

SKRIPSI

**IDENTIFIKASI KEMATANGAN MENTIMUN
BERDASARKAN TEKSTUR MENGGUNAKAN *SELF
ORGANIZING MAPS***



FARIZI MAKRUF

16.0504.0034

**PROGRAM STUDI INFORMATIKA S1
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAGELANG
2022**

HALAMAN PENEGASAN

Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Farizi Makruf

NPM :16.0504.0034

Disusun Sebagai Salah Satu Syarat Memperoleh Gelar Komputer (S.Kom) Program Studi Teknik Informatika Jenjang Strata (S-1) Fakultas Teknik Universitas Muhammadiyah Magelang

Magelang, 15 Februari 2022



Farizi Makruf
16.0504.0034

HALAMAN PENGESAHAN

HALAMAN PENGESAHAN
SKRIPSI
IDENTIFIKASI KEMATANGAN MENTIMUN BERDASARKAN
TEKSTUR MENGGUNAKAN *SELF ORGANIZING MAPS*

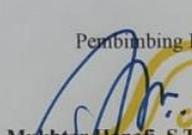
Dipersiapkan dan disusun oleh :
FARIZI MAKRUF
NPM. 16.0504.0034

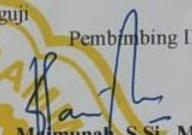
Telah dipertahankan di depan Dewan Penguji
Pada Tanggal 15 Februari 2022

Susunan Dewan Penguji

Pembimbing I

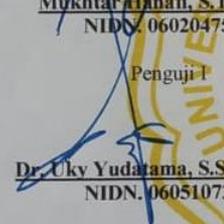
Pembimbing II


Mukhtar Hanafi, S.T., M.Cs.
NIDN. 0602047502


Maimunah, S.Si., M.Kom.
NIDN. 0612117702

Penguji I

Penguji II


Dr. Uky Yudatama, S.Si., M.Kom.
NIDN. 0605107201


Endah Ratna Arumi, M.Cs.
NIDN. 0601129001

Skripsi ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana Komputer
Tanggal, 15 Februari 2022

Dekan




Yun Arifatul Fatimah, MT., Ph.D.
NIK. 987408139

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini :

Nama : Farizi Makruf

NPM : 16.0504.0034

Program Studi : Teknik Informatika S1

Fakultas : Teknik

Alamat : Krajan 1, RT : 05, RW : 02 Grabag

Judul Skripsi : IDENTIFIKASI KEMATANGAN MENTIMUN
BERDASARKAN TEKSTUR MENGGUNAKAN SELF ORGANIZING
MAPS.

Dengan ini menyatakan bahwa skripsi ini merupakan hasil karya saya sendiri dan bukan merupakan plagiat dari hasil karya orang lain. Dan apabila terbukti karya ini plagiat, saya bersedia menerima sanksi administrasi maupun sanksi apapun.

Demikian surat pernyataan ini saya buat dengan penuh kesadaran dan sebenarnya serta penuh tanggung jawab.

Magelang 15 Februari 2022



Farizi Makruf
16.0504.0034

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

LEMBAR PERNYATAAN PERSETUJUAN UNGGAH REPOSITORI UNTUK KEPENTINGAN AKADEMIS

Yang bertanda tangan di bawah ini, saya :

Nama : Farizi Makruf
NPM : 16.0504.0034
Fakultas/ Jurusan : Teknik/ Teknik Informatika S1
E-mail address : farizimakruf.fm@gmail.com

Menyatakan bahwa demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Perpustakaan UNIMMA **Hak Bebas Royalti Non Eksklusif (Non-Royalty- Free-Right)** atas karya ilmiah yang berjudul :

Identifikasi Kematangan Mentimun Berdasarkan Tekstur Menggunakan Self Organizing Maps beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non Eksklusif ini Perpustakaan UNIMMA berhak menyimpan, mengalih media / memformatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir tersebut selama tetap mencantumkan nama saya sebagai penulis / pencipta dan sebagai pemilik hak cipta.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Perpustakaan UNIMMA, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini dibuat dengan sesungguhnya.

Dibuat di : Magelang

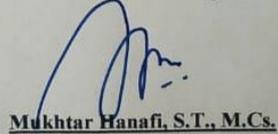
Pada tanggal : 15 Februari 2022

Penulis,

Farizi Makruf

NPM. 16.0504.0034

Mengetahui,
Dosen Pembimbing


Mukhtar Hanafi, S.T., M.Cs.

NIDN. 0602047502

KATA PENGANTAR

Alhamdulillah rabbil'alamin, puji syukur dipanjatkan ke hadirat Allah SWT, karena atas berkat nikmat dan karunia-Nya, Tugas Akhir/Skripsi ini dapat diselesaikan. Penyusunan Tugas Akhir/Skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Ahli Madya atau Sarjana Teknik Program Studi Teknik Informatika S1 Fakultas Teknik Universitas Muhammadiyah Magelang.

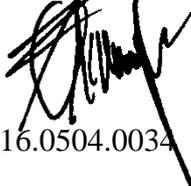
Penyelesaian Tugas Akhir/Skripsi ini banyak memperoleh bantuan dan bimbingan dari berbagai pihak. Oleh karena itu, diucapkan terima kasih kepada :

1. M.Kolyubi dan Mustichatun selaku orang tua, Muhammad Iwan Al Khasni, S.IP., Mutiya Khairunisak, A.Md.Kep., dan Sovian Rizqiawan, S.Kom., Catur Suhartati, S.Kep., Ns. selaku kakak yang saya sayangi dan Haidar Alwan Al Khasni dan Rafaeyza Kael Ardafa selaku keponakan yang saya sayangi dan saya cintai, segenap keluarga yang telah memberikan bantuan dukungan material dan moral;
2. Dr. Lilik Andriyani, S.E., M.Si selaku Rektor Universitas Muhammadiyah Magelang.
3. Yun Arifatul Fatimah, S.T., M.T., Ph.D. selaku Dekan Fakultas Teknik Universitas Muhammadiyah Magelang
4. Endah Ratna Arumi, M.Cs selaku Ketua Program Studi Teknik Informatika S1 Universitas Muhammadiyah Magelang.
5. Mukhtar Hanafi, S.T., M.Cs. selaku dosen pembimbing utama dan Maimunah, S.Si., M.Kom. selaku dosen pembimbing pendamping yang telah memberikan semangat, menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penyusunan skripsi ini;
6. A.Sakir Maksun, S.T. yang telah banyak membantu menyediakan waktu, tenaga, dalam usaha memperoleh data yang diperlukan;
7. Para sahabat dan juga segelas jahe kopi hitam yang telah banyak membantu dalam penyelesaian skripsi ini.

Akhir kata, semoga Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu dan semoga Tugas Akhir/Skripsi ini membawa manfaat bagi pengembangan ilmu.

Magelang, 15 Februari 2022

Farizi Makruf



16.0504.0034

DAFTAR ISI

SKRIPSI	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PENGESAHAN	iii
PERNYATAAN KEASLIAN	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS.....	v
KATA PENGANTAR.....	vi
DAFTAR ISI	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	xii
INTISARI.....	xiii
ABSTRACT	xiv
BAB 1 PENDAHULUAN.....	17
1.1 Latar Belakang Permasalahan	17
1.2 Rumusan Masalah	19
1.3 Tujuan Penelitian.....	19
1.4 Manfaat Penelitian.....	19
BAB 2 TINJAUAN PUSTAKA.....	20
2.1 Penelitian Relevan	20
2.2 Landasan Teori	24
2.2.1 Pengertian Citra.....	24
2.2.2 Mentimun	26
2.2.3 Pengolahan Citra Digital	27
2.2.4 Digitalisasi Citra.....	27
2.2.5 Konversi Citra RGB Menjadi <i>Grayscale</i>	28
2.2.6 Transformasi <i>Wavelet</i>	29
2.2.7 Transformasi <i>Wavelet</i> Diskrit.....	30
2.2.8 Filter <i>Wavelet Daubechies</i>	32

2.2.9	Paket <i>Wavelet</i>	33
2.2.10	Tekstur (<i>Texture</i>)	34
2.2.11	Energi	35
2.2.12	Entropi	36
2.2.13	Jaringan Saraf Tiruan	36
2.2.14	Komponen Jaringan Saraf Tiruan.....	37
2.2.15	Arsitektur Jaringan Syaraf Tiruan	39
2.2.16	Proses Pembelajaran.....	40
2.2.17	Jaringan Saraf Tiruan <i>Self Organizing Map (SOM)</i>	41
2.2.18	Algoritma Kohonen <i>Self Organizing Map (SOM)</i>	42
2.2.19	<i>Confusion Matrix</i>	43
BAB 3 METODOLOGI PENELITIAN.....		45
3.1	Prosedur Penelitian.....	45
3.2	Analisa Sistem	47
3.2.1	Analisa Sistem Yang Berjalan.....	47
3.2.2	Analisa Sistem yang Diusulkan.....	48
3.3	Perancangan Sistem.....	51
3.3.1	Analisis kebutuhan data.....	51
3.3.2	Perancangan object oriented/procedural.....	53
3.3.3	Perancangan data/arsitektur.....	60
3.3.4	Perancangan antar muka.....	61
BAB 5 KESIMPULAN DAN SARAN.....		97
5.1	Kesimpulan.....	97
5.2	Saran.....	97
DAFTAR PUSTAKA.....		98

DAFTAR TABEL

Tabel 2.1 Bentuk <i>Confusion matrix</i>	43
Tabel 3.1 <i>Cropping</i> Citra Asli dan Tekstur <i>Grayscale</i>	52

DAFTAR GAMBAR

Gambar 2.1 Matriks Citra	25
Gambar 2.2 Citra Biner	25
Gambar 2.3 <i>Grayscale</i>	25
Gambar 2.4 Citra warna dengan komposisi RGB	26
Gambar 2.5 Fungsi Digitalisasi Citra Keabuan	28
Gambar 2.6 Fungsi Citra Digital.....	28
Gambar 2.7 <i>Index</i> Citra Digital.....	28
Gambar 2.8 Transformasi <i>Wavelet</i> Diskrit Dua Dimensi dengan <i>Level</i> Dekomposisi Satu	30
Gambar 2.9 Dekomposisi <i>Level</i> 1	31
Gambar 2.10 <i>Wavelet Tree</i>	31
Gambar 2.11 Tranformasi Paket <i>Wavelete</i>	33
Gambar 2.12 Paket Detail	34
Gambar 2.13 Model Neuron dengan Banyak Lapisan	37
Gambar 2.14 Susunan Neuron Jaringan Syaraf Tiruan.....	38
Gambar 2.15 Jaringan Saraf dengan Lapisan Tunggal	39
Gambar 2.16 Jaringan Saraf dengan Banyak Lapisan	40
Gambar 2.17 Jaringan Saraf dengan Lapisan Kompetitif	40
Gambar 2.18 Perhitungan <i>Confusion Matrix</i>	44
Gambar 3.1 Alur Penelitian.....	45
Gambar 3.2 Alur Penelitian Sistem Yang Diusulkan	49
Gambar 3.3 <i>Flowchart</i> Proses Perhitungan Data.....	53
Gambar 3.4 <i>Use Case Diagram</i>	54
Gambar 3.5 <i>Activity Diagram</i> Pelatihan.....	54
Gambar 3.6 <i>Activity Diagram</i> Pengujian.....	55
Gambar 3.7 <i>Activity Diagram</i> Tentang.....	56
Gambar 3.8 <i>Activity Diagram</i> Keluar.....	57
Gambar 3.9 <i>Sequence Diagram</i> Pelatihan	58
Gambar 3.10 <i>Sequence Diagram</i> Pengujian.....	59

Gambar 3.11 <i>Sequence</i> Diagram Tentang.....	59
Gambar 3.12 <i>Sequence</i> Diagram Keluar	60
Gambar 3.13 Perancangan Data.....	60
Gambar 3.14 Rancangan <i>Form</i> Utama.....	61
Gambar 3.15 Rancangan <i>Form</i> Pelatihan	62
Gambar 3.16 Rancangan <i>Form</i> Pengujian	63
Gambar 3.17 Rancangan <i>Form</i> Tentang	63

DAFTAR LAMPIRAN

INTISARI

IDENTIFIKASI KEMATANGAN MENTIMUN BERDASARKAN TEKSTUR MENGGUNAKAN *SELF ORGANIZING MAPS*

Nama : Farizi Makruf

Dosen Pembimbing : 1. Mukhtar Hanafi, S.T., M.Cs.

2. Maimunah, S.Si., M.Kom.

Mentimun (*Cucumis sativus L*) merupakan buah yang dapat dikonsumsi dan dapat diolah lebih lanjut sebagai bahan baku pada industri kecantikan. Kemiripan tekstur mentimun matang, setengah matang dan mentah mengakibatkan orang kesulitan dalam mengidentifikasi mentimun matang dari segi ciri tekstur kulit dan seseorang penilaian seseorang yang bersifat subyektif terhadap tingkat kematangan buah mentimun menyebabkan penilaian tingkat kematangan mentimun berbeda dari satu penilai dengan penilai yang lainnya. Dari permasalahan tersebut, sehingga dilakukan pengolahan citra (*Image Processing*) untuk mendeteksi kematangan mentimun berdasarkan tekstur menggunakan jaringan syaraf tiruan *Self Organizing Maps (SOM)*. Tujuan dari penelitian ini membuat sebuah sistem yang dapat mengenali tingkat kematangan buah mentimun berdasarkan tekstur dan mengetahui tingkat keakuratan dengan menggunakan jaringan syaraf tiruan *Self Organizing Maps (SOM)* untuk pendeteksian kematangan pada citra buah mentimun. Tahapan pengolahan citra yang digunakan yaitu *cropping*, *grayscale*, *discrete wavelet transforms (DWT)* dan nilai bobot *energy* dan *entropy*. Auisisi citra yang digunakan berjumlah 90 *sample* dataset tekstur citra buah mentimun yang digunakan, 72 *sample* sebagai data pelatihan (*training*) dengan pengamatan manual. 18 *sample* sebagai data pengujian (*testing*) dari 6 *sample* tekstur citra uji mentimun matang sistem menunjukkan *confusion matrix* (akurasi) yang diperoleh 100%, 6 tekstur citra uji mentimun setengah matang sistem menunjukkan akurasi yang diperoleh 66,67%, dan 6 tekstur citra uji mentimun setengah matang sistem menunjukkan akurasi yang diperoleh 100%, Dari hasil 18 *sample* citra uji, tingkat keberhasilan sistem dalam mengidentifikasi kematangan mentimun berdasarkan tekstur kulit buah sesuai pengamatan manual dari perhitungan nilai *energy* dan *entropy* memperoleh akurasi sebesar 88,89%.

Kata kunci : Kematangan Mentimun, Jaringan Syaraf Tiruan, Pengolahan Citra

ABSTRACT

IDENTIFICATION OF CUCUMBER MATURITY BASED ON TEXTURE USING SELF ORGANIZING MAPS

Name : Farizi Makruf

Advisory Dose : 1. Mukhtar Hanafi, S.T., M.Cs.
2. Maimunah, S.Si., M.Kom.

Cucumber (Cucumis sativus L) is a fruit that can be consumed and can be further processed as a raw material for the beauty industry. The similarity of the texture of ripe, half-ripe and raw cucumbers makes it difficult for people to identify ripe cucumbers in terms of skin texture characteristics and someone's subjective assessment of the maturity level of cucumbers causes the assessment of the maturity level of cucumbers to differ from one rater to another. From these problems, image processing was carried out to artificial neural network Self Organizing Maps (SOM). artificial neural network Self Organizing Maps (SOM) for detection of ripeness in cucumber images. The stages of image processing used are cropping, grayscale, discrete wavelet transforms (DWT) weights energy and energy. The image acquisition used is 90 samples of the cucumber fruit texture image dataset used, 72 samples as training data with manual observations. 18 samples as test data (testing) from 6 samples of the system's ripe cucumber test image texture showed a confusion matrix (accuracy) obtained 100%, 6 half-ripe cucumber test image textures system showed an accuracy of 66.67%, and 6 test image textures half-ripe cucumber system showed 100% accuracy. From the results of 18 sample test images, the system's success rate in identifying ripe cucumbers based on fruit skin texture according to manual observations from calculating energy and entropy obtained an accuracy of 88.89%.

Keywords : Cucumber Maturity, Artificial Neural Networks, Image Processing

BAB 1

PENDAHULUAN

1.1 Latar Belakang Permasalahan

Mentimun, timun, atau ketimun (*cucumis sativus L*) merupakan tumbuhan yang menghasilkan buah yang dapat dikonsumsi baik dalam kondisi segar maupun diolah lebih lanjut. Selain untuk bahan makanan, mentimun juga banyak digunakan sebagai bahan baku pada industri kecantikan. Mentimun dikelompokkan menjadi dua golongan, yaitu mentimun yang pada buahnya terdapat bintik-bintik di bagian pangkalnya dan mentimun yang buahnya halus (tidak berbintik-bintik). Mentimun yang berbintik-bintik dibedakan tiga macam, yaitu mentimun biasa (konsumsi), watong, dan wuku. Mentimun biasa ditandai dengan penampilan kulit buah yang tipis, lunak dan pada saat buah muda berwarna hijau dengan larik-larik putih kekuningan tetapi setelah buah masak warna luar buah berubah menjadi hijau pucat sampai putih. (Ardi Pujiyanta, Sovian Rizqiawan 2016)

Mentimun dikatakan matang dan siap panen adalah apabila umur panen 2-3 bulan atau 60-75 hari setelah penanaman ditandai dengan ciri-ciri yaitu setelah terjadi perubahan warna buah dan ukuran fisik buah yang bertambah panjang, maka buah tersebut bisa dipetik (dipanen). Salah satu cara meningkatkan nilai ekonomis mentimun adalah dengan melakukan sortasi sebelum buah dikirim atau dijual. Sortasi dilakukan untuk memisahkan buah yang belum matang, kurang baik bentuknya/bengkok, ataupun busuk dari buah yang baik. Sortasi kualitas diklasifikasikan sesuai dengan kebutuhan konsumen, dan sasaran pemasaran.

Umumnya, sortasi dilakukan dengan dua cara, yaitu manual (menggunakan indera manusia) dan mekanis (menggunakan mesin atau sistem). Sortasi yang dilakukan secara manual adalah sortasi yang berdasarkan pada persepsi manusia terhadap faktor komposisi tekstur kulit, warna ataupun kerusakan yang dimiliki buah para petani awam dan konsumen buah dalam mengenali tingkat kematangan buah mentimun secara langsung menjadi kurang efektif dan efisien. Industri pengolahan hasil pertanian dan perkebunan sangat berkembang pesat.

Kegiatan pasca panen erat kaitannya dengan mutu produk yang dihasilkan, yang pada akhirnya menentukan pula harga jual yang dapat diterima oleh petani. Salah satu tahap dalam proses pengolahan hasil perkebunan pasca panen adalah pemilahan produk berdasar kualitasnya yang dapat menjamin keseragaman mutu. Mutu buah mentimun ditentukan oleh berbagai parameter diantaranya adalah parameter tingkat usia tanam, bentuk dan ukuran fisik, serta kematangannya. (BPPSDMP, Kementerian Pertanian, 2008)

Berdasarkan deskripsi di atas pengolahan citra digital dapat mendeteksi tekstur kulit buah yang terdapat pada buah mentimun. Kulit buah mentimun bisa digunakan untuk acuan mengklasifikasikan tingkat kematangan.

Penelitian terdahulu (Kadek Dwi Diani Shinta Pratiwi, 2016) dengan judul "Identifikasi Kematangan Mentimun Berdasarkan Warna Kulit Buah Menggunakan Metode *K-Nearest Neighbor*". Pengujian dilakukan terhadap 60 citra uji terdiri dari 30 mentimun muda (10 kondisi pagi, 10 kondisi siang dan 10 kondisi sore) dan 30 mentimun tua (10 kondisi pagi, 10 kondisi siang dan 10 kondisi sore). Pengujian ini meliputi pengujian perparameter ciri, kombinasi 2 parameter ciri, kombinasi 3 parameter ciri, kombinasi 4 parameter ciri, dan kombinasi 5 parameter ciri dengan tingkat keberhasilan rata-rata 90%.

Jaringan syaraf tiruan paradigma pemrosesan suatu informasi yang terinspirasi oleh sistem sel syaraf biologi, sama seperti otak manusia yang memproses suatu informasi. Jaringan syaraf tiruan dibentuk untuk memecahkan suatu masalah tertentu seperti pengenalan pola atau klasifikasi karena proses pembelajaran. Jaringan syaraf tiruan banyak diaplikasikan dalam kehidupan sekarang ini untuk pengenalan bentuk angka, pengenalan pola bentuk, pola tekstur dan masih banyak lagi. Algoritma dalam jaringan syaraf tiruan yang banyak digunakan untuk pengenalan pola/bentuk adalah algoritma *Learning Vector Quantization (LVQ)* dan *Self Organizing Map (SOM)*.

Hal inilah yang mendasari penelitian ini dilaksanakan sehingga memberi inspirasi untuk membangun pengolahan citra untuk mendeteksi kematangan buah

mentimun berdasarkan ciri tekstur kulit, maka topik yang di bahas dalam penelitian ini ialah “Identifikasi Kematangan Mentimun Berdasarkan Tekstur Menggunakan *Self Organizing Maps (SOM)*”.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah, maka rumusan masalahnya adalah bagaimana merancang dan mengimplementasikan pengolahan citra yang dapat digunakan untuk mengidentifikasi kematangan buah mentimun berdasarkan tekstur menggunakan metode *SOM (Self Organizing Map)*.

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah, maka tujuan penelitian yang akan dicapai adalah:

1. Membuat sebuah sistem yang dapat mengenali tingkat kematangan buah mentimun berdasarkan tekstur.
2. Mengetahui tingkat keakuratan model *Self Organizing Map (SOM)* untuk pendeteksian kematangan buah mentimun berdasarkan fitur tekstur kulit buah menggunakan *GUI (Graphical User Interface)* pada bahasa pemrograman *Matlab*.

1.4 Manfaat Penelitian

Berdasarkan tujuan penelitian yang telah disebutkan di atas, maka hasil penelitian ini dapat memberikan manfaat sebagai berikut:

1. Membuat perangkat lunak yang dapat mengenali tingkat kematangan pada buah mentimun berdasarkan tekstur dan ketepatan penggunaan jaringan syaraf tiruan metode *Self Organizing Map (SOM)*.
2. Sebagai media aplikasi ilmu pengetahuan dan pengembangan khususnya dalam bidang pertanian serta untuk menambah wawasan tentang pengolahan citra dan jaringan syaraf tiruan dalam mengenali pola kematangan buah mentimun.

BAB 2

TINJAUAN PUSTAKA

2.1 Penelitian Relevan

1. Imam Fathurrahman, Amri Muliawan Nur, dan Fathurrahman (2019)

Penelitian berjudul Identifikasi Kematangan Buah Mentimun Berbasis Citra Digital Menggunakan Jaringan Syaraf Tiruan *Backpropagation*. Penelitian ini bertujuan untuk mengidentifikasi kematangan buah mentimun berbasis citra digital menggunakan algoritma jaringan syaraf tiruan *backpropagation* berdasarkan tekstur kulit buah dan mengukur tingkat akurasi identifikasi kematangan buah mentimun.

Dari hasil eksperimen terhadap 212 sampel citra pada buah mentimun dalam identifikasi menggunakan jaringan syaraf tiruan *backpropagation* menghasilkan akurasi yang baik yaitu mencapai 89.6% secara keseluruhan baik matang maupun mentah.

2. Hasil Penelitian Kadek Dwi Diani Shinta Pratiwi , Bulkis Kanata, I Made Budi Suksmadana (2016)

Penelitian berjudul Identifikasi Kematangan Mentimun Berdasarkan Warna Kulit Buah Menggunakan Metode *K-Nearest Neighbor*. Penelitian ini bertujuan identifikasi kematangan mentimun berdasarkan warna kulit buah dengan melakukan pengolahan citra dari buah mentimun menjadi citra *RGB*. Citra *RGB* diolah untuk mendapatkan nilai statistik orde pertama. Data statistik tersebut akan disimpan sebagai database data latih dan data uji serta pengklasifikasiannya menggunakan *k-nearest neighbor*.

Berdasarkan pengujian yang telah dilakukan terhadap 60 sampel citra, diperoleh tingkat keberhasilan terbesar pada pengujian perparameter ciri *mean* citra *red* mencapai 90%, pada pengujian kombinasi 2 parameter ciri *mean* dan *entropy* citra *red* mencapai 100%, kombinasi 3 parameter ciri *mean*, *skewness* dan kurtosis citra *red* mencapai 90% dan kombinasi 4 parameter ciri *mean*, *skewness*, kurtosis dan *entropy* citra *red* mencapai 90%.

3. Hasil penelitian Bagus Hardiansyah (2018)

Penelitian Bagus Hardiansyah (2018) berjudul “Sistem Pakar Pengenalan Ekspresi Wajah Manusia Menggunakan Metode Kohonen *Self Organizing* dan

Principal Componen Analysis". Penelitian ini bertujuan untuk melakukan pengenalan ekspresi wajah menggunakan Jaringan Syaraf Tiruan Kohonen *Self Organizing Map* (K-SOM). Metode dalam penelitian yang digunakan oleh Bagus Hardiansyah adalah Kualitatif.

Berdasarkan hasil eksperimen yang telah dilakukan, maka diperoleh nilai *bmqhb* 0.8 dengan tingkat akurasi yang tinggi dan jumlah data pengujian terbaik 30 citra ekspresi wajah dengan kinerja 80.00%. Kesalahan dalam mengenali citra ekspresi wajah disebabkan ada beberapa noise sehingga hasil dari pengenalan citra ekspresi wajah tidak maksimal.

4. Hasil penelitian Didik Styawan (2018)

Penelitian Didik Styawan (2018) berjudul "Pengenalan Plat Nomor Kendaraan dengan Metode Kohonen *SOM*". Penelitian ini bertujuan untuk menganalisis seberapa besar pengaruh iterasi pada tahap pelatihan terhadap hasil pengenalan karakter pada citra plat nomor. Metode dalam penelitian yang digunakan oleh Didik Styawan adalah Kualitatif.

Berdasarkan hasil eksperimen yang telah dilakukan, bahwa metode Kohonen *SOM* dapat mengenali karakter pada citra digital. Dibandingkan dengan penghentian pada jumlah iterasi tertentu, hasil dari penghentian jika bobot mendekati sama pada tahap pelatihan relatif lebih tinggi.

Berdasarkan hasil analisis, penulis memberikan beberapa saran yaitu hasil pengenalan dengan metode yang digunakan penulis masih rendah, dan belum bisa diterapkan pada kehidupan sehari-hari karena masih ada kemungkinan besar terjadi kesalahan dalam pengenalan plat nomor sehingga diharapkan pada penelitian selanjutnya bisa menghasilkan akurasi yang lebih baik. Selain itu diharapkan bisa diterapkan pada kendaraan bermotor roda dua.

5. Hasil penelitian Shabrina Elha Putri (2018)

Penelitian Shabrina Elha Putri (2018) berjudul "Identifikasi Pola Enamel Gigi Menggunakan Metode *Discrete Wavelet Transform (DWT)* dan *Self Organizing Map (SOM)* untuk Aplikasi Forensik Kedokteran Gigi". Penelitian ini bertujuan untuk mempermudah proses identifikasi dan klasifikasi pola enamel gigi. Pembuatan aplikasi ini dimulai dengan pengambilan *sample* foto dari gigi *incisivus* (gigi seri) agar

diperoleh pola-pola enamel gigi dari setiap individu. Metode dalam penelitian yang digunakan oleh Shabrina Elha Putri adalah Kualitatif.

Berdasarkan hasil eksperimen yang telah dilakukan, maka diperoleh kesimpulan bahwa metode *DWT* cocok dipasangkan dengan metode klasifikasi *SOM* karena menghasilkan klasifikasi yang baik dengan akurasi sebesar 88.5% dengan waktu komputasi 30.6636s. Tingkat akurasi tersebut diperoleh antara lain dengan parameter *DWT* dekomposisi *level 3*, karena dekomposisi *level 3* sudah mampu menghasilkan vektor ciri lebih banyak dibandingkan dekomposisi *level* yang lain, jenis *wavelet coiflets1*, karena *coiflet1* dapat mengembalikan penskalaan filter sehingga menghasilkan vektor ciri yang baik, filter LL, karena filter LL berisi sebagian besar dari informasi citra dan klasifikasi *SOM* dengan parameter topologi *Randtop* karena neuron yang disusun secara acak sehingga neuron yang sudah didapatkan mudah untuk diplotkan dan jenis jarak *Boxdist* yang berfungsi untuk menemukan jarak antar neuron *layer* serta karena *boxdist* berbentuk persegi sehingga mudah diaplikasikan dengan topologi *randtop*.

6. Hasil penelitian David M. Wewo (2018)

Penelitian David M. Wewo (2018) berjudul “Aplikasi Penentuan Golongan Darah Manusia dengan Metode *Seed Region Growing* dan *Self Organizing Map*”. Penelitian ini bertujuan untuk mendapatkan ciri dari pola gumpalan darah yang telah ditetesi *reagen*. Metode dalam penelitian yang digunakan oleh David M. Wewo adalah Kualitatif.

Berdasarkan hasil eksperimen yang telah dilakukan, bahwa sistem dapat dengan baik mengidentifikasi golongan darah berdasarkan pelatihan yang telah dilakukan, dengan nilai presentase sebesar 100 %, hal ini disebabkan karena kemampuan metode *SOM* dalam mengidentifikasi citra kedalam kelas tertentu. Namun untuk proses ekstraksi ciri citra dengan memilih nilai pengambangan untuk mendapatkan ciri dengan baik, jika salah dalam memilih nilai pengambangan maka kemungkinan tidak mendapatkan hasil ciri yang baik. Dari hasil pengujian yang dilakukan terhadap data yang belum pernah dilatih terlihat bahwa sistem kurang dapat mengidentifikasi semua citra data uji dengan baik, dengan nilai presentase sebesar 83.33 %, hal ini disebabkan oleh beberapa hal yang telah dijelaskan di atas.

Berdasarkan hasil pengujian sistem yang telah dilakukan, maka diberikan beberapa saran untuk pengembangan sistem yaitu menggunakan data latih dan data uji yang lebih banyak dan menggunakan teknik untuk *cropping* citra yang lain agar mendapatkan daerah darah yang lebih akurat.

Berdasarkan keenam penelitian di atas, bermaksud untuk melakukan penelitian untuk mengetahui kematangan buah mentimun dengan menggunakan metode Kohonen *Self Organizing Map* (*K-SOM*). Adapun persamaan penelitian yang saya ajukan dengan keenam penelitian terdahulu adalah semuanya menggunakan konsep jaringan syaraf tiruan.

Sedangkan perbedaan penelitian ini dengan penelitian terdahulu yaitu:

1. Penelitian terdahulu menggunakan metode *Backproagation*, *K-Nearest Neighbor*, sedangkan penelitian ini menggunakan metode *Kohonen Self Organizing Maps*.
2. Jumlah klasifikasi buah mentimun, dimana penelitian terdahulu hanya terdapat dua klasifikasi (mentah dan matang), sedangkan penelitian ini terdapat tiga klasifikasi (matang, setengah matang, dan mentah).
3. Jumlah sampel yang digunakan pada penelitian terdahulu ada 212 buah dan 60 buah sampel citra, sedangkan pada penelitian ini menggunakan 90 sampel citra buah mentimun.
4. Variabel yang digunakan pada penelitian terdahulu diantaranya yaitu *mean*, *skewness*, kurtosis dan *entropy* sedangkan penelitian ini menggunakan *variable entropy* dan *energy*.

2.2 Landasan Teori

2.2.1 Pengertian Citra

Secara harfiah, citra (*image*) adalah gambar pada bidang dwimatra (dua dimensi). Ditinjau dari sudut pandang matematis, citra merupakan fungsi menerus (*continue*) dari intensitas cahaya pada bidang dwimatra. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh alat-alat optik, seperti mata pada manusia, kamera, pemindai (*scanner*), dan lain-lain sehingga bayangan objek dalam bentuk citra dapat terekam. Citra dapat dikelompokkan menjadi dua bagian yaitu citra diam (*still images*) dan citra bergerak (*moving images*). Citra diam adalah citra tunggal yang tidak bergerak. Sedangkan citra bergerak adalah rangkaian citra diam yang ditampilkan secara beruntun (sekuensial), sehingga memberi kesan pada mata sebagai gambar yang bergerak. Setiap citra didalam rangkaian itu disebut *frame*. Gambar-gambar yang tampak pada film layar lebar atau televisi pada hakekatnya terdiri dari ratusan sampai ribuan *frame*. (Aris Budi S, Suma'inna, Hata Maulana, 2016)

Citra digital adalah sebuah fungsi 2D, $f(x,y)$, yang merupakan fungsi intensitas cahaya, dimana nilai x dan y merupakan koordinat spasial dan nilai fungsi di setiap titik (x,y) merupakan tingkat keabuan citra pada titik tersebut.

Citra digital dinyatakan dengan sebuah matriks dimana baris dan kolomnya menyatakan suatu titik pada citra tersebut dan elemen matriksnya (yang disebut sebagai elemen gambar atau piksel) menyatakan tingkat keabuan pada titik tersebut. Matriks dari citra digital berukuran $N \times M$ (tinggi x lebar), dimana:

$$N = \text{jumlah baris } 0 < y = N - 1$$

$$M = \text{jumlah kolom } 0 = x = M - 1$$

$$L = \text{derajat keabuan } 0 = f(x,y) = L - 1$$

Berikut ini adalah gambaran matriks dari citra digital:

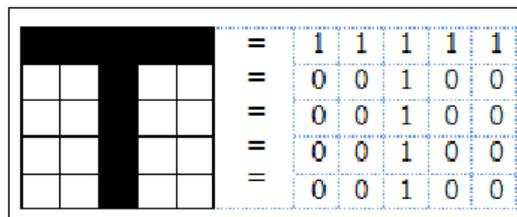
$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & f(1,1) & \dots & f(1,M-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

Gambar 2.1 Matriks Citra

Dimana indeks baris (x) dan indeks kolom (y) menyatakan suatu koordinat titik pada citra, sedangkan $f(x,y)$ merupakan intensitas (derajat keabuan) pada titik (x,y) . Berdasarkan jenisnya, citra digital dapat dibagi menjadi tiga (Aris Budi S, Suma'inna, Hata Maulana, 2016), yaitu:

I. Citra Biner (Monokrom)

Memiliki 2 buah warna, yaitu hitam dan putih. Warna hitam bernilai 1 dan warna putih bernilai 0. Untuk menyimpan kedua warna ini dibutuhkan 1 bit di memori. Contoh dari susunan piksel pada citra monokrom adalah sebagai berikut:



Gambar 2. 2 Citra Biner

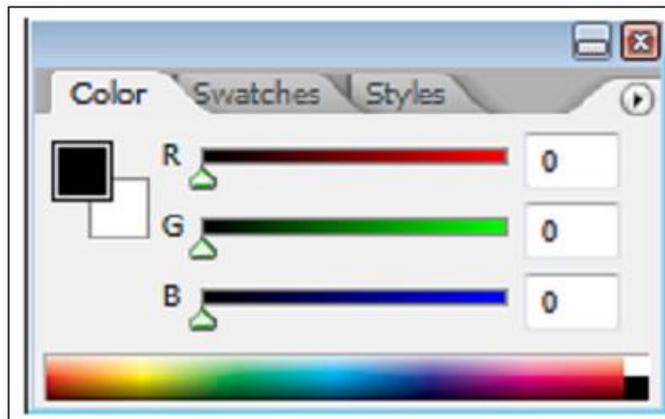
II. Citra *Grayscale* (skala keabuan)

Citra *grayscale* mempunyai kemungkinan warna hitam untuk nilai minimal dan warna putih untuk nilai maksimal. Banyaknya warna tergantung pada jumlah bit yang disediakan di memori untuk menampung kebutuhan warna tersebut. Semakin besar jumlah bit warna yang disediakan di memori, maka semakin halus gradasi warna yang terbentuk.



Gambar 2. 3 *Grayscale*

III. Citra Warna (*true color*)



Gambar 2. 4 Citra warna dengan komposisi RGB

Setiap piksel pada citra warna mewakili warna yang merupakan kombinasi tiga warna dasar, yaitu merah, hijau, dan biru (RGB = *Red, Green, Blue*). Setiap warna dasar menggunakan penyimpanan 8 bit = 1 *byte* (nilai maksimum 255 warna), jadi satu piksel pada citra warna diwakili oleh 3 *byte*.

2.2.2 Mentimun

Mentimun (*Cucumis sativus L*) merupakan buah yang dapat dikonsumsi dan dapat diolah lebih lanjut sebagai bahan baku pada industri kecantikan, dan memiliki pangsa pasar yang luas mulai dari pasar tradisional hingga pasar modern. Hal ini menunjukkan bahwa komoditas mentimun sudah dikonsumsi masyarakat secara luas dan memiliki daya saing. Dalam rangka meningkatkan daya saing tersebut maka buah mentimun yang dihasilkan harus dapat memenuhi standar pasar dalam negeri maupun pasar internasional dan di terima secara luas oleh konsumen. (Ardi Pujiyanta, Sovian Rizqiawan 2016)

Selain dari usia tanam, segi warna kulit, dan ukuran fisik buah, mentimun juga bisa diketahui kematangannya dari sisi tekstur kulitnya. Mentimun mentah tekstur garisnya kurang terlihat dan lebih kasar, mentimun setengah matang teksturnya sudah terlihat tetapi belum terlalu halus, dan mentimun matang yang jelas terlihat dan halus (*smooth*), sedangkan dari sisi jumlah sebaran teksturnya mentimun matang memiliki sebaran lebih sedikit, dibandingkan antara mentimun setengah matang, dan mentimun mentah. (Ardi Pujiyanta, Sovian Rizqiawan 2016)

2.2.3 Pengolahan Citra Digital

Pengolahan citra digital adalah salah satu bentuk pemrosesan informasi dengan inputan berupa citra (*image*) dan keluaran yang juga berupa citra atau dapat juga bagian dari citra tersebut. Tujuan dari pemrosesan ini adalah memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau mesin komputer. Operasi-operasi pada pengolahan citra digital secara umum dapat diklasifikasikan sebagai berikut (Aris Budi S, Suma'inna, Hata Maulana, 2016):

- a. Perbaikan kualitas citra (*image enhancement*), contohnya perbaikan kontras gelap/terang, penajaman (*sharpening*), dan perbaikan tepian objek (*edge enhancement*).
- b. Restorasi citra (*image restoration*), contohnya penghilangan kesamaran (*deblurring*).
- c. Pemampatan citra (*image compression*).
- d. Segmentasi citra (*image segmentation*).
- e. Pengorakan citra (*image analysis*), contohnya pendeteksian tepi objek (*edge enhancement*) dan ekstraksi batas (*boundary*).
- f. Rekonstruksi citra (*image reconstruction*).

Di dalam bidang komputer, sebenarnya ada tiga bidang studi yang berkaitan dengan data citra, namun tinjauan ketiganya berbeda, yaitu (Abdullah, Usman, 2016):

- a. Grafika Komputer (*computer graphics*).
- b. Pengolahan Citra (*image processing*).
- c. Pengenalan Pola (*pattern recognition/image interpretation*).

2.2.4 Digitalisasi Citra

Representasi citra dari fungsi malar (kontinue) menjadi nilai-nilai diskrit disebut digitalisasi. Citra yang dihasilkan inilah disebut citra digital (*digital image*). Pada umumnya citra digital berbentuk empat persegi panjang, dan dimensi ukurannya dinyatakan sebagai tinggi x lebar (atau lebar x panjang) (Dhanar Intan Surya Saputra, Tu Bagus Pranata, Sitaresmi Wahyu Handani, 2016). Citra digital yang tingginya N , lebarnya M , dan memiliki L derajat keabuan dapat dianggap sebagai fungsi :

$$f(x, y) = \begin{cases} 0 \leq x \leq M \\ 0 \leq y \leq N \\ 0 \leq f \leq L \end{cases}$$

Gambar 2.5 Fungsi Digitalisasi Citra Keabuan

Citra digital yang berukuran $N \times M$ lazim dinyatakan dengan matriks berukuran N baris dan M kolom sebagai berikut:

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & \dots & \dots & f(1,M-1) \\ \dots & \dots & \dots & \dots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

Gambar 2. 6 Fungsi Citra Digital

Indeks baris (i) dan indeks kolom (j) menatakan suatu koordinat titik pada citra, sedangkan $f(i, j)$ merupakan intensitas (derajat keabuan) pada titik (i, j). Masing-masing elemen pada citra digital (berarti elemen matriks) disebut *image element*, *picture element* atau *pixel* atau *pel*. Jadi, citra yang berukuran $N \times M$ mempunyai NM buah *pixel* (Dhanar Intan Surya Saputra, Tu Bagus Pranata, Sitaresmi Wahyu Handani, 2016). Sebagai contoh, misalkan sebuah citra berukuran 256×256 *pixel* dan direpresentasikan secara numerik dengan matriks yang terdiri dari 256 buah baris (di-indeks dari 0 - 255) dan 256 buah kolom (di-indeks dari 0 - 255) seperti contoh berikut:

$$\begin{pmatrix} \mathbf{0} & \mathbf{134} & \mathbf{145} & \dots & \dots & \mathbf{231} \\ \mathbf{0} & \mathbf{167} & \mathbf{201} & \dots & \dots & \mathbf{197} \\ \mathbf{220} & \mathbf{187} & \mathbf{189} & \dots & \dots & \mathbf{120} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{221} & \mathbf{219} & \mathbf{210} & \dots & \dots & \mathbf{156} \end{pmatrix}$$

Gambar 2.7 Index Citra Digital

- a. Proses digitalisasi spasial (x, y), sering disebut sebagai pengerokan (*sampling*).
- b. Digitalisasi intensitas $f(x, y)$, sering disebut sebagai kuantisasi.

2.2.5 Konversi Citra RGB Menjadi *Grayscale*

Dalam *image processing* banyak dilakukan konversi citra berwarna menjadi *grayscale*. Tujuan dari konversi citra ini adalah untuk memperkecil memori yang dibutuhkan sehingga dapat mempercepat proses yang dilakukan.

Proses ini sangat membantu dalam pemrograman karena dapat memanipulasi bit yang tidak terlalu banyak. Citra berwarna terdiri dari 3 *layer* matrik yaitu *R-layer*, *G-layer*, dan *B-layer*. (Ivan Maulana, Pulung Nurtantio Andono, 2016)

Untuk mengubah citra berwarna yang mempunyai nilai matrik masing-masing *r*, *g*, dan *b* menjadi *grayscale* dengan nilai *s*, maka konversi dapat dilakukan dengan mengambil rata-rata dari nilai *r*, *g*, dan *b* sehingga dapat dituliskan menjadi:

$$I(i,j) = \frac{R(i,j) + G(i,j) + B(i,j)}{3}$$

Gambar Rumus 2.1 *Grayscale*

Dimana :

$I(i,j)$ = Nilai intensitas citra *grayscale*

$R(i,j)$ = nilai intensitas warna merah dari citra asal

$G(i,j)$ = nilai intensitas warna hijau dari citra asal

$B(i,j)$ = nilai intensitas warna biru dari citra asal.

2.2.6 Transformasi *Wavelet*

Cara kerja transformasi *wavelet* adalah dengan melakukan dekomposisi pada sinyal ke dalam komponen-komponen yang berbeda kemudian melakukan analisis dari sisi skala dan resolusinya. Lalu melakukan proses penapisan dimana saat sinyal dalam domain waktu dilewatkan ke dalam *High Pass Filter* (HPF) dan *Low Pass Filter* (LPF), untuk memisahkan komponen frekuensi tinggi dan frekuensi rendah. (Eljire Bagas Lewi, Inung Wijayanto, Raditiana Patmasari, 2021). *Wavelet* merupakan sebuah fungsi dengan fungsi variabel *real*, diberi notasi Ψ dalam ruang fungsi:

$$\psi_{a,b}(t) = a^{-\frac{1}{2}} \psi\left(\frac{t-b}{a}\right); a > 0, b \in \mathfrak{R}$$

$$\psi_{j,k}(t) = a^{j/2} \psi(2^j t - k); j, k \in \mathbf{Z}$$

Gambar Rumus 2.2 *Transform Wavelet*

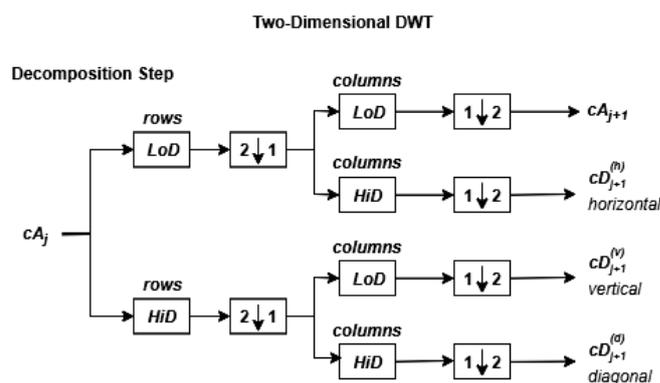
Dimana:

a = parameter dilatasi

b	=	parameter translasi
\mathcal{R}	=	mengkondisikan nilai a dan b bernilai real
$2j$	=	parameter dilatasi
k	=	parameter waktu atau lokasi ruang
Z	=	mengkondisikan nilai j dan k bernilai integer

2.2.7 Transformasi Wavelet Diskrit

Transformasi Wavelet diskrit secara umum merupakan dekomposisi citra pada frekuensi *subband* citra tersebut dimana komponennya dihasilkan dengan cara penurunan *level* dekomposisi. Implementasi transformasi Wavelet diskrit dapat dilakukan dengan cara melewatkan sinyal frekuensi tinggi atau *highpass filter* dan frekuensi rendah atau *lowpass filter*. Dibawah ini adalah gambar dari transformasi Wavelet diskrit dua dimensi dengan *level* dekomposisi satu. (Miftahur Danar Ramadhan, 2018)



Gambar 2.8 Transformasi Wavelet Diskrit Dua Dimensi dengan *Level* Dekomposisi Satu

Dimana:

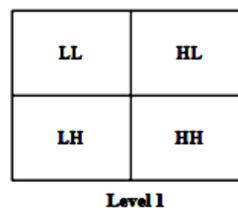
- $\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$ merupakan downsample kolom
- $\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}$ merupakan downsample baris

Jika suatu citra dilakukan proses transformasi Wavelet diskrit dua dimensi dengan *level* dekomposisi satu, maka akan menghasilkan empat buah *subband*, yaitu:

1. Koefisien Aproksimasi (CA_{j+1}) atau disebut juga *subband* LL

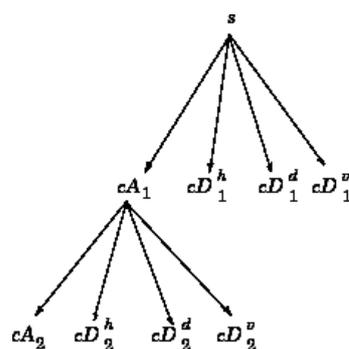
2. Koefisien Detil Horizontal ($CD(h)_{j+1}$) atau disebut juga *subband* HL
3. Koefisien Detil Vertikal ($CD(v)_{j+1}$) atau disebut juga *subband* LH
4. Koefisien Detil Diagonal ($CD(d)_{j+1}$) atau disebut juga *subband* HH

Dengan *Level* Dekomposisi 1 *subband* hasil dekomposisi dapat *didekomposisikan* lagi karena *level* dekomposisi *Wavelet* bernilai dari 1 sampai n atau disebut juga transformasi *Wavelet* *multilevel*. Jika dilakukan dekomposisi lagi, maka *subband* LL yang akan *didekomposisikan* karena *subband* LL berisi sebagian besar dari informasi citra. Jika dilakukan dekomposisi dengan *level* dekomposisi dua maka *subband* LL akan menghasilkan empat buah *subband* baru, yaitu *subband* LL2 (Koefisien Aproksimasi 2), HL2 (Koefisien Detil Horizontal 2), LH2 (Koefisien Detil Vertikal 2), dan HH2 (Koefisien Detil Diagonal 2). Dan begitu juga seterusnya jika dilakukan dekomposisi lagi.



Gambar 2.9 Dekomposisi *Level* 1

Dengan *Level* Dekomposisi 2 dapat juga dilihat seperti gambar *Wavelet tree* di bawah ini



Gambar 2.10 *Wavelet Tree*

Bila citra asli f dengan $M \times N$ *pixel* *didekomposisikan* menjadi empat *subband* sesuai frekuensinya yakni LL, LH, HL, dan HH dengan menggunakan transformasi *Wavelet* dengan filter *Haar* (*Daubechies* orde 1).

Secara matematis dihasilkan dengan menggunakan persamaan sebagai berikut:

$$\begin{aligned}
 ll_{(x,y)} &= \frac{1}{4} \sum_{i=0}^1 f(2x + i, 2y + j) \\
 lh_{(x,y)} &= \frac{1}{4} \sum_{i=0}^1 f(2x + i, 2y) - \frac{1}{4} \sum_{i=0}^1 f(2x + i, 2y + 1) \\
 hl_{(x,y)} &= \frac{1}{4} \sum_{i=0}^1 f(2x, 2y + 1) - \frac{1}{4} \sum_{i=0}^1 f(2x + 1, 2y + i) \\
 hh_{(x,y)} &= \frac{1}{4} \{f(2x, 2y) + f(2x + 1, 2y + 1) - f(2x + 1, 2y) - f(2x, 2y + 1)\}
 \end{aligned}$$

Gambar Rumus 2.3 Persamaan Dekomposisi

Dimana:

$-0 \leq x < M/2, 0 \leq y < N/2$ dan $f(x,y)$ merupakan nilai piksel pada koordinat (x,y) pada citra f .

$-ll(x,y), lh(x,y), hl(x,y),$ dan $hh(x,y)$ secara berturut-turut adalah komponen pada koordinat (x,y) dari LL, LH, HL, dan HH.

$-LL$ merupakan setengah dari resolusi citra asli, LH merupakan *subband* detail vertikal, HL merupakan *subband* detail horisontal, HH merupakan *subband* detail diagonal. (Torkis Nasution, Adyanata Lubis, 2021)

2.2.8 Filter Wavelet Daubechies

Wavelet Daubechies merupakan salah satu jenis *wavelet* yang memiliki support yang kompak. Jenis *wavelet* ini dikembangkan oleh Ingrid *Daubechies* sekitar tahun 1990-an. *Wavelet Daubechies* merupakan keluarga fungsi yang disebut dengan *Wavelet Daubechies* orde N (DbN), untuk suatu N bilangan asli. *Wavelet Daubechies* orde N dibangun berdasarkan Polinomial *Daubechies* orde $N-1$ didefinisikan sebagai berikut:

$$P_{N-1}(x) = \sum_{k=0}^{N-1} \binom{2N-1}{k} x^k (1-x)^{N-1-k}.$$

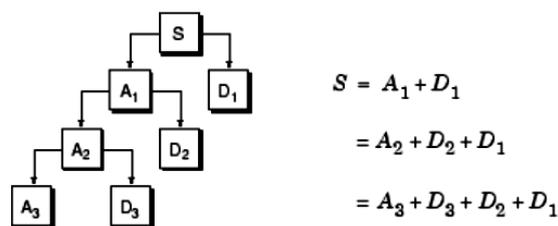
Gambar Rumus 2.4 *Daubechies*

Wavelet Daubechies yang dibangun oleh polinom *Daubechies* orde N pada Persamaan (2.3). (Syamsul Bahri, Syamsuddin, dan Mustika Hadijati, 2019)

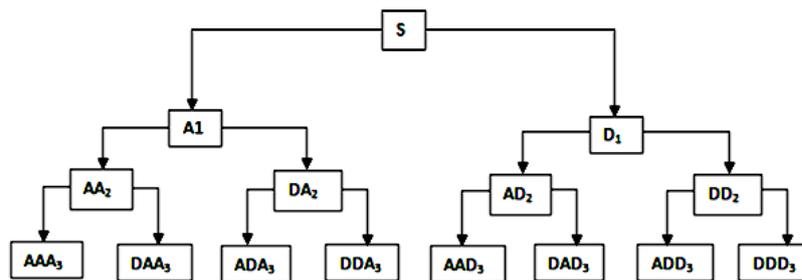
2.2.9 Paket *Wavelet*

Metode transformasi paket *wavelet* adalah suatu generalisasi dari dekomposisi *wavelet* yang menawarkan kemungkinan analisa sinyal yang lebih luas. Di dalam analisa *wavelet*, suatu sinyal dipecah atau didekomposisi menjadi koefisien aproksimasi dan koefisien detail. Selanjutnya koefisien aproksimasi tersebut didekomposisi kembali menjadi koefisien aproksimasi dan detail untuk *level* selanjutnya, sehingga pada n *level* dekomposisi menghasilkan $n+1$ koefisien yang baru $A_n + D_n + D_{n-1} + \dots + D_1$. Ilustrasi dari dekomposisi *wavelet* dapat dilihat pada gambar 2-10.

Level satu menghasilkan A_1 dan D_1 dimana A_1 adalah koefisien aproksimasi dan D_1 adalah koefisien detail. Lalu untuk *level* dua koefisien, aproksimasi di *level* satu dipecah kembali menjadi koefisien aproksimasi (A_2) dan detail (D_2) demikian seterusnya. (Abdul Hafidh Zaini, Bambang Hidayat, Rudy Hartanto, 2019)

Gambar 2.11 Tranformasi Paket *Wavelete*

Pada analisa paket *wavelet* koefisien aproksimasi dan koefisien detail sama-sama didekomposisikan kembali pada *level* selanjutnya sehingga pada n *level* dekomposisi menghasilkan koefisien yang berbeda. Namun, karena proses *downsampling* jumlah keseluruhan koefisien masih sama dan tidak ada *redundancy* terhadap jumlah koefisien aslinya. Ilustrasi paket *wavelet* dapat dilihat pada gambar berikut:



Gambar 2. 12 Paket Detail

Pada *level* satu menghasilkan A1 dan D1 dimana A1 adalah koefisien aproksimasi dan D1 adalah koefisien detail. Lalu untuk *level* dua, koefisien aproksimasi dan koefisien detail di *level* satu dipecah kembali, koefisien aproksimasi dipecah menjadi koefisien aproksimasi (AA2) dan koefisien detail (DA2) koefisien detail juga dipecah menjadi koefisien aproksimasi AD2 dan koefisien detail (DD2). Jadi untuk *level* 2 koefisien yang dihasilkan adalah (AA2), (DA2), AD2, (DD2). Demikian seterusnya sampai *level* ke-n yang diinginkan. (Abdul Hafidh Zaini, Bambang Hidayat, Rudy Hartanto, 2019)

2.2.10 Tekstur (*Texture*)

Tekstur dicirikan sebagai distribusi spasial dari derajat keabuan didalam sekumpulan *pixel-pixel* yang bertetangga. Jadi, tekstur tidak diidentifikasi untuk sebuah *pixel*, melainkan suatu citra dianggap sebagai suatu kesatuan. Dapat pula dikatakan bahwa tekstur (*texture* adalah sifat sifat atau karakteristik yang dimiliki oleh suatu daerah yang cukup besar sehingga secara alami sifat tersebut dapat berulang dalam daerah tersebut. Pengertian dari tekstur dalam hal ini adalah keteraturan pola-pola tertentu yang terbentuk dari susunan *pixel-pixel* dalam citra. Suatu permukaan dikatakan mempunyai informasi tekstur, jika luasannya diperbesar tanpa mengubah skala, maka sifat-sifat permukaan hasil perluasan mempunyai sifat kemiripan dengan permukaan asalnya.

Pola-pola yang teratur muncul secara berulang-ulang dengan interval jarak dan arah yang tertentu. Suatu permukaan tak berwarna dalam suatu citra dapat mengandung informasi tekstur bila permukaan itu mempunyai pola-pola tertentu seperti permukaan kayu bekas digergaji, permukaan batu, hamparan pasir, kumpulan biji-bijian dan sebagainya. Informasi tekstur dapat digunakan

untuk membedakan sifat permukaan suatu benda dalam citra yang berhubungan dengan kasar dan halus, sifat-sifat spesifik dari kekasaran dan kehalusan permukaan tersebut, yang sama sekali lepas dari warna permukaan tersebut. Analisis tekstur lazim dimanfaatkan sebagai proses antara untuk melakukan klasifikasi dan interpretasi citra. Suatu proses klasifikasi citra berbasis analisis tekstur pada umumnya membutuhkan tahapan ekstraksi ciri (Ardi Pujiyanta, Sovian Rizqiawan 2016)

2.2.11 Energi

Energi menyatakan ukuran konsentrasi pasangan dengan intensitas keabuan tertentu pada matriks. Nilai energi (E) dapat dihitung dengan persamaan satu. (Danar Putra Pamungkas, 2019)

$$\sum_{i,j=0} P(i,j)^2$$

Gambar Rumus 2.5 Energi

Dimana :

i = tingkat keabuan baris ke – i

j = tingkat keabuan kolom ke – j

pi,j = Peluang keabuan baris ke – i, kolom ke – j

Nilai dari koefisien WPD untuk setiap bagian dinotasikan sebagai n_c

Energi WPD didefinisikan sebagai berikut:

$$E_B(r) = \sum_{n=1}^{n_c} C^2(n,r), \quad r = 2,3, \dots 32, \dots$$

Gambar Rumus 2.6 Energi Koefisien WPD

Energi rata-rata WPD untuk setiap bagian didefinisikan sebagai berikut:

$$\tilde{E}_B(r) = \frac{E_B(r)}{n_c}$$

Gambar Rumus 2.7 Energi Rata-Rata

Maximum energi rata-rata WPD didefinisikan sebagai berikut:

$$E_M(r) = \max_r(\tilde{E}_B(r))$$

Gambar Rumus 2.8 *Maximum Energi Rata-Rata*

2.2.12 Entropi

Entropi menunjukkan ketidakteraturan dari bentuk, jika nilai entropi-nya besar untuk citra dengan transisi nilai derajat keabuan yang teratur, dan bernilai kecil jika struktur citra tidak teratur (bervariasi). (Damar Putra Pamungkas, 2019)

Nilai Entropi (EN) dapat dihitung dengan persamaan nomor sebagai berikut:

$$EN = \sum_{i,j} p_{i,j} (-\ln(p_{i,j}))$$

Gambar Rumus 2.9 *Entropy*

Dimana :

$p_{i,j}$ = Peluang keabuan baris ke – i, kolom ke – j

$\ln p_{i,j}$ = Logaritma natural dari $p_{i,j}$

2.2.13 Jaringan Saraf Tiruan

Jaringan Syaraf Tiruan adalah merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplemintasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran (Eka Pandu Cynthia, Edi Ismanto, 2017).

Pada penelitian ini akan dirancang jaringan syaraf tiruan model lapisan banyak yang mana arsitektur tipe ini memiliki satu atau lebih lapisan yang terletak antara lapisan masukan dan lapisan keluaran, memiliki juga satu atau lebih lapisan tersembunyi.

Umumnya, ada lapisan bobot-bobot yang terletak antara dua lapisan yang bersebelahan. Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit dibandingkan lapisan dengan lapisan tunggal, tentu saja dengan pembelajaran yang lebih rumit juga.

Gambar di bawah ini merupakan salah satu contoh model neuron dengan banyak lapisan dan hanya memiliki satu lapisan tersembunyi. V_{ij} adalah bobot-

bobot yang menghubungkan antara neuron-neuron pada lapisan masukan dengan lapisan tersembunyi. Sedangkan W_{jk} adalah bobot-bobot yang menghubungkan antara neuron-neuron pada lapisan tersembunyi dengan lapisan keluaran. Z_{inj} adalah hasil pengolahan data pada lapisan tersembunyi dengan fungsi aktivasi F1 untuk menghasilkan z_j ($j=1, \dots, k$) (Eka Pandu Cynthia, Edi Ismanto, 2017):

$$z_{in_j} = \sum_{i=1}^n x_i \cdot w_{ij}$$

$$z_j = F1(z_{in_j})$$

Gambar Rumus 2. 10 Fungsi Aktivasi F1

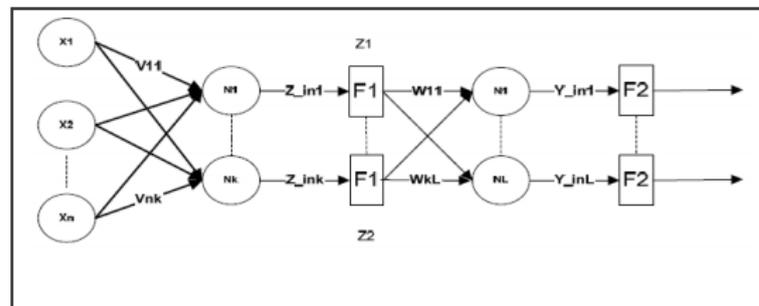
Sedangkan y_{ink} adalah hasil pengolahan data pada lapisan keluaran dengan fungsi aktivasi F2 untuk menghasilkan keluaran jaringan.

$$y_k; (k = 1, \dots, L)$$

$$y_{in_k} = \sum_{j=1}^n z_j \cdot w_{jk}$$

$$z_j = F2(y_{in_k})$$

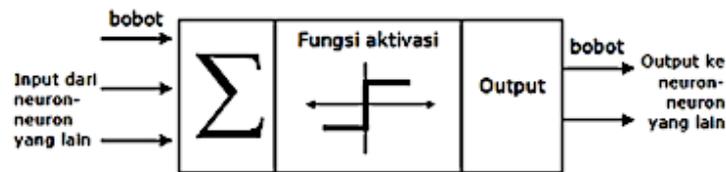
Gambar Rumus 2.11 Fungsi Aktivasi F2



Gambar 2.13 Model Neuron dengan Banyak Lapisan

2.2.14 Komponen Jaringan Saraf Tiruan

Ada beberapa tipe jaringan saraf, namun demikian hampir semuanya memiliki komponen-komponen yang sama. (Haryo, 2011)



Gambar 2.14 Susunan Neuron Jaringan Syaraf Tiruan

Pada gambar di atas dapat kita lihat, Informasi (*input*) akan terkirim ke neuron dengan bobot kedatangan tertentu. *Input* ini akan diproses oleh suatu fungsi perambatan yang akan menjumlahkan nilai-nilai semua bobot yang datang. Hasil penjumlahan ini kemudian akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap neuron. Apabila *input* tersebut melewati suatu nilai ambang tertentu, maka neuron tersebut akan diaktifkan, tapi kalau tidak, maka neuron tersebut tidak diaktifkan. Apabila neuron tersebut diaktifkan, maka neuron tersebut akan mengirimkan keluaran (*output*) melalui bobot-bobot keluarannya ke semua neuron yang berhubungan dengannya. Demikian seterusnya. (Musli Yanto, 2017)

a. Neuron

Neuron adalah bagian terkecil dalam pemrosesan pada Jaringan Syaraf Tiruan yang terkoneksi satu sama lain. Neuron-neuron tersebut akan mentransformasikan informasi yang diterima melalui interkoneksi yang terbentuk menuju neuron-neuron yang lain.

b. Bobot (*weight*)

Unsur kunci dari Jaringan Syaraf Tiruan adalah bobot. Bobot menunjukkan suatu kekuatan relatif (*relative strength*) atau nilai matematik dari *input* data atau banyaknya koneksi yang memindahkan data dari suatu lapisan ke lapisan lainnya. Intinya adalah bobot itu nilai yang menunjukkan derajat keterhubungan antara neuron pada lapisan yang satu dengan lapisan yang lain.

c. Fungsi Aktivasi

Sebuah sinyal aktivasi diperlukan oleh suatu neuron untuk mengaktifkan atau nonaktifkan penjalaran sinyal dari neuron tersebut. Sinyal aktivasi dalam Jaringan Syaraf Tiruan ditentukan oleh suatu fungsi aktivasi.

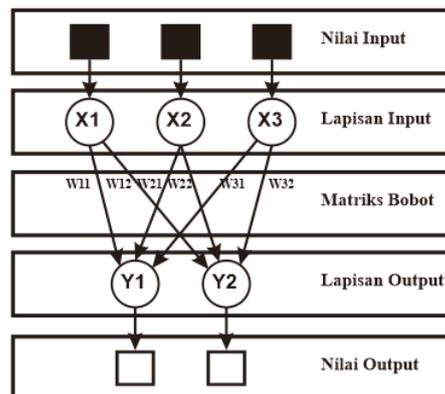
2.2.15 Arsitektur Jaringan Syaraf Tiruan

Sebagaimana yang telah dijelaskan di atas bahwa neuron-neuron dikelompokkan dalam lapisan-lapisan. Umumnya, neuron-neuron yang terletak pada lapisan yang sama akan memiliki keadaan yang sama. Faktor terpenting dalam menentukan perilaku suatu neuron adalah fungsi aktivasi dan pola bobotnya. Pada setiap lapisan yang sama, neuron-neuron akan memiliki fungsi aktivasi yang sama. (Musli Yanto, 2017)

Menurut Haryo (2011) ada beberapa arsitektur JST terdiri antara lain:

1. Jaringan dengan lapisan tunggal (*single layer net*)

Jaringan dengan lapisan tunggal hanya memiliki satu lapisan dengan bobot-bobot terhubung. Jaringan ini hanya menerima *input* kemudian secara langsung akan mengolahnya menjadi *output* tanpa harus melalui lapisan tersembunyi.

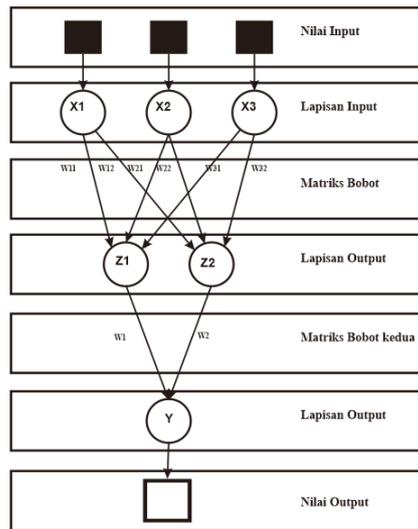


Gambar 2.15 Jaringan Saraf dengan Lapisan Tunggal

Pada Gambar 2-15 tersebut, lapisan *input* memiliki 3 *neuron*, yaitu X1, X2 dan X3. Sedangkan pada lapisan *output* memiliki 2 *neuron* yaitu Y1 dan Y2. *Neuron-neuron* pada kedua lapisan saling berhubungan seberapa besar hubungan antara 2 *neuron* ditentukan oleh bobot yang bersesuaian. Semua unit *input* akan dihubungkan dengan setiap unit *output*.

2. Jaringan dengan banyak lapisan (*multilayer net*)

Jaringan dengan banyak lapisan memiliki satu atau lebih lapisan yang terletak di antara lapisan *input* dan lapisan *output* (memiliki satu atau lebih lapisan tersembunyi), seperti terlihat pada Gambar 2.16

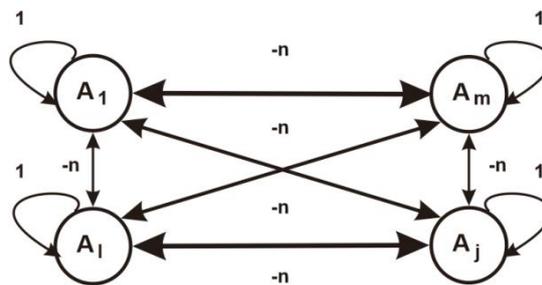


Gambar 2.16 Jaringan Saraf dengan Banyak Lapisan

Umumnya ada lapisan bobot-bobot yang terletak antara dua lapisan yang bersebelahan. Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit dari pada lapisan dengan lapisan tunggal, tentu saja dengan pembelajaran yang lebih rumit. Namun demikian, pada banyak kasus, pembelajaran pada jaringan dengan banyak lapisan ini lebih sukses dalam menyelesaikan masalah.

3. Jaringan dengan lapisan kompetitif (*competitive layer net*)

Umumnya hubungan antar *neuron* pada lapisan kompetitif ini tidak diperlihatkan pada diagram arsitektur. Gambar 2.17 menunjukkan salah satu contoh arsitektur jaringan dengan lapisan kompetitif yang memiliki bobot $-\eta$.



Gambar 2.17 Jaringan Saraf dengan Lapisan Kompetitif

2.2.16 Proses Pembelajaran

Pada jaringan syaraf tiruan (JST) terdapat proses pembelajaran yang mana terdapat 2 proses belajar yaitu (Dwi Wahyuning Jati, 2018);

a. Pembelajaran terawasi (*supervised learning*)

Supervised adalah proses belajar yang membutuhkan guru. Yang dimaksud guru disini adalah sesuatu yang memiliki pengetahuan tentang lingkungan. Guru bisa direpresentasikan sebagai sekumpulan sampel *input-output*. Pembangunan pengetahuan dilakukan oleh guru dengan memberikan respon yang diinginkan kepada JST. Respon yang diinginkan tersebut merepresentasikan aksi optimum yang dilakukan JST. Parameter-parameter jaringan berubah-ubah berdasarkan *vector* latih dan sinyal kesalahan. Proses perubahan ini dilakukan secara berulang-ulang, selangkah demi selangkah, dengan tujuan agar JST dilatih untuk dapat memetakan sekumpulan sampel *input-output* dengan akurasi tinggi. Contoh algoritma jaringan saraf tiruan yang menggunakan metode *supervised learning* adalah *hebbian (hebb rule)*, *perceptron*, *delta rule*, *heteroassociative memory*, *bidirectional associative memory (BAM)*, *backpropagation*, *Learning Vector Quantization (LVQ)*.

b. Pembelajaran tak terawasi (*unsupervised learning*)

Sesuai dengan namanya, *unsupervised* tidak membutuhkan guru untuk memantau proses belajar. Dengan kata lain, tidak ada sekumpulan sampel *input-output* atau fungsi tertentu untuk dipelajari oleh jaringan. Contoh algoritma jaringan saraf tiruan yang menggunakan metode *unsupervised* ini adalah kohonen.

2.2.17 Jaringan Saraf Tiruan *Self Organizing Map (SOM)*

Jaringan saraf tiruan *Self Organizing Maps (SOM)* atau disebut juga dengan jaringan *Kohonen* telah banyak dimanfaatkan untuk pengenalan pola baik berupa citra, suara, dan lain-lain. Jaringan *SOM* sering pula digunakan untuk ekstraksi ciri (*feature*) pada proses awal pengenalan pola. Ia mampu mereduksi dimensi *input* pola ke jumlah yang lebih sedikit sehingga pemrosesan komputer menjadi lebih hemat. JST Kohonen-*SOM* merupakan salah satu model JST yang menggunakan metode *unsupervised learning*, yaitu mempelajari distribusi himpunan pola-pola tanpa informasi kelas. Jaringan ini terdiri dari dua lapisan (*layer*), yaitu lapisan *input* dan lapisan *output*. Setiap neuron dalam lapisan *input*

terhubung dengan setiap neuron pada lapisan *output*. Setiap neuron dalam lapisan *output* merepresentasikan kelas (*cluster*) dari *input* yang diberikan.

Prinsip kerja dari algoritma *SOM* adalah pengurangan *node-node* tetangganya (*neighbor*), sehingga pada akhirnya hanya ada satu *node output* yang terpilih (*winner node*). Pertama kali yang dilakukan adalah melakukan inialisasi bobot untuk tiap-tiap *node* dengan nilai *random*. Setelah diberikan bobot random, maka jaringan diberi *input* sejumlah dimensi *node/neuron input* (10×10). Setelah *input* diterima jaringan, maka jaringan mulai melakukan perhitungan jarak vektor yang didapatkan dengan menjumlah selisih/jarak antara vektor *input* dengan vektor bobot. Pada jaringan ini, suatu lapisan yang berisi neuron-neuron akan menyusun dirinya sendiri berdasarkan *input* nilai tertentu dalam suatu kelompok yang dikenal dengan istilah *cluster*. Selama proses penyusunan diri, *cluster* yang memiliki vektor bobot paling cocok dengan pola *input* (memiliki jarak yang paling dekat) akan terpilih sebagai pemenang. Neuron yang menjadi pemenang beserta neuron-neuron tetangganya akan memperbaiki bobot-bobotnya.

Bobot vektor-vektor contoh berfungsi sebagai penentu kedekatan antara vektor tersebut dengan masukan yang diberikan. Selama proses pengaturan, vektor contoh pada saat itu paling dekat dengan masukan akan muncul sebagai pemenang, kemudian vektor pemenang dan vektor-vektor sekitarnya akan dimodifikasi bobotnya. Dalam *JST SOM*, neuron target tidak diletakkan dalam sebuah baris seperti layaknya model jaringan syaraf tiruan yang lain. Neuron target diletakkan dalam dua dimensi yang bentuk/topologinya dapat diatur, dalam matlab disediakan fasilitas untuk mendefinisikan topologi jaringan yang berbeda-beda. Topologi yang berbeda akan menghasilkan neuron sekitar neuron pemenang yang berbeda sehingga bobot yang dihasilkan juga akan berbeda. (Abdul Hafidh Zaini, Bambang Hidayat, Rudy Hartanto, 2019)

2.2.18 Algoritma Kohonen Self Organizing Map (SOM)

Adapun *algoritma* dari Kohonen *Self Organizing Map* adalah sebagai berikut:

Langkah 1: Inialisasi bobot w_{ij} . Set parameter-parameter tetangga dan set parameter *learning rate*.

Langkah 2: Selama kondisi berhenti masih bernilai salah, kerjakan langkah-langkah berikut ini:

Untuk masing-masing vektor *input* x , lakukan langkah 3-5:

- a. Untuk setiap j , hitung:

$$D(j) = \sum_i (w_{ij} - x_i)^2$$

Gambar Rumus 2.12 *Euclidean Distance*

- b. Temukan J sampai $D(j)$ bernilai minimum.
 c. Untuk masing-masing unit j dengan spesifikasi tetangga tertentu pada J dan untuk semua I , kerjakan:

$$w_{ij}(\text{baru}) = w_{ij}(\text{lama}) + \alpha[x_i - w_{ij}(\text{lama})]$$

Gambar Rumus 2.13 Perulangan Sepesifikasi Tetangga Tertentu

- d. Perbaiki *learning rate*.
 e. Kurangi radius tetangga pada waktu-waktu tertentu.
 f. Tes kondisi berhenti.

2.2.19 Confusion Matrix

Pada data mining untuk mengukur atau ada beberapa cara untuk mengukur kinerja dari model yang dihasilkan salah satunya menggunakan *confusion matrix* (akurasi). *Confusion matrix* adalah suatu metode yang digunakan untuk melakukan perhitungan akurasi pada konsep data mining. Presisi atau *confidence* adalah proporsi kasus yang diprediksi positif yang juga positif benar pada data yang sebenarnya. *Recall* atau *sensitivity* adalah proporsi kasus positif yang sebenarnya yang diprediksi positif secara benar.

Berikut merupakan bentuk dan model perhitungan Confusion matrix :

Tabel 2.1 Bentuk *Confusion matrix*

Kelas	Terklasifikasi positif	Terklasifikasi Negatif
Positif	<i>true positive (TP)</i>	<i>false negative (FN)</i>
Negative	<i>false positive (FP)</i>	<i>true negative (TN)</i>

Berdasarkan nilai *True Negative*, *False Positive*, *False Negative* dan *True Positive* dapat dihitung dan diperoleh nilai akurasi, presisi dan *recall*. Nilai akurasi ini menggambarkan seberapa akurat sistem dapat mengklasifikasikan data secara benar, nilai akurasi merupakan perbandingan antara data yang terklasifikasi benar dengan keseluruhan data. Nilai akurasi dapat diperoleh dengan Persamaan 1. Nilai presisi menggambarkan jumlah data kategori positif yang diklasifikasikan secara benar dibagi dengan total data yang diklasifikasi positif untuk nilai presisi dapat diperoleh dengan menggunakan persamaan 2. Sementara itu, *recall* menunjukkan berapa persen data kategori positif yang terklasifikasikan dengan benar oleh sistem. Nilai *recall* dihitung dan dengan persamaan 3.

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} * 100\%$$

$$Presisi = \frac{TP}{FP+TP} * 100\%$$

$$Recall = \frac{TP}{FN+TP} * 100\%$$

Gambar 2.18 Perhitungan *Confusion Matrix*

Dimana:

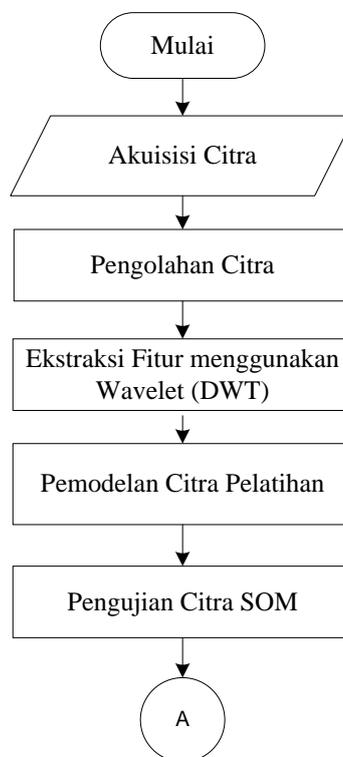
- a. TP adalah *True Positive*, yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem.
- b. TN adalah *True Negative*, yaitu jumlah data negatif yang terklasifikasi dengan benar oleh sistem.
- c. FN adalah *False Negative*, yaitu jumlah data negatif namun terklasifikasi salah oleh sistem.
- d. FP adalah *False Positive*, yaitu jumlah data positif namun terklasifikasi salah oleh sistem. (Auni Aulia Muftiany, 2018)

BAB 3

METODOLOGI PENELITIAN

3.1 Prosedur Penelitian

Berdasarkan permasalahan tersebut, maka dibutuhkan sebuah *system* yang dapat mengenali tingkat kematangan pada buah mentimun berdasarkan tekstur kulit. Untuk menyelesaikan masalah tersebut, adapun tahap penelitian secara grafis yang dapat dilihat pada gambar 3.1 sebagai berikut :



Gambar 3.1 Alur Penelitian

Gambar 3.1 Alur penelitian terdiri dari tahapan sebagai berikut :

1. Akuisisi Citra

Citra mentimun yang digunakan yaitu berjumlah 90 citra dataset yang diambil dengan menggunakan kamera digital pada siang hari . Citra tersebut dibagi menjadi 2 data yang terdiri dari data pelatihan (Citra *Database*) dan data pengujian (Citra yang akan diuji) 72 buah citra sebagai

pelatihan (*training*). Adapun dari 72 citra pelatihan terdiri dari 1 *folder* yang berisikan 24 tekstur citra mentimun matang, 1 *folder* 24 tekstur citra mentimun setengah matang, dan 1 *folder* 24 citra mentimun dengan mentah. Sisanya citra mentimun yang digunakan untuk *testing* atau citra uji yang berjumlah 18 buah. Citra mentimun yang akan diujikan terdiri dari 6 citra mentimun matang, 6 citra mentimun setengah matang, dan 6 citra mentimun mentah.

2. Pengolahan Citra

1) *Cropping* Citra

Selanjutnya citra tersebut dibaca menggunakan aplikasi matlab dengan kordinat *crop region* citra yang diambil pada penelitian ini tekstur kulit dengan tekstur bintik-bintik halus (*smooth*) untuk matang, kulit mentimun dengan tekstur bintik-bintik sedang untuk setengah matang, kemudian tekstur kulit mentimun yang berwarna hijau dengan tekstur bintik-bintik kasar untuk mentah (Ardi Pujiyanta, Sovian Rizqiawan 2016) dari setiap ciri untuk mendapatkan nilai *energy* dan *entropy* dari masing masing citra yang akan digunakan sebagai sampel pada data *training database* dan *testing* untuk citra uji. Setelah dilakukan proses *cropping*, kemudian ukuran citra disamakan yaitu 512 x 512 piksel.

2) *Grayscale* Tekstur Citra

Setelah citra berhasil *cropping* pada region, selanjutnya konversi *grayscale*, hasil *cropping* maka secara otomatis citra *truecolor* 24 bit langsung dikonversi ke citra *grayscale*.

3. Ekstrasi Fitur DWT (*Dekomposisi Wavelet Transform*)

Proses selanjutnya ekstraksi fitur citra digital menggunakan transformasi *wavelet* diskrit dengan tekstur penskalaan filter dekomposisi *level 1* dan *level 2* untuk menghasilkan vektor ciri sehingga menghasilkan vektor ciri LL dan LH sebelum didapatkan nilai *entropy* dan *energy*.

4. Pemodelan Citra Pelatihan

Pada tahapan ini dilakukan dengan menggunakan 72 *sample* data *training* yang terdiri dari 3 *folder* yang terdiri dari 24 tekstur citra matang ,

24 tekstur citra setengah matang, dan 24 tekstur citra mentah yang sudah di *cropping* menggunakan kordinat *position* pada *system* . Pelatihan digunakan untuk mengumpulkan perhitungan tekstur *range entropy* dan *energy* dari setiap tekstur citra latih, yang akan disimpan ke dalam *workspace* pada *matlab* yang bernama *trainingdata.mat* yang nantinya akan diambil kembali pada proses pengujian dengan terhadap 3 *folder* tekstur .

5. Pengujian Citra *Self Organizing Maps*

Pada tahapan ini dilakukan dengan menggunakan 18 pengujian data sample citra uji tekstur mentimun yang terbagi menjadi 6 uji tekstur citra mentimun matang, 6 uji tekstur citra setengah matang dan 6 tekstur citra uji mentimun mentah sebagai pengamatan dan pengujian manual, Kedua perhitungan tekstur (*entropy* dan *energy*) dari citra yang akan diuji dilatih menggunakan jaringan syaraf tiruan metode *self organizing map* (*SOM*) berdasarkan tekstur data latih yang dihasilkan dari proses pelatihan yaitu *file trainingdata.mat*. Hasil dari pengambilan bobot merupakan kode kematangan dari tekstur buah mentimun yang akan mendapatkan seberapa ketepatan dan kesesuaian *Self Organizing Maps* pada aplikasi untuk identifikasi kematangan mentimun. yang akan mendapatkan hasil matang, setengah matang atau mentah berdasarkan tekstur buah mentimun yang diuji.

3.2 Analisa Sistem

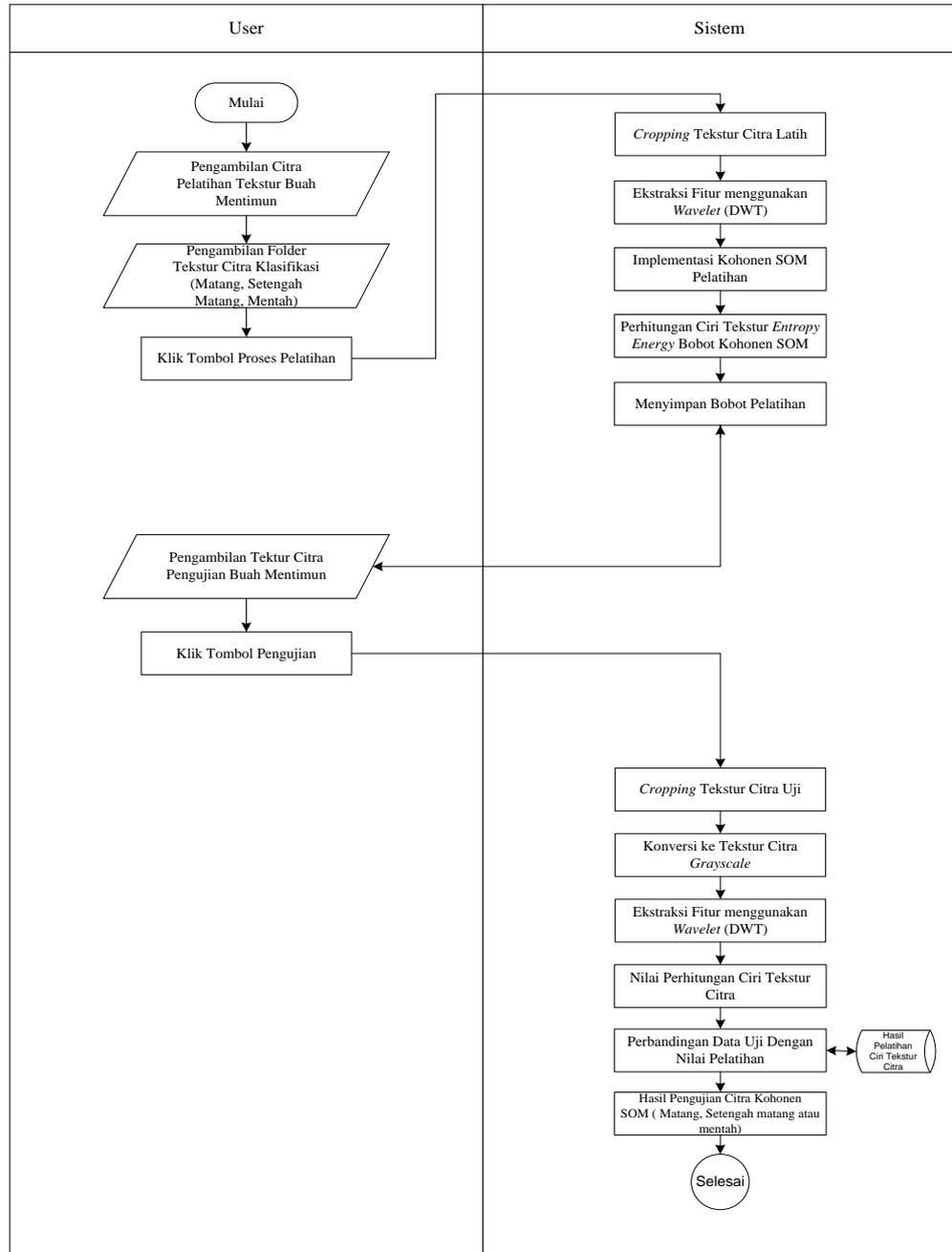
3.2.1 Analisa Sistem Yang Berjalan

Berdasarkan observasi di lapangan, diperoleh informasi bahwa penyortiran hasil panen buah mentimun masih dilakukan secara manual, yaitu dengan penginderaan mata memisahkan satu persatu buah mentimun dengan kategori mentah, setengah matang, dan matang Sortasi yang dilakukan secara manual adalah sortasi yang berdasarkan pada persepsi manusia terhadap faktor komposisi tekstur, warna ataupun kerusakan yang dimiliki buah. Penilaian manusia dapat berbeda dari satu penilai dengan penilai lainnya.

Seorang penilai A bisa mengatakan bahwa buah mentimun yang dinilai telah matang akan tetapi penilai B bisa mengatakan bahwa buah mentimun tersebut setengah matang ataupun mentah. Kesalahan penggolongan tingkat kematangan mentimun bisa merugikan produsen atau penjual dan tentunya pembeli.

3.2.2 Analisa Sistem yang Diusulkan

Berdasarkan hasil analisis sistem yang sedang berjalan (penyortiran buah mentimun secara manual), maka diusulkan untuk membuat *system* aplikasi komputer pengolahan citra yang dapat digunakan untuk mengenali pola tingkat kematangan buah mentimun pada tekstur kulit menggunakan JST *Self Organizing Maps* berdasarkan 3 (tiga) kategori dibidang pengembangan pertanian dengan memanfaatkan potret kamera digital. Adapun *flowmap system* yang diusulkan adalah sebagai berikut:



Gambar 3. 2 Alur Penelitian Sistem Yang Diusulkan

Langkah kerja pada sistem yang diusulkan dimulai dengan *user* pelatihan (*training*) dengan mengambil *folder* data latih klasifikasi tekstur kematangan mentimun, dari 1 *folder* yang berisikan 24 citra latih tekstur matang, 1 *folder* yang berisikan 24 citra tekstur setengah matang, dan 1 *folder* yang berisikan 24 citra tekstur mentah. Selanjutnya dari 72 citra *training* tersebut akan otomatis dibaca menggunakan aplikasi *matlab* dan *system* akan melakukan *cropping* tekstur sesuai pemotongan *rec pos* kedominanan sebagai penelitian sampel data pelatihan dari tekstur citra latih.

Setelah dilakukan *cropping* pada *system*, kemudian *system* melakukan proses ekstraksi fitur citra digital menggunakan transformasi *wavelet* diskrit sebagai urutan untuk mencari nilai *entropy* dan *energy*. Dari bobot *energy* dan *entropy* pelatihan akan tersimpan otomatis didalam *traindata.mat* pada *matlab*.

Langkah pada proses pengujian, yaitu *user* mengambil sebuah sample tekstur citra uji matang, tekstur setengah matang dan tekstur mentah masing masing 6 buah citra, kemudian sama seperti sebelumnya *system* akan *cropping* otomatis dan *system* mengkonversi hasil *cropping* ke Tekstur *Grayscale*, hingga tranformasi *wavelet* diskrit (*DWT*) untuk mendapatkan nilai ciri *energy* dan *entropy* dari citra yang sedang diuji.

Pada langkah terakhir *system* melakukan pengujian nilai ciri *entropy* dan *energy* dari tekstur citra uji, pengambilan bobot citra uji terhadap 72 *training* yang berada di *trainingdata.mat* sesuai nilai *range entropy* dan *energy* dari setiap masing-masing citra *range* tekstur matang, *range* tekstur setengah matang dan *range* tekstur mentah yang tersimpan. yang akan menghasilkan salah satu kelas kematangan dari tekstur buah mentimun citra yang sedang diuji dengan menggunakan jaringan syaraf tiruan metode *Self Organizing Maps (SOM)*.

3.3 Perancangan Sistem

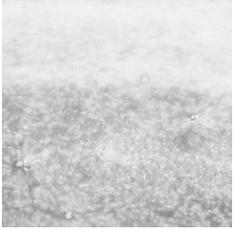
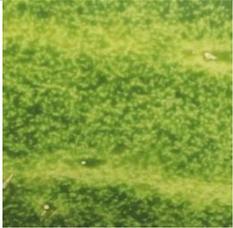
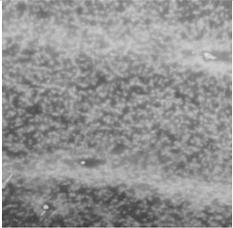
3.3.1 Analisis kebutuhan data

Analisis data dilakukan dengan pengambilan citra buah mentimun dengan jarak 20 cm sejumlah 90 sampel pada siang hari, dimana 72 sampel yang diambil dengan menggunakan kamera digital digunakan untuk data *training* dalam *database*, yang terdiri dari 24 tesktur citra mentimun matang, 24 tesktur citra mentimun setengah matang, dan 24 tesktur citra mentimun mentah. sedangkan 18 sampel tesktur citra buah mentimun digunakan untuk *testing* yang terdiri dari 6 citra uji tekstur mentimun matang, 6 citra uji tekstur mentimun setengah matang, dan 6 citra uji tekstur mentimun mentah.

Pengambilan citra dilakukan satu kali dengan prosedur pengambilan citra sebagai berikut :

1. Citra mentimun ditangkap oleh kamera digital melalui lensa waktu siang hari, kemudian data pada kamera tersebut ditransfer ke perangkat komputer menggunakan kabel data dan ditampilkan dimonitor.
2. Citra mentimun yang telah direkam berformat *default *.jpg* kemudian dikonversi ke *format *.bmp* untuk 72 data pelatihan dan sisanya 18 berformat **.jpg* untuk data pelatihan dan pengujian citra uji kemudian pengamatan *cropping* menggunakan kordinat *position* masing-masing tesktur citra dengan menggunakan program *rec pos sample* sebagai pengujian manual sebagai berikut :

Tabel 3.1 *Cropping* Citra Asli dan Tekstur *Grayscale*

<i>Cropping</i> Tekstur Mentimun	<i>Cropping</i> Tekstur Grayscale	Keterangan
		Mentimun Matang
		Mentimun Setengah Matang
		Mentimun Mentah

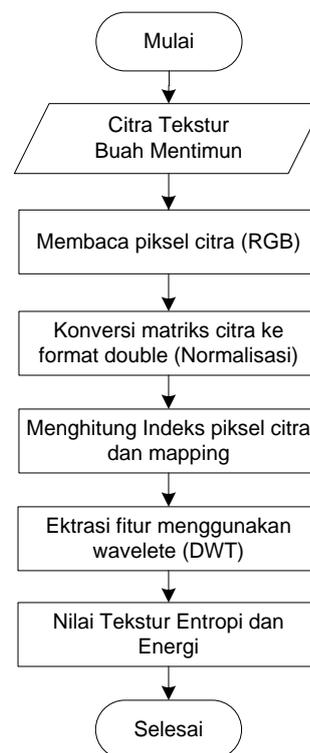
Mentimun matang yang jelas terlihat dan halus (*smooth*), mentimun setengah matang teksturnya sudah terlihat tetapi belum terlalu halus, dan mentimun mentah tekstur garisnya kurang terlihat dan lebih kasar, sedangkan dari sisi jumlah sebaran teksturnya mentimun matang memiliki sebaran lebih sedikit, dibandingkan antara mentimun setengah matang, dan mentimun mentah sesuai pengamatan. (Ardi Pujiyanta, Sovian Rizqiawan 2016)

Pada tahap analisis data dilakukan penetapan jenis citra *input* dan *output* serta perancangan tampilan. Kemudian mengimplementasikan aplikasi deteksi fitur tekstur kulit buah untuk mengetahui kematangan mentimun secara statistik.

Topologi yang digunakan pada penelitian ini yaitu *hextop* dalam struktur jaringan *SOM* tidak mempunyai *hidden layer* dengan

pengklasifikasian 1 *input* an (*entropy energy*) dan *output* (1) Matang, (2) Setengah Matang Dan (3) Mentah.

Nilai *entropy* dan *energy* didapat dari proses dekomposisi *wavelet*. Selanjutnya data tersebut dilatih oleh jaringan syaraf tiruan metode *SOM* dan menghasilkan bobot untuk masing-masing sampel yang digunakan. Data yang diperoleh sebagaimana tercantum pada bab 3 kemudian dilakukan pemrosesan dan syaraf tiruan metode *SOM*.

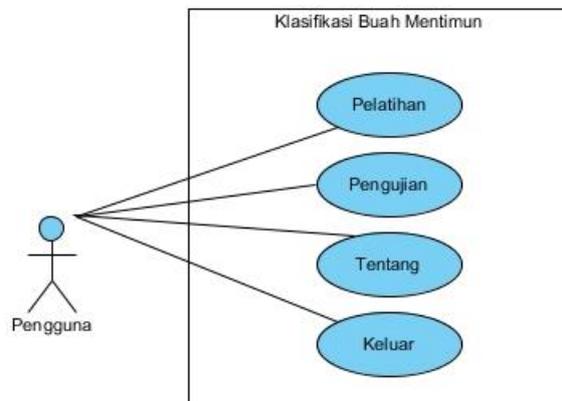


Gambar 3.3 *Flowchart* Proses Perhitungan Data

3.3.2 Perancangan object oriented/procedural

Dengan adanya sistem yang diusulkan, maka dibutuhkan perancangan sistem menggunakan *Unified Modeling Language* (UML) sebagai alur program *object oriented*

1. Use Case Diagram



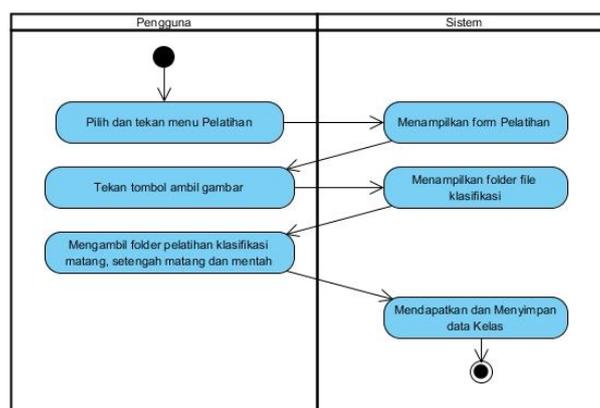
Gambar 3. 4 *Use Case Diagram*

Aplikasi yang akan dibuat pada penelitian ini terdiri dari satu aktor yaitu pengguna dan tiga modul (sub sstem) utama yaitu pelatihan, pengujian, tentang, dan keluar. Pengguna dapat secara langsung berinteraksi dengan semua modul tersebut tanpa perlu adanya proses autentikasi.

2. Activity Diagram

Untuk diagram aktifitas sistem identifikasi kematangan buah mentimun pada tekstur kulit sebagai berikut:

a. *Activity Diagram Pelatihan*

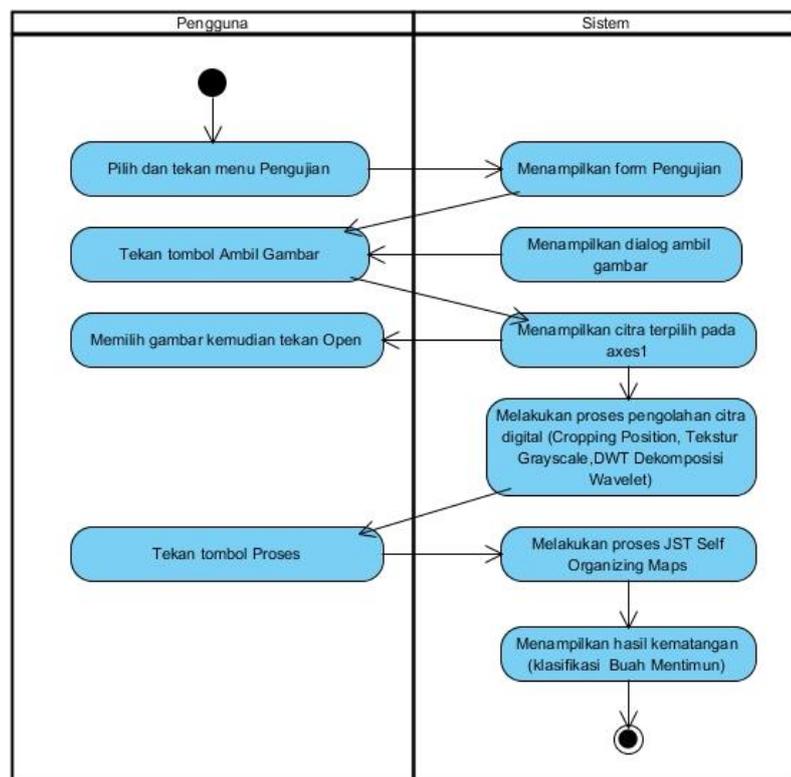


Gambar 3.5 *Activity Diagram Pelatihan*

Pada proses pelatihan, pengguna harus menekan tombol Pelatihan pada *form* utama aplikasi agar sistem membuka *form* pelatihan. Langkah selanjutnya adalah mengambil *folder* klasifikasi citra buah mentimun. Urutan proses yang akan dikerjakan oleh sistem yaitu *cropping* kordinat, transformasi *wavelet* diskrit yang hasil akhirnya adalah nilai tekstur *entropy* dan *energy*.

Aplikasi yang akan dibuat pada penelitian ini terdiri dari satu aktor yaitu pengguna dan tiga modul (sub sistem) utama yaitu pelatihan, pengujian, tentang, dan keluar. Pengguna dapat secara langsung berinteraksi dengan semua modul tersebut tanpa perlu adanya proses autentikasi.

b. Activity Diagram Pengujian



Gambar 3.6 Activity Diagram Pengujian

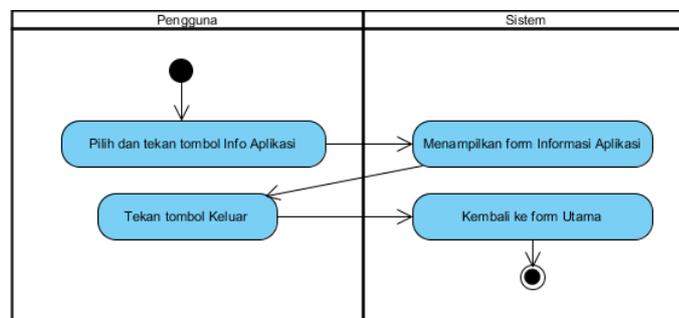
Pada proses pengujian, pengguna harus menekan tombol Pengujian pada *form* utama aplikasi agar sistem membuka *form* pengujian. Langkah selanjutnya adalah dengan menekan tombol

ambil gambar kemudian pada direktori pilih citra uji buah mentimun (matang, setengah matang atau mentah) sesuai dari sampel data citra yang akan diujikan.

Setelah citra yang akan diuji berhasil ditampilkan, kemudian Urutan proses yang akan dikerjakan oleh sistem yaitu *cropping*, tekstur *grayscale*, transformasi *wavelet* diskrit dan juga waktu proses yang hasil akhirnya akan mendapatkan nilai tekstur *entropy* dan *energy* dari citra yang sedang diujikan. Pengguna harus menekan tombol proses untuk mendapatkan nilai *energy entropy* serta waktu proses yang diperoleh. setelah menekan tombol proses *system* memvisualisasi *mapping* hasil *JST Self Organizing Maps (SOM) & Pohon pemfaktoran bobot frekuensi analisis sinyal dari pemfilteran DWT (Discrete wavelet transform)* dari citra yang sedang diuji, visualisasi *SOM* terdiri dari 72 *mapping training*, 24 *mapping* citra latih matang, 24 *mapping* citra latih setengah matang dan 24 *mapping* citra latih mentah dan 1 data *mapping* citra yang sedang diujikan yang akan menampilkan hasil akhir klasifikasi kematangan.

Aplikasi yang akan dibuat pada penelitian ini terdiri dari satu aktor yaitu pengguna dan tiga modul (sub sistem) utama yaitu pelatihan, pengujian, tentang, dan keluar

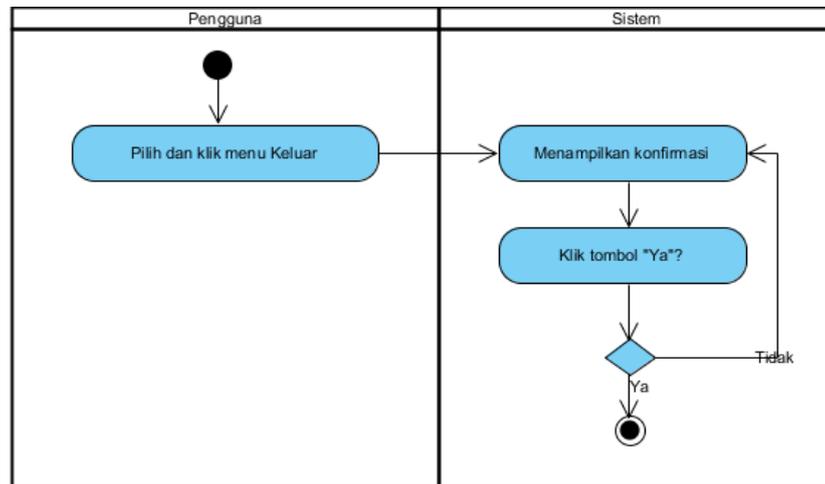
c. Activity Diagram Tentang



Gambar 3. 7 Activity Diagram Tentang

Pengguna dapat melihat informasi aplikasi dengan menekan tombol Info Aplikasi, kemudian sistem akan menampilkan *form* informasi aplikasi.

d. *Activity Diagram Keluar*



Gambar 3.8 *Activity Diagram Keluar*

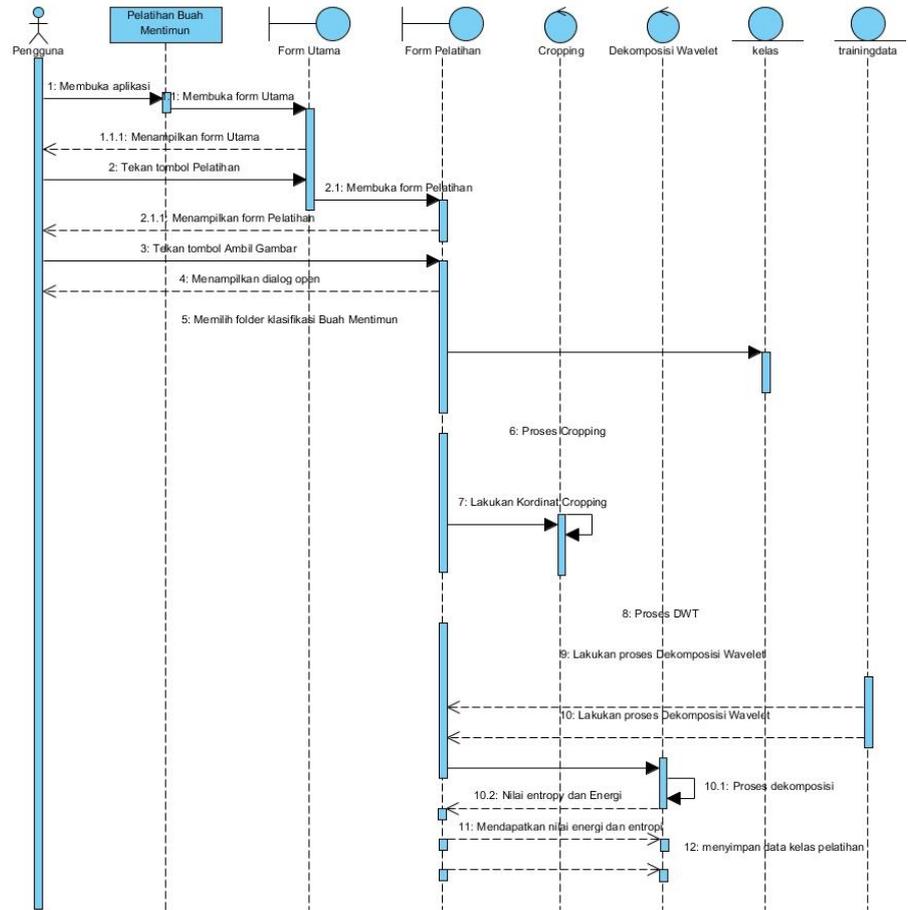
Untuk dapat keluar dari aplikasi, pengguna dapat menekan tombol Keluar. Sistem akan menampilkan konfirmasi keluar dari aplikasi. Apabila pengguna menekan tombol “Ya” maka sistem akan menutup aplikasi, tetapi jika pengguna menekan tombol “Tidak”, maka akan kembali ke *form* utama aplikasi.

3. Sequence Diagram

Sequence Diagram menggambarkan interaksi objek berdasarkan urutan waktu. Selain itu *Sequence* Diagram juga menggambarkan urutan tahapan yang harus dilakukan. *Sequence* Diagram sistem identifikasi kematangan buah mentimun pada tekstur kulit dapat dilihat pada gambar 3.9

a. Sequence Diagram Pelatihan

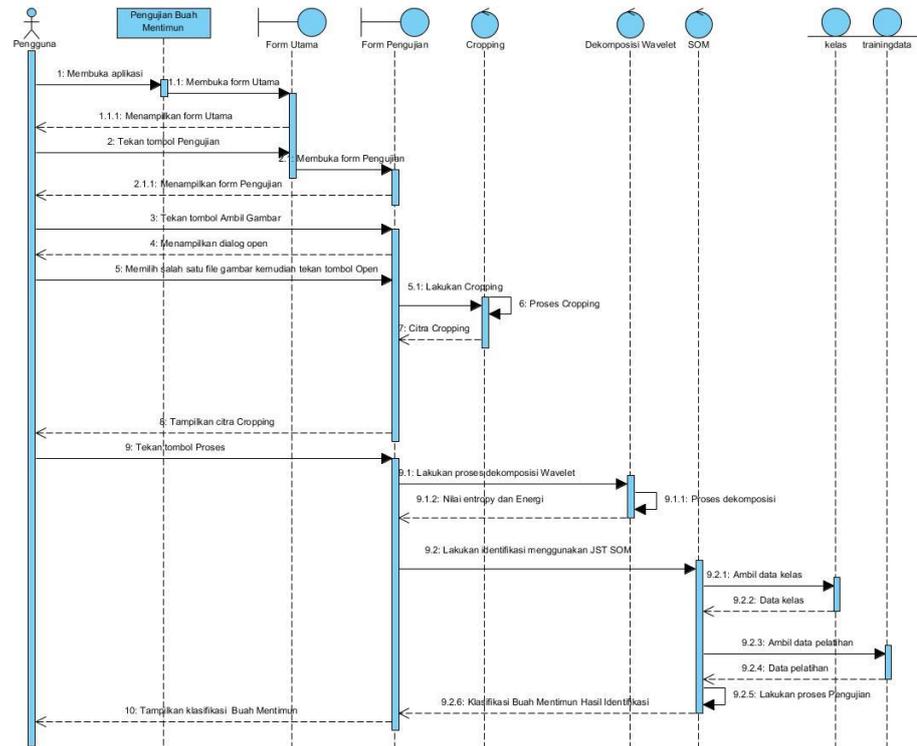
□



Gambar 3.9 Sequence Diagram Pelatihan

Urutan langkah pada proses pelatihan sebagaimana dijelaskan pada *Activity Diagram* Pelatihan.

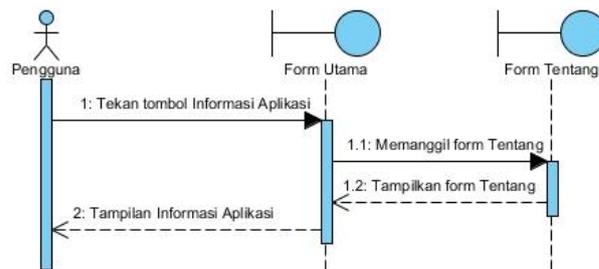
b. Sequence Diagram Pengujian



Gambar 3.10 Sequence Diagram Pengujian

Urutan langkah pada proses pengujian sebagaimana dijelaskan pada *Activity Diagram* Pengujian.

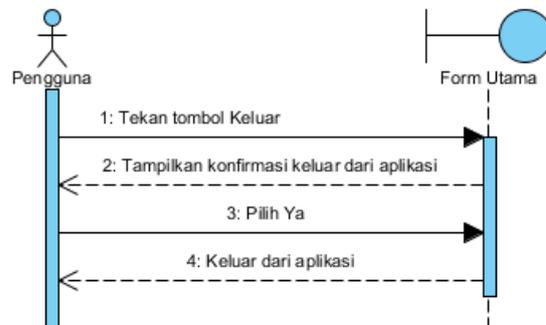
c. Sequence Diagram Tentang



Gambar 3.11 Sequence Diagram Tentang

Urutan langkah pada menampilkan informasi aplikasi sebagaimana dijelaskan pada *Activity Diagram* Tentang.

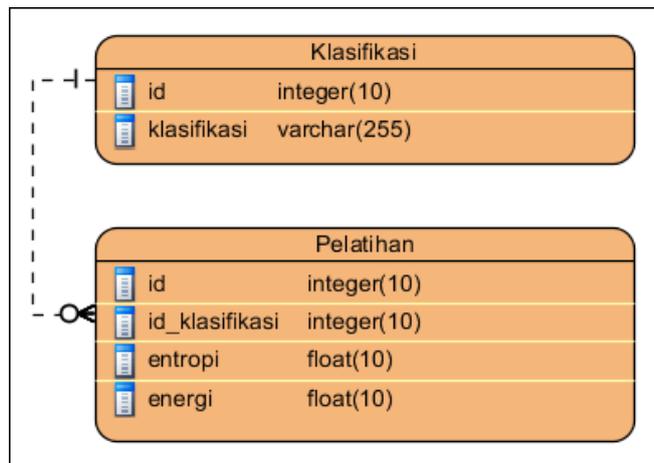
d. *Sequence Diagram* Keluar



Gambar 3.12 *Sequence Diagram* Keluar

Urutan langkah pada saat keluar dari aplikasi sebagaimana dijelaskan pada *Activity Diagram* Keluar.

3.3.3 Perancangan data/arsitektur



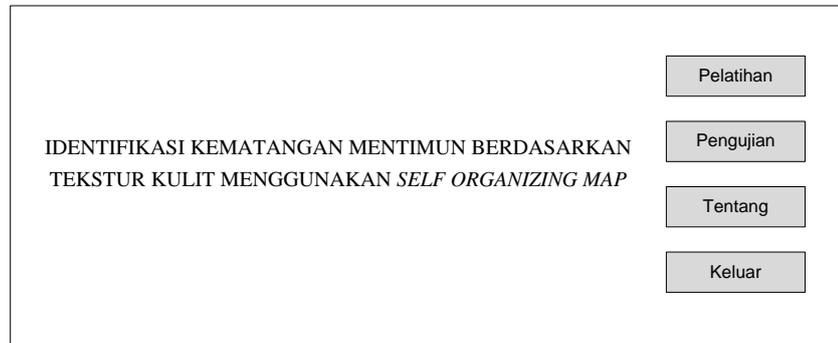
Gambar 3.13 Perancangan Data

Perancangan data yang digunakan pada aplikasi klasifikasi tekstur buah mentimun sangat sederhana. Data yang tersimpan hanya mencakup data klasifikasi dan pelatihan. Sementara struktur jaringan syaraf tiruan disimpan pada *SomToolBox*.

3.3.4 Perancangan antar muka

1. *Form* Utama

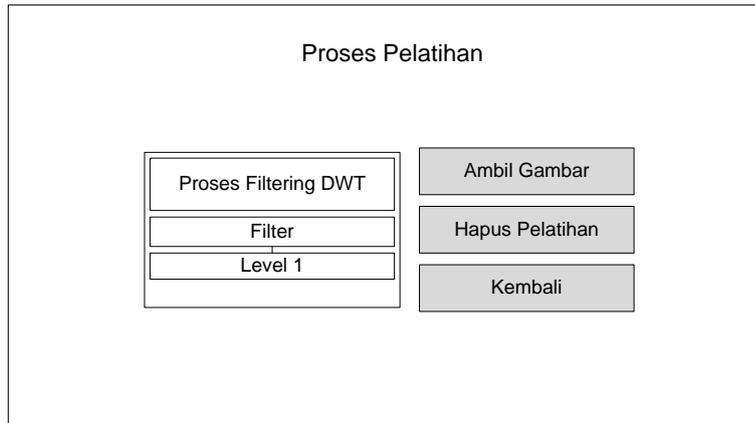
Form Utama merupakan tampilan *form* pelatihan , *form* pengujian, *form* tentang dan *form* keluar.



Gambar 3.14 Rancangan *Form* Utama

2. *Form* Pelatihan

Pada *Form* pelatihan merupakan tampilan *form* tombol ambil gambar dimana gambar citra dari 3 *folder* klasifikasi tekstur buah mentimun yang diinput untuk dilatih, citra yang sudah diinputkan dicrop kordinat menggunakan program bantuan untuk mendapatkan *rec pos* tekstur buah mentimun pada sistem, kemudian pada *Form* Menu merupakan *form* dimana akan dikembalikan ke tampilan menu aplikasi, pada *Form* Proses merupakan *form* pemrosesan citra yang akan dihitung menggunakan transformasi *wavelet* diskrit untuk mendapatkan nilai *energy* dan *entropy* kemudian dilatih menggunakan *Self Organizing Maps*, pada *Form* Hapus Pelatihan dimana citra yang sudah dilatih dari setiap *folder* akan dihapus, setelah pemrosesan hasil nilai yang diperoleh akan disimpan kedalam data basis dan pada tombol *form* keluar untuk keluar dari aplikasi.



Gambar 3. 15 Rancangan *Form* Pelatihan

3. *Form* Pengujian

Pada *Form* Pengujian merupakan tampilan *form* Ambil Gambar digunakan untuk mengambil citra uji penilaian manual, *Form* Proses digunakan untuk *cropping* kordinat sesuai dari pengamatan masing masing tekstur, setelah *system* melakukan *cropping* selanjutnya Tekstur citra buah mentimun asli akan dikonversi menjadi *Grayscale*, dan selanjutnya dilakukan analisi pemfilteran *Wavelete*, *Form* Hasil Perhitungan dan *Form* Hasil Kematangan pada *Form* Proses pengujian merupakan pemrosesan dimana hasil nilai tekstur *entropy* dan *energy* yang didapatkan dari citra uji akan mencari nilai kedekatannya sesuai *range* dari *datatraining.mat* dari hasil pelatihan dari *Form* Pelatihan sebelumnya.

The image shows a testing form titled "Proses Pengujian". It is divided into several sections:

- Top Row:** Three boxes labeled "Gambar Awal", "Cropping", and "Tekstur Grayscale". Below each box is a placeholder labeled "axes1", "axes2", and "axes3" respectively.
- Right Side:** A "Proses Filter" section containing a "Filter" box and a "Level 1" box.
- Bottom Left:** An "Ambil Gambar" button, followed by a "Hasil Perhitungan" box. Below this are input fields for "Entropi" and "Energi".
- Bottom Right:** A "Hasil Kematangan" box, followed by a "Kelas" dropdown menu and a "Waktu" input field.
- Bottom Center:** Two buttons labeled "Proses" and "Kembali".

Gambar 3.16 Rancangan *Form* Pengujian

4. *Form* Tentang

Form tentang merupakan tampilan dari informasi dari aplikasi.

The image shows an "About" form with the following content:

**IDENTIFIKASI KEMATANGAN MENTIMUN BERDASARKAN
TEKSTUR KULIT MENGGUNAKAN *SELF ORGANIZING MAPS***

Informasi Aplikasi

Informasi aplikasi dan cara penggunaan disini.

Keluar

Gambar 3.17 Rancangan *Form* Tentang

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan dapat diambil beberapa kesimpulan sebagai berikut:

- a. Sistem yang dibuat dapat mengenali pola kematangan buah mentimun berdasarkan tekstur nilai *Energy dan Entropy*.
- b. Pada proses teknik pengambilan data mempengaruhi citra uji setengah matang (3) dan citra uji setengah matang (5) hal yang mempengaruhi yaitu citra blur, Citra yang blur mempengaruhi nilai perhitungan tekstur Entropy dan Energy yang dihasilkan, dimana dua citra uji hasil pengujian sistem tidak sesuai dengan pengujian manual. Pengambilan sampel pada proses teknik pengambilan akan sangat mempengaruhi akurasi dan ketepatan dari metode ini yang berdampak pada pengambilan kesimpulan.
- c. Tingkat keakuratan model *SOM (Self Organizing Map)* untuk pendeteksian kematangan buah mentimun berdasarkan fitur tekstur buah menghasilkan akurasi 100% untuk citra uji matang, 66,67% untuk citra uji mentimun setengah matang, dan 100% untuk citra uji mentimun mentah. Dari 18 citra uji pengujian akurasi yang diperoleh 88,89%.

5.2 Saran

Saran yang diberikan untuk pengembangan lebih lanjut dari penelitian ini adalah:

- a. Perlu dilakukan pengembangan deteksi otomatis ketika buah mentimun didekatkan pada kamera.
- b. Peningkatan akurasi dan penajaman .
- c. Perlu dilakukan optimalisasi deteksi tepi dengan segmentasi citra.

DAFTAR PUSTAKA

- Badan Penyuluhan Pengembangan Sumber Daya Manusia Kementrian Pertanian BPPSDMP) 2008. Budidaya dan Klasifikasi Varietas Mentimun.
<http://cybex.pertanian.go.id/1/materipenyuluhan/detail/3455>
- Bahri, Syamsul., Syamsuddin, dan Mustika Hadijati. (2019). Model *Wavelet Neural Network* untuk Pemodelan Polusi Udara di Kota Mataram Lombok Nusa Tenggara Barat Indonesia. Prosiding Seminar Nasional Matematika, Statistika, dan Aplikasinya 2019. Terbitan I, 5 Oktober 2019, Samarinda, Indonesia. e-ISSN: 2657-232X
- Budi S, Aris ., Suma'inna, Hata Maulana. (2016). Pengenalan Citra Wajah Sebagai *Identifier* Menggunakan Metode *Principal Component Analysis* (PCA). Jurnal Teknik Informatika Vol 9 No. 2, Oktober 2016. ISSN 1979-9160
- Cynthia, Eka Pandu., Edi Ismanto. (2017). Jaringan Syaraf Tiruan Algoritma *Backpropagation* dalam Memprediksi Ketersediaan Komoditi Pangan Provinsi Riau. RABIT: Jurnal Teknologi dan Sistem Informasi Univrab Volume 2 No. 2 | Juli 2017 : 83-98. ISSN ONLINE: 2502-891X
- Elha Putri, Shabrina, dkk. (2018). Identifikasi Pola Enamel Gigi Menggunakan Metode *Discrete Wavelet Transform (DWT)* dan *Self organizing maps (SOM)* untuk Aplikasi Forensik Kedokteran Gigi. *e-Proceeding of Engineering* : Vol.5, No.1 Maret 2018. ISSN : 2355-9365
- Fathurrahman, Imam, dkk. (2019). Identifikasi Kematangan Buah Mentimun Berbasis Citra Digital Menggunakan Jaringan Syaraf Tiruan *Backpropagation*. Jurnal Informatika dan Teknologi Vol. 2 No. 1, Januari 2019. e-ISSN 2614-8773
- Hardiansyah, Bagus., dan Puteri Noraisya Primandari. (2018). Sistem Pakar Pengenalan Ekspresi Wajah Manusia Menggunakan *Metode Kohonen Self Organizing dan Principal Componen Analysis*. *Journal of Information Technology*, Vol 3, No 2, 2018. ISSN: 2579-566X ISBN: 978-602-1034-40-8
- Jati, Dwi Wahyuning. (2018). Identifikasi Jenis Jambu Air Berdasarkan Tulang Daun Menggunakan Metode *Learning Vector Quantization*. Program

Studi Teknik Informatika Sekolah Tinggi Manajemen Informatika dan
Komputer Akakom

- Lewi, Eljire Bagas., Inung Wijayanto, Raditiana Patmasari. (2021). Studi Penggunaan Tapis Wavelet untuk Penghilangan Derau Sinyal Sensor Unit Pengukuran Inersia Pada Sistem Pengenalan Postur. *e-Proceeding of Engineering* : Vol. 8, No. 1 Februari 2021. ISSN : 2355-9365
- Maulana, Ivan., Pulung Nurtantio Andono. (2016). Analisa Perbandingan Adaptif Median Filter Dan Median Filter Dalam Reduksi *Noise Salt & Pepper*. *Cogito Smart Journal/VOL. 2/NO. 2/Desember 2016*.
- Muftiany, Auni Aulia. (2018) Klasifikasi Surat Masuk Berdasarkan Program Studi Fakultas Kedokteran Universitas Lambung Mangkurat Banjarmasin Menggunakan Metode *Naive Bayes Text Classifier*
- Nasution, Torkis., Adyanata Lubis. (2021). Implementasi *Watermarking* Pada Gambar Menggunakan Metode *Discrete Wavelet Transform*. Riau Journal Of Computer Science Vol. 3 No. 2 Juli 2017 : 125-138. RJoCS ISSN : 2460-0679
- Pamungkas, Damar Putra. (2019). Ekstraksi Citra menggunakan Metode GLCM dan KNN untuk Identifikasi Jenis Anggrek (*Orchidaceae*). *Innovation in Research of Informatics (INNOVATICS)*. ISSN (*Online*) 2656-8993
- Parhankangas, Juha. Juha Vesanto, Johan Himberg, and Esa Alhoniemi (April, 2000) *SOM Toolbox for matlab 5*. Helsinki University of Technology.ISBN:951-22-4951-0
- Parhankangas, Juha. Juha Vesanto, Johan Himberg, and Esa Alhoniemi (April, 2000) *SOM Toolbox for matlab 5*. Helsinki University of Technology.ISBN:951-22-4951-0
- Pratiwi, Kadek Dwi Diani Shinta, dkk. (2016) *Identifikasi Kematangan Mentimun Berdasarkan Warna Kulit Buah Menggunakan Metode K-Nearest Neighbor*. Universitas Mataram: Teknik Elektro
- Pujiyanta, Ardi., Sovian Rizqiawan (2016) Identifikasi Kematangan Mentimun Berdasarkan Pada Tekstur Kulit Buah dengan *Fuzzy C-Mean*. Seminar Nasional Ilmu Komputer (SNIK 2016) - Semarang, 10 Oktober 2016
- Ramadhan, Miftahur Damar. (2018). Pengolahan Citra untuk Mengetahui Tingkat Kesegaran Ikan Menggunakan Metode Transformasi *Wavelet* Diskrit. Surabaya: Departemen Matematika Fakultas Matematika,

Komputasi, dan Sains Data Institut Teknologi Sepuluh Nopember
Surabaya

- Saputra, Dhanar Intan Surya., Tu Bagus Pranata, Sitaresmi Wahyu Handani. (2016). *Prototype Aplikasi Pengolah Citra Invert Sebagai Media Pengolah Klise Foto. Conference on Information Technology, Information System and Electrical Engineering*. ISBN: 978-602-60280-1-3
- Styawan, Didik., dan Catur Supriyanto. (2018). Pengenalan Plat Nomor Kendaraan dengan *Metode Kohonen SOM*. Universitas Dian Nuswantoro: Teknik Informatika
- Usman, Abdullah. (2016). Sistem Cerdas untuk Klasifikasi Buah Kelapa Menggunakan Metode *Backpropagation*. Selodang Mayang. ISSN: 2442-7845
- Wewo, David M., dkk. (2018). Aplikasi Penentuan Golongan Darah Manusia Dengan Metode *Seed Region Growing Dan Self organizing maps*. J-ICON, Vol. 6 No. 1, Maret 2018. ISSN 2337-7631
- Yanto, Musli. (2017). Penerapan Jaringan Syaraf Tiruan dengan Algoritma *Perceptron* Pada Pola Penentuan Nilai Status Kelulusan Sidang Skripsi. Jurnal Teknoif Vol. 5 No. 2 Oktober 2017. ISSN: 2338-2724
- Zaini, Abdul Hafidh., Bambang Hidayat, Rudy Hartanto. (2019). Deteksi Infeksi Pada Rongga Mulut Berbasis Pemrosesan Sinyal Wicara dengan Metode *Wavelet Packet* dan Klasifikasi *Self Organizing Map (SOM)*. *e-Proceeding of Engineering*: Vol. 6, No. 1 April 2019. ISSN : 2355-9365