail.com¹, norisahrin84@gmail.com²**Kata Kunci***Image Mining, Jaringan Saraf Tiruan, Backpropagation, Tomat, RGB***Abstrak**

Klasifikasi dan prediksi terhadap tingkat kematangan tomat secara otomatisasi menggunakan metode Jaringan Saraf Tiruan Backpropagation. Penentuan kematangan bidang pertanian masih diterapkan secara manual. Dengan adanya perkembangan teknologi di bidang image mining, penentuan kematangan tomat dapat dilakukan secara otomatisasi. Metode yang dipakai dalam pembuatan sistem ini ialah Jaringan Saraf Tiruan. Algoritma yang digunakan ialah backpropagation. Output kematangan tomat terdiri dari tiga kategori yaitu belum matang, setengah matang dan matang. Data training dan data testing sebanyak 60 buah yang digunakan. Arsitektur backpropagation pada penelitian ini berupa 3 input layer, 4 hidden layer, dan 1 output layer. Fungsi aktivasi yang digunakan dari input ke hidden layer ialah sigmoid biner, sedangkan dari hidden layer ke output ialah fungsi identitas (purelin). Ekstraksi citra dalam bentuk nilai minimum RGB berguna sebagai input. Diproses dan menghasilkan output tingkat kematangan dan prediksi kematangan. Hasil pengujian sistem data training memperoleh nilai keakuratan sebanyak 96,67% dan data testing senilai 90%.

Keywords*Image Mining, Artificial Neural Networks, Backpropagation, Tomato, RGB***Abstract**

Classification and prediction of tomato maturity level are automated using the Backpropagation Neural Network method. The determination of maturity in agriculture is still applied manually. With the development of technology in the field of image mining, determining the maturity of tomatoes can be done automatically. The method used in making this system is an Artificial Neural Network. The algorithm used is backpropagation. The output of tomato ripeness consists of three categories, namely immature, half ripe and ripe. 60 training data and testing data were used. The backpropagation architecture in this study consists of 3 input layers, 4 hidden layers, and 1 output layer. The activation function used from input to hidden layer is binary sigmoid, while from hidden layer to output is the identity function (purelin). Image extraction in the form of RGB minimum value is useful as input. Processed and produces output maturity level and maturity prediction. The results of testing the training data system obtained an accuracy value of 96.67% and testing data of 90%.

1. Pendahuluan

Indonesia merupakan negara agraris dengan kekayaan sumber daya alam yang melimpah dan posisi Indonesia yang strategis. Curah hujan yang tinggi, iklim yang tropis, serta tanah yang subur menjadi keuntungan yang dapat memudahkan tumbuhnya berbagai produksi pertanian. Salah satu hasil sektor pertanian ialah tomat. Tomat atau *Lycopersicon esculentum* menjadi salah satu produk pertanian yang bermanfaat untuk tubuh manusia^[1]. Berdasarkan data dari Badan Pusat Statistik dan

Direktoral Jenderal Hortikultura, diketahui tingkat produksi tomat di provinsi Sumatera Barat tahun 2017 mencapai 101,292 ton dan menduduki posisi kedua penghasil terbesar di Indonesia. Hal ini tentu merupakan peluang yang besar bagi kelompok tani yang ada di Sumatera Barat untuk menghasilkan tomat dengan kualitas yang baik dan menjaga mutu tomat agar tetap baik saat berada di tangan konsumen.

Namun penentuan tingkat kematangan tomat masih dilakukan secara manual melalui

pengamatan langsung visual manusia. Identifikasi dengan cara manual memiliki kelemahan yaitu membutuhkan waktu yang lama dan bersifat subjektif. Permasalahan utamanya ialah waktu perjalanan dari produsen hingga ke tangan konsumen. Selama waktu tersebut berlalu, proses pematangan tomat terjadi. Jika waktu yang digunakan tidak efektif, maka kemungkinan tomat untuk busuk sebelum dikonsumsi juga dapat terjadi sehingga dapat merugikan petani maupun penjual.

Teknologi informasi memungkinkan penentuan tingkat kematangan tomat secara otomatisasi yaitu menggunakan *data mining* dengan metode Jaringan Syaraf Tiruan (JST). JST adalah sistem komputasi dimana arsitekturnya meniru konsep dan cara kerja dari sel saraf biologis di dalam otak. Tujuannya untuk memberikan informasi kepada masyarakat yang membutuhkan dalam pengklasifikasian tingkat kematangan tomat dengan hasil yang lebih objektif, waktu yang cepat, dan hasil yang akurat. Serta diperlukan prediksi kematangan tomat yang efektif hingga sampai ke tangan konsumen. Jaringan Saraf Tiruan dapat diguahkan untuk klasifikasi buah karena menghasilkan tingkat akurasi yang lebih tinggi dibandingkan dengan metode lainnya^[2].

Beberapa tahun terakhir, teknik *image processing* telah banyak ditemukan semakin berguna dalam industri buah^[3]. Penelitian terkait *image processing* dengan metode jaringan saraf tiruan (*artificial neural network*) untuk bidang pertanian telah dilakukan oleh beberapa peneliti, seperti K. Sandhiya, dkk. dengan judul "Smart Fruit Classification using Neural Networks", Sella Kusumaningtyas dan Rosa Andrie A. dengan judul jurnal "Identifikasi Kematangan Buah Tomat Berdasarkan Warna Menggunakan Metode Jaringan Syaraf Tiruan", M. Angga Anggriawan, dkk. pada jurnalnya dengan judul "Pengenalan Tingkat Kematangan Tomat Berdasarkan Citra Warna Pada Studi Kasus Pembangunan Sistem Pemilihan Otomatis", Dila Deswari, dkk. dengan judul jurnal "Identifikasi Kematangan Buah Tomat Menggunakan Metoda Backpropagation", jurnal "Identifikasi Tingkat Ketuaan dan Kematangan Pepaya (*Carica papaya L.*) IPB 1 dengan Pengolahan Cira Digital dan Jaringan Syaraf Tiruan" oleh Enrico Syaefullah, Hadi K. Purwadaria, Sutrisno, dan Suroso.

Maka dari itu, penelitian menggunakan metode klasifikasi jaringan saraf tiruan untuk mengklasifikasikan tingkat kematangan tomat pada kelompok tani penghasil tomat berdasarkan warna menggunakan teknik pengolahan citra serta algoritma *backpropagation*. Salah satu keunggulan metode jaringan saraf tiruan ialah kemampuan yang dimiliki untuk melakukan prediksi. Algoritma *backpropagation* sangat bermanfaat, cukup handal serta cukup mudah dipahami. Objek yang diamati ialah tomat pada tingkat kematangan yang memiliki citra warna yang kontras dari warna buah

muda hingga matang. Citra diambil menggunakan kamera dengan kualitas pencahayaan yang sama serta sudut pengambilan yang sama.

2. Tinjauan Pustaka

2.1 Data Mining

Data mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terakit dari berbagai *database* besar^[4].

2.2 Image mining

Image mining ialah memahami gambar dengan pola tertentu agar mendapatkan informasi. Saat ini, *image mining* secara cepat mendapat perhatian diantara para peneliti di bidang *data mining*, *information retrieval*, dan *multimedia databases* karena potensinya yang berguna dalam menemukan pola gambar. Sistem *image mining* dapat secara otomatis mendapatkan informasi yang bernilai pengetahuan dari data gambar yang ada. *Image mining* bertujuan untuk penentuan pola gambar yang signifikan dalam koleksi gambar yang diberikan dan data alfanumerik terkait^[5].

2.3 Citra RGB

Citra RGB merupakan citra yang terdiri dari 3 kanal dengan komponen warna merah/*red* (R), hijau/*green* (G), biru/*blue* (B). Rentang nilai pada masing-masing kanal yaitu dari 0 – 255. Gambar 2.4 merupakan contoh citra RGB. Karena citra ini memiliki 3 kanal, maka untuk satu piksel pada citra ini memerlukan $3 \times 8\text{-bit} = 24\text{-bit}$. Citra berwarna disebut juga dengan citra 24-bit. Model warna RGB bertujuan untuk penginderaan dan presentasi gambar yang dilihat dalam bentuk visual pada berbagai alat elektronik seperti televisi dan fotografi^[6].

2.4 Pengolahan citra

Pengolan citra merupakan proses untuk mengubah kualitas citra atau gambar menjadi lebih baik, khususnya menggunakan perangkat komputer sebagai *hardware* (Munir, 2004). Seringkali citra yang asli mengalami penurunan kualitas akibat beberapa pengaruh seperti citra mengandung cacat atau derau (*noise*), warna citra terlalu kontras, kurang tajam (*blur*), dan lain sebagainya yang dapat memengaruhi informasi yang disampaikan. Citra tersebut tentu perlu diperbaiki agar kualitasnya menjadi lebih baik sehingga informasi yang ditangkap dapat akurat dan jelas. *Input* dari program pengolahan citra adalah citra dan *output* adalah citra pula. Namun *output* citra memiliki kualitas yang lebih bagus dibandingkan dengan *input* citra.

2.5 Tomat

Dalam bahasa latin disebut dengan

Lycopersicon esculentum adalah tumbuhan dari keluarga *Solanaceae*. Tomat merupakan salah satu jenis produk hortikultura yang dapat berguna untuk kesehatan tubuh, serta mempunyai peluang pasar yang cukup menjanjikan. Komposisi tomat mengandung gizi yang cukup lengkap. Salah satu perubahan yang menonjol dalam proses pematangan buah ialah perubahan warna (Winarno, 1981).

2.6 Jaringan Saraf Tiruan

Penelitian ini menggunakan salah satu dari metode dalam *data mining* yaitu Jaringan Saraf Tiruan. Jaringan Saraf Tiruan atau *Artificial Neural Network* ialah metode yang menerapkan sistem jaringan saraf hidup makhluk hidup terutama manusia dengan mengadopsi cara kerja neuron secara biologi fokus pada cara kerja otak. Jaringan saraf tiruan terdiri dari kumpulan *node* (neuron) dan relasi. Ada tiga tipe *node*, yaitu *input*, *hidden*, dan *output*.

2.7 Metode *backpropagation*

Metode ini pertama kali diperkenalkan oleh Paul Werbos pada 1974, kemudian dikemukakan kembali oleh David Parker pada 1982 selanjutnya dipopulerkan oleh Rumelhart dan McClelland pada 1986. Aturan pelatihan jaringan *backpropagation* terdiri dari 2 tahapan yaitu *feedforward* dan *backward propagation*. Pada jaringan diberikan sekumpulan contoh pelatihan yang disebut sebagai set pelatihan. Set pelatihan ini terdiri dari vektor input dan vektor output yang akan menjadi target pelatihannya. Keluaran dari jaringan berupa sebuah vektor output aktual.

Siklus setiap perubahan bobot (*epoch*) dilakukan pada setiap set pelatihan hingga kondisi berhenti tercapai. Algoritma *backpropagation* terdiri dari 3 tahapan (Puspitaningrum, 2006) yaitu:

1. Tahapan umpan maju (*feedforward*)
2. Tahapan umpan mundur (*backward propagation*)
3. Tahap meng-*update* bobot dan bias

3. Metode Penelitian

3.1. Analisa Kebutuhan Fungsional dan Non-fungsional Sistem

Kebutuhan fungsional ialah kebutuhan akan proses yang dapat dilakukan oleh sistem. Berdasarkan penelitian ini, kebutuhan fungsional sistem tersebut antara lain sebagai berikut:

1. Sistem dapat melakukan input citra tomat melalui direktori dan melalui pengambilan langsung menggunakan *webcam* yang dihubungkan ke PC.
2. Sistem dapat menampilkan nilai histogram RGB dari citra tomat.
3. Sistem dapat menentukan nilai minimum RGB tomat berdasarkan pengolahan citra.

4. Sistem dapat melakukan pemotongan atau *cropping* pada citra asli sesuai dengan daerah pemotongan yang diinginkan.
5. Sistem dapat menghasilkan klasifikasi dan prediksi kematangan tomat berdasarkan data *training* dengan perhitungan metode *backpropagation*.

Kebutuhan non-fungsional ialah kebutuhan yang berisi proses-proses yang akan mempengaruhi sistem. Berdasarkan penelitian ini, proses-proses tersebut antara lain sebagai berikut:

1. Faktor cahaya yang diberikan.
2. Faktor latar pengambilan citra tomat.

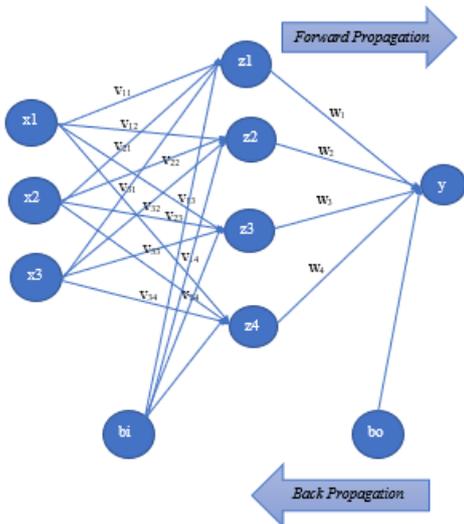
3.2 Tahapan Penelitian



Gambar 3.1 Tahapan Penelitian

3.3 Rancangan Arsitektur *Backpropagation*

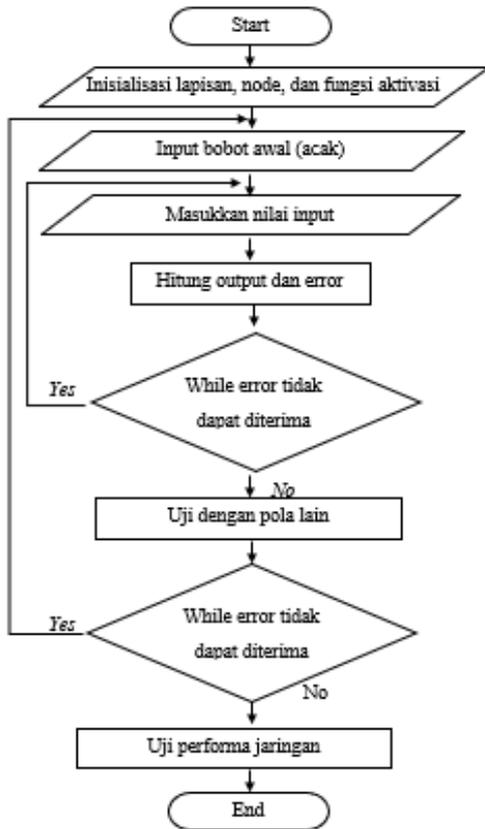
Langkah pertama yang dilakukan dalam perancangan penelitian ini dimulai dari perancangan arsitektur *backpropagation*, seperti pada gambar 3.2. Proses tersebut terdiri dari 3 unit *input layer*, 4 node *hidden layer*, dan 3 unit *output layer*. Pada bagian *input layer* (x) berisi nilai RGB dari tomat yang telah didapatkan dari hasil pengolahan citra.



Gambar 3. 2 Arsitektur Backpropagation

3.4 Rancangan Proses Jaringan Saraf Tiruan

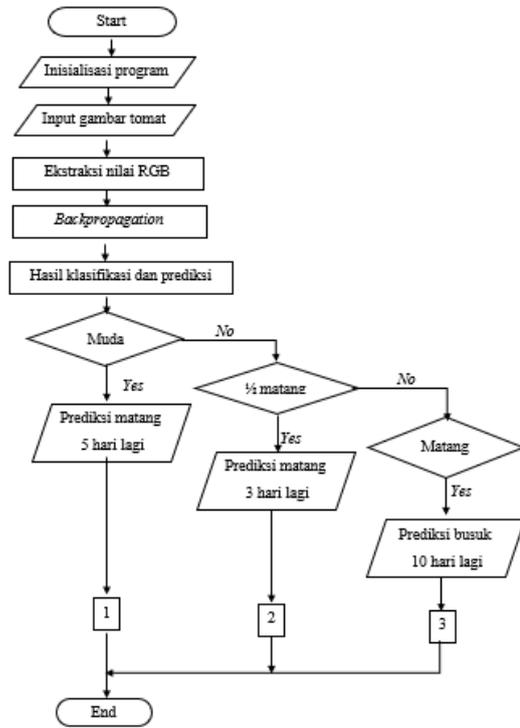
Flowchart jaringan saraf tiruan ditunjukkan pada gambar 3.4 berikut.



Gambar 3. 3 Flowchart Jaringan Saraf Tiruan

3.5 Flowchart Pengujian

Apabila pelatihan jaringan saraf tiruan telah didapatkan dan sistem yang dibangun telah selesai, maka tahap selanjutnya ialah melakukan pengujian terhadap data *training* dan data *testing*. Semua data dihitung nilai kebenaran dan kesalahan. Berdasarkan nilai tersebut didapatkan hasil tingkat akurasi. Flowchart untuk tahapan pengujian ditunjukkan pada gambar 3.4 berikut



Gambar 3. 4 Flowchart Pengujian

4. Implementasi

Pada penelitian ini dibuat sistem untuk mengetahui klasifikasi kematangan tomat dan prediksi dengan metode Jaringan Saraf Tiruan. Objek tomat diambil dari kelompok tani penghasil tomat di Alahan Panjang dapat dilihat pada gambar 4.1.



Gambar 4. 1 Kebun Tomat di Alahan Panjang

Data-data citra tomat yang sudah dipetik dengan menggunakan kamera *smartphone* dengan latar belakang solid berwarna putih menggunakan alat berupa mini *photo studio* seperti pada gambar 4.2. Proses pengambilan gambar tomat disesuaikan dengan pencahayaan yang sama.



Gambar 4. 2 Pengambilan Citra Tomat

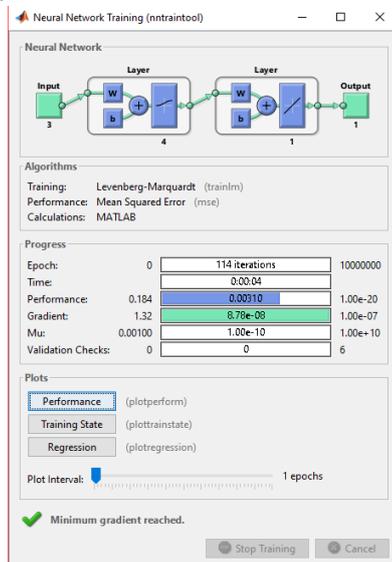
Total data yang didapatkan sebanyak 60 buah citra tomat. Pada data *training* terdapat total 30 citra tomat. Masing-masing kategori terdiri dari 10 citra. Sedangkan pada data *testing* terdapat 30 citra tomat.

4.1 Implementasi Pelatihan Data Latih

Data latih dikumpulkan dan dilatih menggunakan aplikasi MATLAB dengan metode jaringan saraf tiruan agar mendapatkan nilai bobot dan bias dari data tersebut. Gambar tomat sebagai data latih dicari terlebih dahulu nilai minimum RGB pada masing-masing citra.

Inisialisasi jaringan adalah hal pertama yang perlu dilakukan untuk menghitung *backpropagation* menggunakan Matlab. Perintah yang dipakai untuk membentuk jaringan adalah **newff**. Fungsi aktivasi yang digunakan diinput dari unit input ke layer tersembunyi adalah fungsi aktivasi sigmoid biner (logsig), sedangkan fungsi aktivasi dari layer tersembunyi ke output ialah fungsi identitas (purelin).

Setelah program tersebut dijalankan, maka bentuk jaringan pelatihan akan ditampilkan beserta atribut dan hasil dari *training* yang dilakukan.



Gambar 4. 3 Bentuk Jaringan Pelatihan

Hasil perhitungan nilai bobot dan bias dengan menggunakan MATLAB disimpan ke *excel*.

Nilai bobot *Input Weight (IW)* dan *Layer Weight (LW)* dan *Input Bias* dan *Layer Bias (IB dan LB)* dari pelatihan ini adalah:

	v1	v2	v3
IW =	0.267801589	0.120305327	0.280376744
	7.309172349	-7.012592424	1.510508462
	2.438326094	-4.397011898	1.942057488
	-	1.419873875	1.022442117
	1.697743454		

				65
LW =	1.579075076	-	0.278303465	-
		1.690360577		0.3893544414

IB =	0.001888
------	----------

	0.407338
	0.027037
	0.74368

LB =	0.666315622
------	-------------

Setelah memperoleh hasil pelatihan data latih dengan menggunakan MATLAB, selanjutnya menguji hasil tersebut dengan menggunakan rumus *backpropagation*. Nilai output yang didapat berdasarkan metode *backpropagation*. Prosedur pengujian aplikasinya adalah sebagai berikut:

Langkah 0 : Inisialisasi bobot (algoritma pelatihan).

Langkah 1 : Untuk tiap vektor masukan, lakukan langkah 2-4.

Langkah 2 : Hitung nilai di neuron *hidden layer* serta fungsi aktivasi sigmoid biner. Misalnya kita menghitung nilai output untuk nilai input minimum RGB yaitu 87, 97, dan 34.

$$z_{in_j} = \sum_{i=1}^n x_i v_{ji} + b_i$$

$$j=1, \quad z_{in_1} = (x_1 v_{11}) + (x_2 v_{21}) + (x_3 v_{31}) + b_1$$

$$z_{in_1} = (87 * 0.26780158) + (97 * 0.12030532) + (34 * 0.28037674) + 0.00188848$$

$$z_{in_1} = 44.50116429 + 0.00188848$$

$$z_{in_1} = 44.50305277$$

$$j=2, \quad z_{in_2} = (x_1 v_{12}) + (x_2 v_{22}) + (x_3 v_{32}) + b_2$$

$$z_{in_2} = (87 * 7.30917234) + (97 * -7.012592424) + (34 * 1.51050846) + 0.407337765$$

$$z_{in_2} = 7.033816925 + 0.407337765$$

$$z_{in_2} = 7.441154691$$

$$j=3, \quad z_{in_3} = (x_1 v_{13}) + (x_2 v_{23}) + (x_3 v_{33}) + b_3$$

$$z_{in_3} = (87 * 2.43832609) + (97 * -4.39701189) + (34 * 1.94205748) + 0.0270365070$$

$$z_{in_3} = -148.3458293 + 0.0270365070$$

$$z_{in_3} = -148.3187928$$

$$j=4, \quad z_{in_4} = (x_1 v_{14}) + (x_2 v_{24}) + (x_3 v_{34}) + b_4$$

$$z_{in_4} = (87 * 2.43832609) + (97 * 1.4198738752) + (34 * 1.0224421165) + 0.7436802508$$

$$z_{in_4} = 24.78711737 + 0.0270365070$$

$$z_{in_4} = 25.53079762$$

Fungsi Aktivasi

$$z_j = f(z_{in_j}) = \frac{1}{1 + e^{-z_{in_j}}}$$

$$j=1, \quad z_1 = f(z_{in_1}) = \frac{1}{1 + e^{-44.50305277}} = 1$$

$$j=2, \quad z_2 = f(z_{in_2}) = \frac{1}{1 + e^{-7.441154691}} = 0.999414$$

$$j=3, \quad z_3 = f(z_{in_3}) = \frac{1}{1 + e^{-(-148.3187928)}} = 3.85e^{-}$$

$$z_4 = f(z_{in_4}) = \frac{1}{1 + e^{25.53079762}} = 1$$

Langkah 5 :

Masing-masing unit output (y_k , $k=1,2,3,\dots,n$)

dikalikan dengan bobot dan dijumlahkan serta ditambahkan dengan indeks bias outputnya.

$$y_{in_k} = \sum_{j=1}^p z_j w_j + b_o$$

$$y_{in_1} = (z_1 w_1) + (z_2 w_2) + (z_3 w_3) + (z_4 w_4) + 0.666315622$$

$$y_{in_1} = (1 * 1.579075076) + (0.999414 * -1.690360577) + (3.85e - 65 * 0.2783034652) + (1 * -0.3893544413) + 0.666315622$$

$$y_{in_1} = 0.16667$$

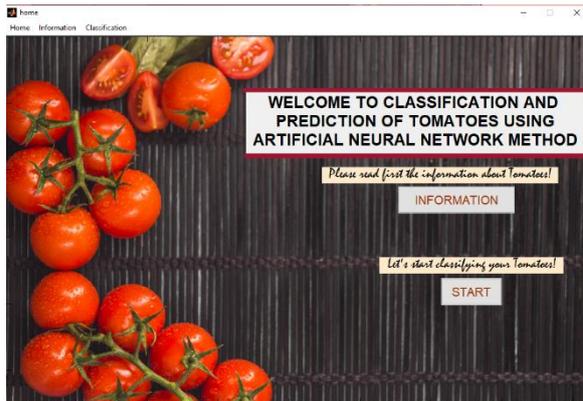
$$y_k = f(y_{in_k}) = y_{in_k} = 0.16667$$

Langkah 6 : Sesuaikan nilai output pada masing-masing kategori

Pada pemrograman sistem ini, nilai keluaran berupa y_k yang telah diperoleh akan menjadi parameter untuk kondisi penentuan tingkat kematangan dari tomat. Berdasarkan nilai y_k tersebut didapatkan tingkat kematangan tomat kategori belum matang bernilai rentang 0.166667, setengah matang bernilai 0.5000, dan matang bernilai 0.8333.

4.2 Implementasi User Interface

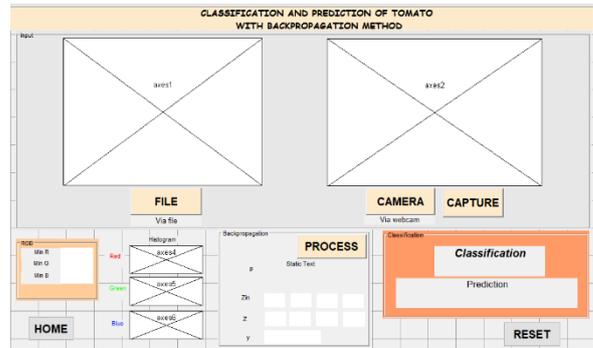
Aplikasi ini ditampilkan dalam bentuk *Graphical User Interface* (GUI). Sistem ini menggunakan tiga halaman, yaitu halaman *home* seperti pada gambar 4.4, *information* pada gambar 4.5, dan *classification* pada gambar 4.6.



Gambar 4. 4 Tampilan Halaman Home

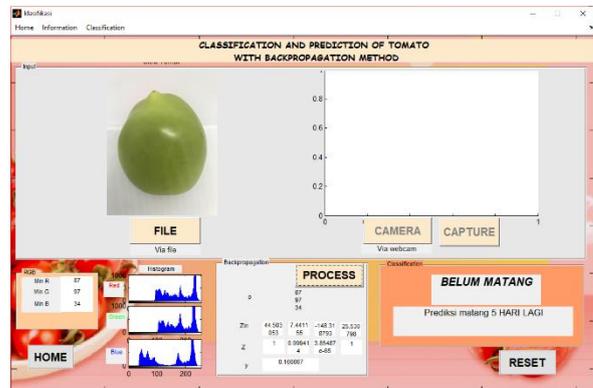


Gambar 4. 5 Tampilan Halaman Information

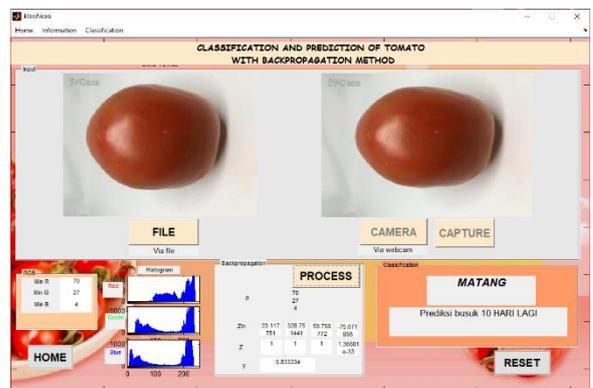


Gambar 4. 6 Tampilan Halaman Classification

Apabila aplikasi tersebut dijalankan, maka akan menampilkan hasil klasifikasi dan prediksi citra seperti pada gambar 4.7 jika melalui direktori citra tomat yang telah ada. Proses aplikasi ini dimulai dari pengambilan gambar tomat dapat melalui direktori yang telah ada maupun melalui *capture* langsung dengan menggunakan *webcam* seperti pada gambar 4.8. Apabila melalui direktori, maka *user* dapat menekan tombol *file* dan memilih citra tomat yang akan diklasifikasikan sehingga citra tersebut dapat ditampilkan. Kemudian tombol *process* akan menampilkan hasil dari ekstraksi citra berupa nilai minimum RGB dan perhitungan metode *backpropagation*.



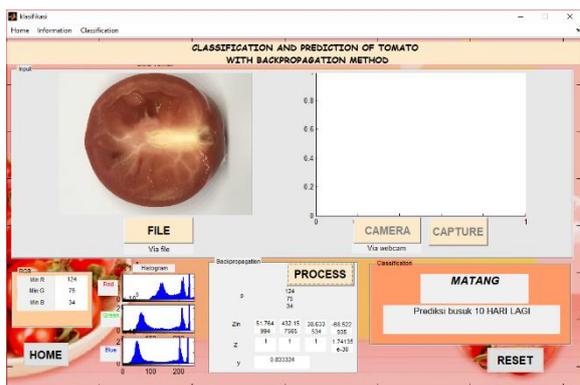
Gambar 4. 7 Input Citra melalui Direktori



Gambar 4. 8 Input Citra melalui Webcam

Input citra tomat juga diuji dengan menggunakan citra penampang tomat yang dipotong melintang. Hal ini untuk mengetahui kondisi kematangan pada tomat bagian dalam. Contoh klasifikasi kematangan tomat dengan

penampang melintang dapat dilihat pada gambar 4.9.



Gambar 4. 9 Pengujian Penampang Tomat

4.3 Pengujian

Pengujian dilakukan dengan cara menguji masing-masing data *training* dan data *testing* dengan menggunakan sistem yang telah dibuat. Pengujian sistem ini menerapkan *black box test*, yaitu pengujian hanya mengamati hasil eksekusi program melalui data uji pada sistem dan memeriksa fungsi-fungsi yang ada pada sistem atau aplikasi tersebut.

1. Pengujian Data Training

Tabel 4.1 menunjukkan data hasil pengujian terhadap data *training*.

Tabel 4. 1 Pengujian Data Training

No	Gambar	Nilai Min RGB			Hasil Hitung	Klasi fikasi	Kebe naran
		R	G	B			
1		87	97	34	0.166	Belum Matang	Benar
2		59	63	19	0.165	Belum Matang	Benar
3		61	64	27	0.165	Belum Matang	Benar
4		82	83	32	0.165	Belum Matang	Benar
5		76	77	28	0.165	Belum Matang	Benar
6		88	90	33	0.165	Belum Matang	Benar
7		73	74	22	0.17	Belum Matang	Benar
8		68	72	38	0.165	Belum Matang	Benar
9		57	60	15	0.169	Belum Matang	Benar
10		68	71	36	0.165	Belum Matang	Benar
11		75	75	18	0.494	Setengah Matang	Benar
12		67	74	20	0.500	Setengah Matang	Benar
13		89	62	17	0.555	Setengah Matang	Benar

14		92	78	28	0.555	Setengah Matang	Benar
15		84	80	26	0.494	Setengah Matang	Benar
16		73	65	16	0.555	Setengah Matang	Benar
17		81	79	22	0.512	Setengah Matang	Benar
18		93	85	21	0.555	Setengah Matang	Benar
19		72	67	24	0.500	Setengah Matang	Benar
20		81	61	23	0.555	Setengah Matang	Benar
21		59	28	9	0.833	Matang	Benar
22		87	54	19	0.833	Matang	Benar
23		65	33	9	0.833	Matang	Benar
24		74	37	14	0.833	Matang	Benar
25		68	59	19	0.555	Setengah Matang	Salah
26		61	28	12	0.833	Matang	Benar
27		59	30	14	0.833	Matang	Benar
28		66	31	14	0.833	Matang	Benar
29		61	31	10	0.833	Matang	Benar
30		57	27	10	0.833	Matang	Benar

$$Akurasi = \frac{\text{jumlah data benar}}{\text{jumlah seluruh data}} \times 100\%$$

$$Akurasi = \frac{29}{30} \times 100\% = 96,67\%$$

2. Pengujian Data Testing

Pengujian Data Testing dapat dilihat pada tabel 4.2.

Tabel 4. 2 Pengujian Data Testing

No	Gambar	Nilai Min RGB			Hasil hitung	Klasi fikasi	Kebe naran
		R	G	B			
1		60	64	17	0.162	Belum Matang	Benar
2		56	58	13	0.248	Belum Matang	Benar
3		60	68	16	1.856	Matang	Salah
4		14	15	17	0.380	Setengah Matang	Salah
5		43	46	11	0.171	Belum Matang	Benar
6		38	40	11	0.171	Belum Matang	Benar
7		59	61	13	0.314	Belum Matang	Benar
8		53	55	24	0.165	Belum Matang	Benar

9		57	59	22	0.165	Belum Matang	Benar
10		45	47	25	0.165	Belum Matang	Benar
11		38	26	10	0.582	Setengah Matang	Benar
12		67	52	8	0.555	Setengah Matang	Benar
13		67	63	11	0.555	Setengah Matang	Benar
14		42	38	12	0.550	Setengah Matang	Benar
15		64	53	9	0.555	Setengah Matang	Benar
16		67	59	13	0.555	Setengah Matang	Benar
17		66	58	14	0.555	Setengah Matang	Benar
18		39	25	4	0.555	Setengah Matang	Benar
19		60	41	18	0.758	Matang	Salah
20		71	68	20	0.532	Setengah Matang	Benar
21		59	25	5	0.833	Matang	Benar
22		58	26	7	0.833	Matang	Benar
23		47	22	2	0.833	Matang	Benar
24		55	23	8	0.833	Matang	Benar
25		63	24	8	0.833	Matang	Benar
26		46	21	5	0.833	Matang	Benar
27		53	29	11	0.833	Matang	Benar
28		51	26	5	0.833	Matang	Benar
29		51	25	8	0.833	Matang	Benar
30		46	13	0	0.833	Matang	Benar

$$Akurasi = \frac{27}{30} \times 100\% = 90\%$$

Berdasarkan hasil pengujian dari klasifikasi tingkat kematangan tomat menggunakan metode Jaringan Saraf Tiruan *Backpropagation*, didapatkan hasil akurasi yang bagus. Nilai akurasi pengujian terhadap data *training* ialah sebanyak 96,7%, nilai akurasi pengujian terhadap data *testing* ialah sebanyak 90%, dan nilai akurasi terhadap pengujian data penampang melintang tomat ialah sebanyak 80%. Sehingga metode yang digunakan sesuai dengan hasil yang diharapkan. Pada beberapa data, terdapat kesalahan dalam menghasilkan klasifikasi tingkat kematangan. Hal ini terjadi karena

kesalahan sistem dalam membaca nilai RGB yang diinputkan. Pada pengujian data *training* terdapat kesalahan satu data. Sedangkan pada pengujian data *testing*, terdapat kesalahan klasifikasi sebanyak 3 buah data. Pada data tomat kategori matang menghasilkan klasifikasi tingkat kematangan matang. Hal ini disebabkan karena terdapat kemiripan nilai RGB pada data matang dengan setengah matang.

5. Kesimpulan dan Saran

Dari hasil penelitian tentang pemanfaatan *image mining* untuk klasifikasi dan prediksi tingkat kematangan tomat menggunakan metode Jaringan Saraf Tiruan, dapat diambil kesimpulan sebagai berikut:

1. Berdasarkan penelitian yang dilakukan terhadap klasifikasi dan prediksi kematangan tomat, metode jaringan saraf tiruan algoritma *backpropagation* dengan struktur 3 *neuron input layer*, 4 *neuron hidden layer* dan 1 *neuron output layer* mampu melakukan pengklasifikasian berdasarkan unsur warna RGB dengan memanfaatkan *image mining*.

2. Berdasarkan penelitian yang dilakukan terhadap klasifikasi dan prediksi kematangan tomat menghasilkan tingkat akurasi pada pengujian menggunakan data *training* diperoleh hasil sebesar 96,7% sedangkan pengujian menggunakan data *testing* diperoleh hasil sebesar 90%. Sedangkan pada pengujian terhadap data penampang tomat diperoleh hasil sebesar 80%. Hasil keakuratan tergantung pada kualitas gambar yang dihasilkan.

3. Perangkat lunak yang dibangun mampu mengelompokkan kematangan tomat yaitu belum matang, setengah matang, dan matang.

Saran

Berdasarkan kesimpulan hasil penelitian, disarankan untuk metode Jaringan Saraf Tiruan yang diterapkan menggunakan lebih banyak *hidden layer* agar mendapatkan keakuratan yang lebih baik. Selain itu, aplikasi dapat dikembangkan berupa aplikasi *android* atau *web* sebagai aplikasi yang dipakai langsung oleh masyarakat. Kualitas citra yang dihasilkan juga dapat ditingkatkan dengan cara segmentasi citra dan metode lainnya.

DAFTAR PUSTAKA

- [1] Prakash, A. (2001). Antioxidant Activity. *Medallion Laboratories Analytical*, vol. 19, No.2.
- [2] Sandhiya, K., Vidhya, M., Shivaranjani, M., & Saranya, S. (2017). Smart Fruit Classification using Neural Networks. *International Journal of Trend in Scientific Research and Development (IJTSRD)*, Vol.2 Page: 1303.

- [3] Prabha D., S., & SateheeshKumar, J. (2012). A Study on Image Processing Methods for Fruit Classification. *Research Gate*, 403.
- [4] Turban, E. (2005). *Decision Support Systems and Intelligent Systems*. Yogyakarta: Andi.
- [5] Zhang, J., Hsu, W., & Lee, M. (2002). Image Mining: Trends and Developments. *Journal of Intelligent Information Systems*, 7-23
- [6] Silvana, M., & Kurnia, R. (2015). Skin and Clothes Matching Seeded by Color System Selection. *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol.14 hal. 509.