

BAB II

LANDASAN TEORI

2.1. Daun Tanaman Obat

2.1.1 Definisi Daun Tanaman Obat

Daun merupakan salah satu organ tumbuhan yang tumbuh dari ranting, umumnya berwarna hijau (mengandung klorofil) dan terutama berfungsi sebagai penangkap energi dari cahaya matahari untuk fotosintesis. Daun merupakan organ terpenting bagi tumbuhan dalam melangsungkan hidupnya karena tumbuhan adalah organisme autotrof obligat, ia harus memasok kebutuhannya sendiri melalui konversi energi cahaya matahari menjadi energi kimia.

Tanaman obat memiliki kegunaan masing-masing termasuk daun. Daun tanaman obat memiliki berbagai macam khasiat untuk menyembuhkan berbagai penyakit yang sudah lama digunakan oleh masyarakat sejak dari dulu.

Kandungan vitamin, mineral dan antioksidannya secara alami menjaga tubuh tetap sehat. Beberapa dari masyarakat yang masih belum tau mengenai kandungan dan fungsi dari daun tanaman yang bisa digunakan sebagai obat penyakit yang kronis.

2.1.2 Jenis Daun Tanaman Obat

Tabel 2.1. adalah jenis daun yang dapat digunakan sebagai obat:

Tabel 2.1. Jenis Tanaman Obat

No	Nama Daun Tanaman Obat
1.	Daun Sirih
2.	Daun Pepaya
3.	Daun Singkong
4.	Daun Alpukat
5.	Daun Jeruk Nipis
6.	Daun Jambu Biji
7.	Daun Cocor Bebek

8.	Daun Binahong
9.	Daun Beluntas
10.	Daun Cermei
11.	Daun Anggur
12.	Daun Waru
13.	Daun Waru
14.	Daun Ubi Jalar
15.	Daun Jambu Biji
16.	Daun Mangga
17.	Daun Cabe
18.	Daun Eceng Gondok
19.	Daun Lidah Buaya
20.	Daun Mengkudu
21.	Daun Nangka
22.	Daun Kemangi
23.	Daun Tapak Dara
24.	Daun Lidah Mertua
25.	Daun Randu
26.	Daun Kangkung
27.	Daun Sukun
28.	Daun Melati
29.	Daun Bayam

Dari jenis daun tanaman obat diatas, skripsi ini hanya mengambil 4 jenis daun yang mudah ditemukan dilingkungan tempat tinggal tetapi masyarakat belum mengetahui kegunaan daun tanaman tersebut. Berikut ini adalah beberapa contoh dari daun tanaman obat yang digunakan dalam Skripsi ini, antara lain :

1. Daun Sirih

Sirih adalah salah satu dari sejumlah tanaman asli indonesia yang memiliki banyak khasiat untuk kesehatan. Tanaman yang

tumbuh merambat pada batang pohon disekelilingnya ini dapat tumbuh dengan subur di wilayah tropis terutama pada tanah dengan kandungan bahan organik dan air yang banyak.

Morfologi / Ciri umum tanaman sirih :

- Tumbuh merambat dengan ketinggian dapat mencapai 15 meter.
- Batang berwarna coklat kehijauan, batang berbentuk bulat, memiliki ruas, bagian ini merupakan bakal tumbuhnya akar.
- Daun sirih berbentuk jantung, tunggal, bagian ujung daun runcing, tumbuh berselang seling, setiap daun memiliki tangkai, bila daun diremas akan mengeluarkan aroma khas, panjang sekitar 5-8 cm dengan lebar sekitar 2-5 cm.
- Bunga sirih majemuk berbentuk bulir, memiliki daun pelindung kurang lebih 1 mm dengan bentuk bulat panjang. Bulir betina memiliki panjang antara 1,5-6 cm. Pada bagian bulir betina ini terdapat kepala putik berjumlah antara 3- 5 buah dengan warna putih dan hijau kekuningan. Bulir jantan memiliki panjang 1,5-3 cm. Pada bulir jantan terdapat dua benang sari yang pendek.
- Akar sirih termasuk akar tunggang dengan bentuk bulat serta warna coklat kekuningan



Gambar 2.1 Daun Sirih

Sumber : [Http://Morfologi dan Kandungan Daun Sirih_TanamanObat.htm](http://Morfologi%20dan%20Kandungan%20Daun%20Sirih_TanamanObat.htm)

2. Daun Randu

Randu atau Kapuk merupakan tanaman tropis dan berasal dari daratan bagian utara Amerika Selatan, Amerika Tengah, dan Karibia.

Pohon tanaman ini mampu tumbuh sampai tinggi 60-70 meter dengan diameter batang mencapai 3 meter. Tanaman ini banyak dibudidayakan di Asia, terutama di pulau Jawa oleh sebab itu tanaman ini juga dikenal sebagai Kapok Jawa.

Sumber : [http// Manfaat Daun Randu, Khasiat Daun Randu, Kegunaan Daun Randu. _ IlmuMu.Com.htm](http://Manfaat Daun Randu, Khasiat Daun Randu, Kegunaan Daun Randu. _ IlmuMu.Com.htm)

Morfologi / Ciri umum tanaman randu :

Ciri-ciri pohon kapuk randu: Batang seperti tiang lurus berduri. Daun sedikit berwarna hijau. Bunga berwarna buram, berbau harum dan indah. Biji mengandung minyak dan bulu biji seperti sutera yang dinamai dengan kapuk. Biji berwarna hijau ketika masih muda dan kulit berwarna coklat setelah randu tua

Sumber : <http://supplykeongmas.blogspot.com/2012/10/pohon-kapuk-randu.html>



Gambar 2.2 Daun Randu

Sumber : <http://supplykeongmas.blogspot.com/2012/10/pohon-kapuk-randu.html>

3. Daun Melati

Melati (*Jasminum sambac*, Ait.) Familia : Oleranceae Melati (*Jasminum sambac*) termasuk tanaman yang mempunyai banyak manfaat. Bunganya berwarna putih mungil dan berbau harum, sering digunakan untuk berbagai kebutuhan. Melati, dapat berbunga sepanjang tahun dan dapat tumbuh subur pada tanah yang gembur.

Sumber : [http// morfologi tumbuhan melati _ Obat Herbal.html](http://morfologi tumbuhan melati _ Obat Herbal.html)

Morfologi / Ciri umum tanaman melati :

Melati dapat digolongkan sebagai semak, bisa juga agak merambat. Melati merambat dengan "berantakan" (terjurai), atau "longgar" ketika masih muda. Batangnya bulat berkayu dengan tinggi 0,3-3 meter. Ia memiliki batang yang bercabang, dan berwarna coklat. Daun melati putih berjenis tunggal, tangkai daun pendek, dengan ukuran sekitar 5 mm, dengan letak yang berhadapan. Helai daunnya berbentuk bulat telur, hingga menjorong, ujungnya runcing, pangkalnya membulat, tepinya rata, tulang daunnya menyirip, dengan ukuran 5-10 cm × 4-6 cm. Perbungaannya termasuk majemuk, tumbuh di ketiak daun, terbatas dengan jumlah 3 bunga atau sebuah tandan padat dengan banyak bunga. Bunganya tunggal atau berpasangan (di varietas kultivasi), dengan 7-10 ruas kelopak, panjang 2,5-7 mm, berbulu halus, panjang tabung mahkota 7-15 mm, sebanyak 5 cuping, bundar telur atau lonjong, panjang 8-15 mm, kebanyakan putih, beraroma kuat. Mahkota bunganya berbentuk lembaran mengerut, seperti terompet, yang berwarna putih, dan berbau wangi. Buahnya termasuk buah buni, mengkilap, dan berwarna hitam, dan dikelilingi kelopak.

Akarnya termasuk tunggang, sulit untuk dipatahkan, -kalaupun dipatahkan-, bekasnya tidak rata, dan juga tidak berserat. Akarnya berbuku-buku/membesar.



Gambar 2.3 Daun Melati

Sumber : http://id.wikipedia.org/wiki/Melati_putih

4. Daun Singkong

Ketela pohon, ubi kayu, atau singkong (*Manihot utilissima*) adalah perdu tahunan tropika dan subtropika dari suku Euphorbiaceae. Umbinya dikenal luas sebagai makanan pokok penghasil karbohidrat dan daunnya sebagai sayuran.

Morfologi / Ciri umum tanaman Singkong :

Perdu, bisa mencapai 7 meter tinggi, dengan cabang agak jarang. Akar tunggang dengan sejumlah akar cabang yang kemudian membesar menjadi umbi akar yang dapat dimakan. Ukuran umbi rata-rata bergaris tengah 2-3 cm dan panjang 50-80 cm, tergantung dari klon/kultivar. Bagian dalam umbinya berwarna putih atau kekuning-kuningan. Umbi singkong tidak tahan simpan meskipun ditempatkan di lemari pendingin. Gejala kerusakan ditandai dengan keluarnya warna biru gelap akibat terbentuknya asam sianida yang bersifat meracun bagi manusia.



Gambar 2.4 Daun Singkong

Sumber : http://id.wikipedia.org/wiki/Ketela_pohon

2.1.3 Manfaat Daun Tanaman Obat

1. Daun Sirih

Manfaat daun sirih yaitu untuk mengobati sakit gigi, gusi bengkak, keputihan, sariawan, demam berdarah, asma, radang

tenggorokan, mimisan, mengobati penyakit kulit, menghilangkan bau mulut, dan bau ketiak.

Sumber : [http// Manfaat Daun Sirih Merah dan Hijau Untuk Kesehatan - Tips Kesehatan.htm](http://Manfaat%20Daun%20Sirih%20Merah%20dan%20Hijau%20Untuk%20Kesehatan%20-%20Tips%20Kesehatan.htm)

2. Daun Randu

Manfaat dari daun randu yaitu digunakan sebagai obat menghilangkan bekas luka, mengobati panas dalam, menyuburkan rambut, dan obat sakit mata.

Sumber : [http// Khasiat Daun Randu.htm](http://Khasiat%20Daun%20Randu.htm)

3. Daun Melati

Manfaat dari daun melati yaitu digunakan sebagai obat untuk mengobati bengkak akibat terkena sengatan lebah, sakit mata, sakit kepala, demam dan sesak nafas.

Sumber : [http// morfologi tumbuhan melati _ Obat Herbal.htm](http://morfologi%20tumbuhan%20melati%20_%20Obat%20Herbal.htm)

4. Daun Singkong

Manfaat dari daun singkong yaitu digunakan sebagai obat anemia, mencegah proses penuaan, meningkatkan daya tahan tubuh, mencegah penyakit tulang, mengobati reumatik, antikanker, mencegah konstipasi, mengobati asam urat, mempercepat pemulihan luka, membantu dalam regenerasi sel tubuh, mengatasi sakit kepala dan mengobati flu.

Sumber : [http// manfaat-daun-singkong-bagi-kesehatan.html](http://manfaat-daun-singkong-bagi-kesehatan.html)

2.2. Preproses Citra Daun

Tanaman dapat diidentifikasi secara fisik berdasarkan bentuk, warna, tekstur dan struktur dari daun, kulit, bunga serta bijinya. Metode pengenalan tanaman berdasarkan bentuk bunga serta biji menjadi langkah yang cukup sulit dalam implementasi karena struktur pola yang digunakan untuk pengenalan

adalah struktur tiga dimensi. Cara paling sederhana yang banyak digunakan dalam mengidentifikasi tanaman adalah berdasarkan bentuk daunnya, yang merupakan representasi citra dua dimensi. Dalam skripsi ini, akan diimplementasikan pengenalan tanaman obat berdasarkan bentuk daunnya. Data daun yang digunakan terdiri 200 citra RGB daun dengan 4 jenis tanaman obat. Data daun tersebut adalah kumpulan gambar daun dengan format jpg.

2.3. Pengantar Citra

Citra disebut juga dengan gambar (Basuki, 2007). Beberapa defenisi dari citra adalah

1. Citra (*Image*) merupakan gambar pada bidang 2(dua) dimensi.
2. Citra (ditinjau dari sudut pandang matematis) merupakan fungsi yang kontinu dari intensitas cahaya pada bidang 2 (dua) dimensi.
3. Citra merupakan sumber cahaya menerangi objek dan dipantulkan kembali dan di tangkap oleh alat-alat optik misalnya mata manusia, kamera scanner, sensor, satelit dan sebagainya kemudian direkam.

Citra sebagai keluaran dari sistem perekam data dapat bersifat :

1. Optik atau photo
2. Analog berupa sinyal video seperti tampilan gambar pada monitor televisi
3. Digital dapat disimpan langsung pada media perekam magnetic

Citra juga dapat dikelompokkan menjadi dua bagian yaitu :

1. Citra tampak (photo, gambar, lukisan, apa yang tampak dilayar monitor/televisi, hologram dll).
2. Citra tidak tampak (data pada photo/gambar pada file, citra yang direpresentasikan dalam fungsi matematis).

Citra digital adalah citra yang disimpan dalam format digital (dalam bentuk file). Hanya citra digital yang dapat diolah menggunakan komputer. Untuk citra non digital sebelum diolah harus dirubah dulu kedalam citra digital.

Pencitraan (*Imaging*) adalah merupakan kegiatan mengubah informasi dari citra tampak atau citra non digital menjadi citra digital. Beberapa alat yang dapat digunakan untuk pencitraan adalah scanner, kamera digital dll.

Pengolahan Citra adalah kegiatan memproses suatu citra sehingga menghasilkan citra yang sesuai dengan keinginan kita atau kualitasnya menjadi lebih baik. (Basuki, 2007)

Umumnya, operasi-operasi pengolahan citra diterapkan pada citra bila :

1. Perbaiki atau modifikasi citra untuk meningkatkan kualitas visual atau menonjolkan beberapa aspek informasi yang terkandung dalam citra.
2. Elemen di dalam citra perlu di kelompokkan, dicocokkan atau diukur.
3. Sebagian citra perlu di gabung dengan bagian citra yang lain.

Terdapat empat kelompok algoritma dasar pengolahan citra :

1. Kelompok Titik (mengubah nilai pixel berdasarkan nilai pixel yang asli)
2. Kelompok Bidang (mengubah nilai pixel berdasar nilai pixel aslinya dan berdasarkan nilai pixel tetangganya)
3. Kelompok Geometris (mengubah posisi atau susunan pixel)
4. Kelompok Frame (membangkitkan nilai-nilai pixel berdasarkan operasi-operasi pada pada 2 citra atau lebih)

Analisis Citra merupakan kegiatan menganalisis sehingga menghasilkan informasi untuk menetapkan keputusan (biasanya didampingi bidang ilmu kecerdasan buatan / artificial intelligent yaitu pengenalan pola (pattern recognition) menggunakan jaringan syaraf tiruan, fuzzy logic dll. (Basuki, 2007)

2.4. Pengolahan Citra Digital

2.4.1. Citra Digital

Sebuah citra digital adalah kumpulan piksel-piksel yang disusun dalam larik dua dimensi. Indeks baris dan kolom (x,y) dari sebuah piksel yang dinyatakan dalam bilangan bulat dan nilai-nilai tersebut mendefinisikan suatu ukuran intensitas cahaya pada titik tersebut. Satuan atau bagian terkecil dari suatu citra disebut piksel (picture element).

Umumnya citra dibentuk dari persegi empat yang teratur sehingga jarak horizontal dan vertikal antara piksel satu dengan yang lain adalah sama pada seluruh bagian citra. Piksel $(0,0)$ terletak pada sudut kiri atas pada citra, dimana indeks x bergerak ke kanan dan indeks y bergerak ke bawah. Untuk menunjukkan koordinat $(m-1,n-1)$ digunakan posisi kanan bawah dalam citra berukuran $m \times n$ pixel. Hal ini berlawanan untuk arah vertikal dan horizontal yang berlaku pada sistem grafik dalam matematika.

Video sebenarnya juga merupakan salah satu bentuk dari citra digital. Video merupakan kumpulan dari beberapa frame dari citra digital. Satu frame merupakan satu citra digital. Oleh karena itu, apabila pada video mempunyai satuan yang menyatakan 12 fps(frame per second) artinya bahwa video tersebut memainkan 12 frame/citra digital dalam tiap detiknya. (Suhendra, 2006)

2.4.2. Jenis Citra Digital

1. Citra RGB

Suatu citra biasanya mengacu ke citra RGB. Sebenarnya bagaimana citra disimpan dan dimanipulasi dalam komputer diturunkan dari teknologi televisi, yang pertama kali mengaplikasikannya untuk tampilan grafis komputer. Jika dilihat dengan kaca pembesar, tampilan monitor komputer akan terdiri dari sejumlah triplet titik warna merah (RED), hijau (GREEN) dan biru (BLUE). Tergantung pada pabrik monitornya untuk menentukan apakah titik tersebut merupakan titik bulat atau kotak kecil, tetapi akan selalu terdiri dari 3 triplet red, green dan blue.

Citra dalam komputer tidak lebih dari sekumpulan sejumlah triplet dimana setiap triplet terdiri atas variasi tingkat keterangan (brightness) dari elemen red, green dan blue. Representasinya dalam citra, triplet akan terdiri dari 3 angka yang mengatur intensitas dari Red (R), Green (G) dan Blue (Blue) dari suatu triplet. Setiap triplet akan merepresentasikan 1 pixel (picture element). Suatu triplet dengan nilai 67, 228 dan 180 berarti akan mengeset nilai R ke nilai 67, G ke nilai 228 dan B ke nilai 180. Angka-angka RGB ini yang seringkali disebut dengan color values.

Sumber : <http://digilib.itelkom.ac.id>

2. Citra YCbCr

YCbCr merupakan standar internasional bagi pengkodean digital gambar televisi yang didefinisikan di CCIR Recommendation. Y merupakan komponen luminance, Cb dan Cr adalah komponen chrominance. Pada monitor monokrom nilai luminance digunakan untuk merepresentasikan warna RGB, secara psikologis ia mewakili intensitas sebuah warna RGB yang diterima oleh mata. Chrominance merepresentasikan corak warna dan saturasi (saturation). Nilai komponen ini juga mengindikasikan banyaknya komponen warna biru dan merah pada warna. YCbCr (256 level) dapat diperoleh dari RGB 8 bit dengan menggunakan rumus berikut:

$$Y = 0.299 R + 0.587 G + 0.114 B \dots\dots\dots(2.1)$$

$$Cb = -0.1687 R - 0.3313 G + 0.5 B + 128 \dots\dots\dots(2.2)$$

$$Cr = 0.5 R - 0.4187 G - 0.0813 B + 128 \dots\dots\dots(2.3)$$

Sedangkan untuk konversi YCbCr ke RGB dapat dilakukan dengan rumus:

$$R = Y + 1.402 (Cr-128) \dots\dots\dots(2.4)$$

$$G = Y - 0.34414 (Cb-128) - 0.71414 (Cr - 128) \dots\dots\dots(2.5)$$

$$B = Y + 1.772 (Cb - 128) \dots\dots\dots(2.6)$$

Sumber : <http://digilib.itelkom.ac.id>

3. Citra Biner

Citra biner (*binary image*) adalah citra yang hanya mempunyai dua nilai derajat keabuan yaitu hitam dan putih. Alasan masih digunakannya citra biner dalam pengolahan citra digital hingga saat ini adalah algoritma untuk citra biner telah berkembang dengan baik dan waktu pemrosesan lebih cepat karena jumlah bit untuk tiap pikselnya lebih sedikit.

Konversi Citra ke Biner bertujuan untuk mempermudah dalam proses pengolahan citra sehingga akan didapatkan citra yang berbentuk hitam dan putih. Proses konversi citra ke biner sebenarnya masih memiliki kesamaan pada proses konversi citra ke greyscale.

Citra daun, pada umumnya berwarna hijau. Namun, terkadang akibat bayangan yang tertangkap kamera menyebabkan perubahan warna pada citra daun. Selain itu, perbedaan kandungan air, nutrisi, atmosfer, lingkungan, serta cuaca dapat juga menyebabkan perbedaan warna pada daun, walaupun daun-daun tersebut berasal dari spesies yang sama atau bahkan dari tanaman yang sama.

Fitur yang digunakan dalam skripsi ini adalah fitur bentuk daun (*Digital Morphological Feature*), yang didapatkan dari citra biner. Sehingga langkah selanjutnya adalah merubah citra daun dengan skala keabuan (grayscale) yang telah didapatkan menjadi citra biner. Citra biner merupakan citra yang telah melalui proses pemisahan piksel-piksel berdasarkan derajat keabuan yang dimiliki. Pembentukan citra biner memerlukan nilai batas keabuan yang akan digunakan sebagai nilai patokan.
Sumber : <http://digilib.ittelkom.ac.id>

2.4.3. Image Enhancement

Teknik image enhancement digunakan untuk meningkatkan kualitas suatu citra digital, baik dalam tujuan untuk menonjolkan suatu ciri tertentu dalam citra tersebut, maupun untuk memperbaiki aspek tampilan. Proses ini biasanya didasarkan pada prosedur yang bersifat eksperimental, subjektif, dan amat bergantung pada tujuan yang hendak dicapai. (Basuki, 2007)

1. Grayscale

Salah satu proses pengolahan citra adalah mengkonversi sebuah citra dari warna menjadi keabuan. Beberapa alasan yang dikemukakan oleh Stephens untuk dilakukan konversi keabuan adalah :

- a. Citra keabuan umumnya terlihat lebih baik pada resolusi warna yang rendah dari pada resolusi yang dimiliki citra berwarna. Untuk sebuah citra yang memiliki jumlah warna lebih dari 256 warna, sebuah sistem mungkin saja menampilkan warna dengan titik-titik kecil yang mendekati warna tersebut karena sistem tidak dapat menampilkan warna aslinya. Hal ini dapat menyebabkan tekstur citra yang tidak seperti aslinya. Sedangkan citra dengan tingkat warna 256 keabuan, sistem warna 8 bit akan menampilkan hasil yang halus.
- b. Citra keabuan membuat banyak proses pengolahan citra menjadi lebih mudah untuk dimengerti. Contohnya untuk memperjelas garis pada sebuah citra. Hasil dari citra keabuan akan lebih mudah dimengerti daripada sebuah citra berwarna.

Untuk mengubah citra ke bentuk keabuan yaitu menghitung rata-rata piksel RGB, dengan menjumlahkan nilai piksel 3 (tiga) warna dasar yaitu red, green, dan blue kemudian dibagi 3 (tiga). Algoritma grayscale adalah :

1. Ambil warna piksel citra (W)
2. Grayscale = konversi_ke_integer ((merah+hijau+biru)/3)

Algoritma *grayscale* yang lain adalah dengan menghitung nilai piksel masing-masing warna (RGB), yaitu : $(0299*R + 0587*G + 0114*B) \dots (2.7)$ (Aminuddin, 2011)

2. Threshold

Untuk sebuah citra hasil pindaian harus diubah terlebih dahulu tingkat keabuannya sebelum dimanipulasi agar objek menjadi lebih jelas.

Thresholding digunakan untuk mengatur jumlah derajat keabuan yang ada pada citra. Dengan menggunakan thresholding maka derajat keabuan

bisa diubah sesuai keinginan, misalkan diinginkan menggunakan derajat keabuan 16, maka tinggal membagi nilai derajat keabuan dengan 16. Proses thresholding ini pada dasarnya adalah proses pengubahan kuantisasi pada citra, sehingga untuk melakukan thresholding dengan derajat keabuan dapat digunakan rumus :

$$x = b \cdot (w/b) \quad (2.8)$$

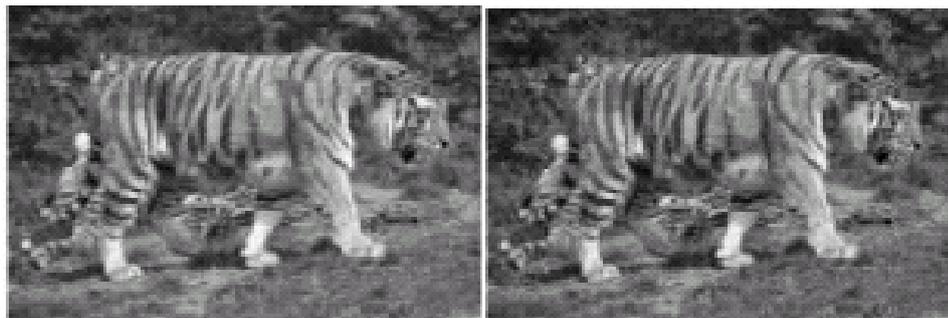
dimana :

w adalah nilai derajat keabuan sebelum thresholding

x adalah nilai derajat keabuan setelah thresholding

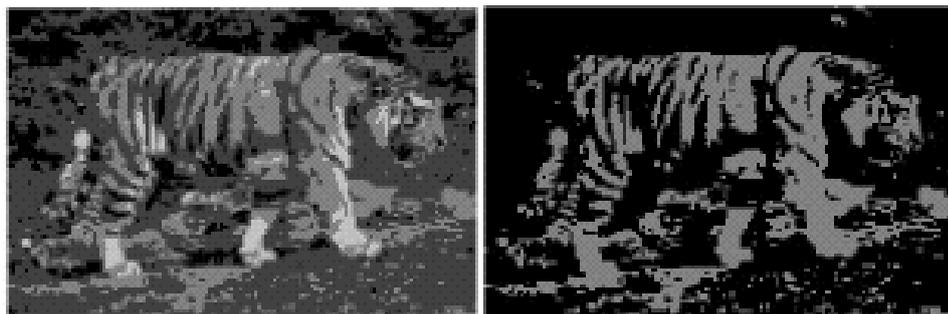
$$b = \text{int.} (256/a) \quad (2.9)$$

Berikut ini contoh thresholding mulai di 256, 16, 4 dan 2 :



256 Grayscale

16 Grayscale



4 Grayscale

2 Grayscale

Gambar 2.5 Contoh Tresholding

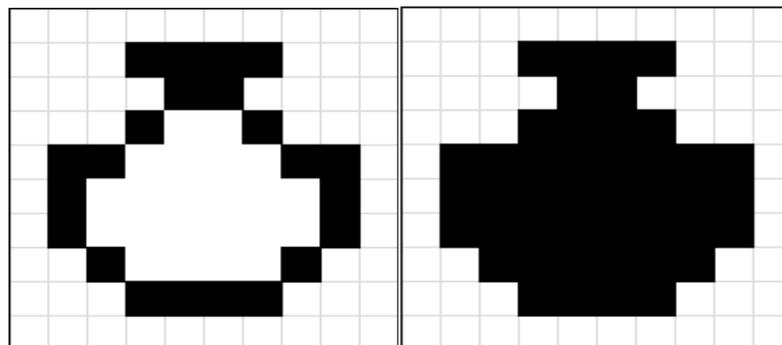
Untuk mencoba melakukan proses thresholding, perlu dibuat program untuk dapat mengubah-ubah nilai thresholding sesuai keinginan. Sehingga perlu ditampilkan dua citra, yaitu citra asli (gray-scale) dan hasil thresholdingnya dengan nilai thresholding yang ditentukan melalui input seperti terlihat pada gambar 2.5. (Aminuddin, 2011)

3. Adaptive Histogram Equalization (CLAHE)

Hasil dari citra aras keabuan masih memiliki kekontrasan yang rendah dan detail pada daun tanaman obat masih kurang jelas sehingga akurasi yang dihasilkan kurang baik. Oleh karena itu citra tersebut harus ditingkatkan dengan menggunakan Ekualisasi Histogram Adaptif (*Adaptive Histogram Equalization*) atau pada program matlab disebut CLAHE (*Contrast-Limited Adaptive Histogram Equalization*) menggunakan perintah `adapthisteq`. Cara ini bertujuan untuk mendapatkan hasil citra dengan kontras yang lebih baik tetapi tanpa mengurangi kualitas dari citra tersebut. (Suhendra, 2006)

4. Filling (Pengisian)

Operasi pengisian merupakan kebalikan dari operasi pencarian batas citra. Pada operasi ini, citra masukan adalah citra batas/kontur, kemudian dilakukan pengisian sehingga diperoleh segmen objek yang solid. Prosesnya dimulai dengan menentukan titik awal pengisian yang terletak di dalam objek, kemudian bergerak ke arah titik-titik tetangganya.



Gambar 2.6 Proses *Image Filling*

			1	1	1	1					1	1	1	1					
				1	1							1	1						
				1	0	0	1					1	1	1	1				
	1	1	0	0	0	0	1	1		1	1	1	1	1	1	1	1		
	1	0	0	0	0	0	0	1		1	1	1	1	1	1	1	1		
	1	0	0	0	0	0	0	1		1	1	1	1	1	1	1	1		
			1	0	0	0	1				1	1	1	1	1	1	1		
			1	1	1	1					1	1	1	1					

Gambar 2.7 Pixel Biner Proses *Image Filling*

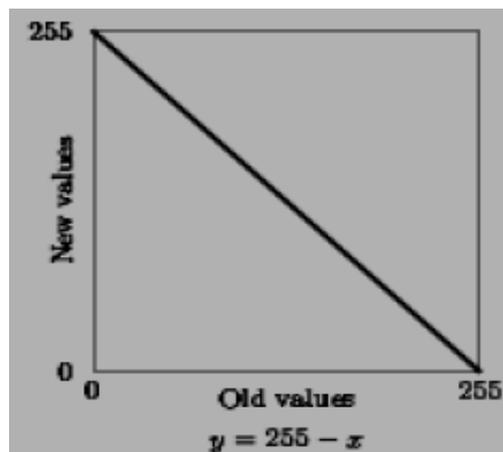
(Sakinah, 2011)

2.4.4. Komplemen

Komplemen dari sebuah citra aras keabuan adalah negatif dari citra tersebut (*photographic negative*). Fungsi untuk menghasilkan komplemen sebuah citra aras keabuan adalah

$$y = L - x \quad (2.10)$$

dengan L adalah aras atau level tertinggi piksel (255) dan x adalah nilai piksel citra input. Grafik pada gambar 2.10 berikut menggambarkan fungsi komplemen sebuah citra aras keabuan.



Gambar 2.8 Fungsi komplemen sebuah citra aras keabuan

(Rifa, 2014)

2.4.5. Morfologi

Morfologi adalah teknik pengolahan citra digital dengan menggunakan bentuk (shape) sebagai pedoman dalam pengolahan. Nilai dari setiap pixel dalam citra digital hasil diperoleh melalui proses perbandingan antara pixel yang bersesuaian pada citra digital masukan dengan pixel tetangganya. Operasi morfologi bergantung pada urutan kemunculan dari pixel, tidak memperhatikan nilai numeric dari pixel sehingga teknik morfologi sesuai apabila digunakan untuk melakukan pengolahan binary image dan grayscale image.

Operasi morfologi banyak digunakan dalam pengolahan dan analisis citra misalkan untuk operasi perbaikan citra (image enhancement) , ekstraksi fitur, deteksi tepi, analisis bentuk, dan beberapa implementasi operasi pengolahan citra lain.

Dalam operasi morfologi, pemilihan structuring element (strel) sangat mempengaruhi hasil pemrosesan citra. Penggunaan dua buah structuring element yang berbeda akan menghasilkan hasil yang berbeda juga meski objek/citra yang dianalisa sama.

Ada beberapa bentuk structuring element (SE) yang biasa digunakan, ada yang berbentuk rectangle, square, disk, linear, dan diamond. Setiap bentuk structuring element (SE) tersebut memiliki kelebihan dan kekurangan masing-masing. Structuring element berbentuk rectangle dan square, dapat digunakan untuk mendeteksi tepi bagian atas, bawah, pinggir kiri, dan kanan dari sebuah objek. Sedangkan structuring element berbentuk disk dapat digunakan untuk melakukan operasi dilasi/rotasi yang tidak berhubungan dengan arah karena structuring element berbentuk disk simetris terhadap objek aslinya. Structuring element berbentuk line/linear hanya dapat mendeteksi single border.

Belum ada pedoman dalam pemilihan bentuk structuring element. Umumnya pemilihan bentuk structuring element hanya didasarkan pada kemiripan dengan bentuk objek yang diteliti. Salah satu atribut yang penting untuk mengenali sebuah objek adalah shape (bentuk). Bentuk

merupakan representasi dari sebuah objek. Shape (bentuk) adalah salah satu atribut yang penting untuk mengenali sebuah objek. Pemilihan bentuk structuring element lebih didasarkan pada kemiripan dengan bentuk objek. Oleh karena itu bentuk objek dapat digunakan sebagai penentuan bentuk structuring element.

1. Operasi Dasar Morfologi

a. Dilasi

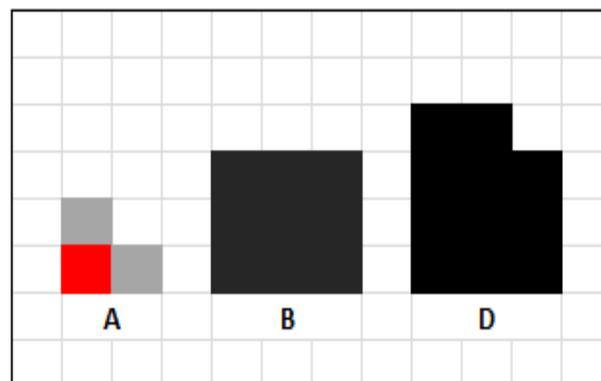
Dilasi adalah suatu proses menambahkan piksel pada batasan dari objek dalam suatu gambar sehingga nantinya apabila dilakukan operasi ini maka gambar hasilnya lebih besar ukurannya dibandingkan dengan gambar aslinya. Operasi dilasi akan melakukan proses pengisian pada citra asal yang memiliki ukuran lebih kecil dibandingkan structuring element (strel).

Dilasi A oleh B dinotasikan dengan $A \oplus B$ dan didefinisikan sebagai:

$$D(A,B) = A \oplus B = \{x : B_x \cap A \neq \emptyset\} \quad (2.11)$$

Dengan \emptyset menyatakan himpunan kosong.

Gambar 2.9 menunjukkan proses operasi dilasi ,terdapat objek awal A dan B sedangkan objek D objek hasil dilasi.



Gambar 2.9 Proses Dilasi

b. Erosi

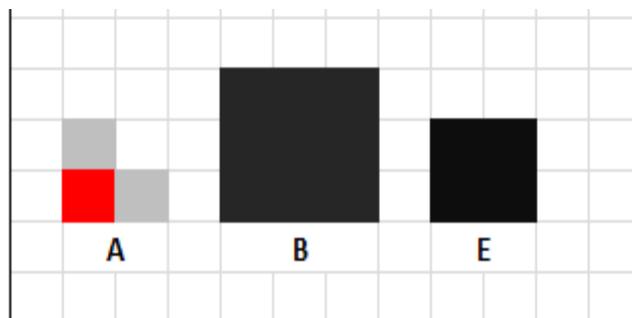
Operasi erosi merupakan kebalikan dari operasi dilasi. Pada operasi ini, ukuran objek diperkecil dengan mengikis sekeliling objek. sehingga citra hasil cenderung diperkecil menipis. Operasi erosi akan melakukan pengurangan pada citra asal yang lebih kecil dibanding elemen penstruktur (strel).

Erosi A oleh B dinotasikan $A \ominus B$ didefinisikan sebagai :

$$E(A,B) = A \ominus B = \{ x : Bx \subset X \} \quad (2.12)$$

Sama seperti dilasi, proses erosi dilakukan dengan membandingkan setiap piksel citra input dengan nilai pusat SE dengan cara melapiskan SE dengan citra sehingga SE tepat dengan posisi piksel citra yang diproses.

Gambar 2.10 menunjukan proses operasi dilasi ,terdapat objek awal A dan B sedangkan objek E objek hasil erosi.



Gambar 2.10 Proses Erosi

Sumber:<http://abdullahbasuki.files.wordpress.com>

c. Opening (Pembukaan)

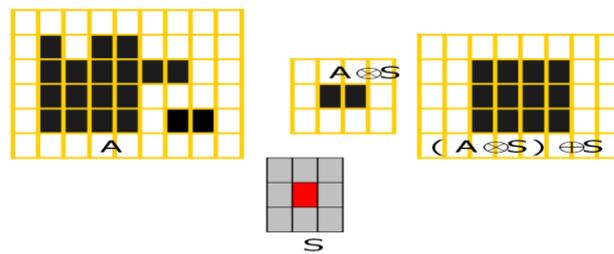
Operasi opening (pembukaan) juga merupakan kombinasi antara operasi erosi dan dilasi yang dilakukan secara berurutan, tetapi citra asli dierosi terlebih dahulu baru kemudian hasilnya didilasi. Operasi ini digunakan untuk memutus bagian-bagian dari objek yang hanya terhubung dengan 1 atau 2 buah titik saja.

Secara matematis prose opening dapat dinyatakan dengan :

$$O(A,B) = A \circ B = D(E(A,B),B) \quad (2.13)$$

Operasi opening digunakan untuk memutus bagian-bagian dari objek yang hanya terhubung dengan 1 atau 2 buah titik saja, dan menghilangkan objek yang sangat kecil. Operasi opening bersifat memperhalus kenampakan citra, menyambung fitur yang terputus (break narrow joins), dan menghilangkan efek pelebaran pada objek (remove protrusions).

Gambar 2.11 menunjukan proses operasi dilasi, terdapat objek awal A dan S.



Gambar 2.11 Proses Operasi Opening

Sumber: <http://abdullahbasuki.files.wordpress.com/>

d. Closing (Penutupan)

Operasi closing (penutupan) adalah kombinasi antara operasi dilasi dan erosi yang dilakukan secara berurutan [SRI]. Citra asli didilasi terlebih dahulu, kemudian hasilnya dierosi. Operasi ini digunakan untuk menutup atau menghilangkan lubang-lubang kecil yang ada dalam segmen objek. Operasi penutupan juga digunakan untuk menggabungkan 2 segmen objek yang saling berdekatan (menutup sela antara 2 objek yang sangat berdekatan).

Operasi closing dapat dinyatakan sebagai berikut:

$$C(A,B) = A \bullet B = E(D(A, -B), -B) \quad (2.14)$$

Hasil operasi closing hampir mirip seperti hasil operasi dilasi yakni memperbesar batas luar dari objek foreground dan juga menutup lubang kecil yang terletak di tengah objek, namun hasil operasi closing tidak sebesar hasil dilasi.



Gambar 2.12 Proses Operasi Closing

Sumber: <http://abdullahbasuki.files.wordpress.com/>

e. Rekonstruksi

Rekonstruksi adalah transformasi morfologi yang melibatkan dua citra dan sebuah strel (sebenarnya adalah sebuah citra dan sebuah strel). Citra yang satu adalah *marker*, adalah *starting point* transformasi. Citra yang lain adalah *mask*, yang merupakan constrain transformasi. Strel digunakan untuk mendefinisikan konektivitas. Dalam sub-bab ini yang digunakan adalah 8-connectivity (default), yang mengimplikasikan bahwa B adalah matriks 3 x 3 bernilai 1, dengan pusat yang didefinisikan di koordinat (2,2).

Jika g adalah mask dan f adalah marker, rekonstruksi g dari f, dinyatakan $R_g(f)$, didefinisikan dengan prosedur iteratif berikut:

1. Inisialisasi h_1 menjadi marker f.
2. Buat strel $B = \text{ones}(3)$.
3. Ulangi

$$H_{k+1} = (h_k \oplus B) \cap g \quad (2.15)$$

sampai $h_{k+1} = h_k$

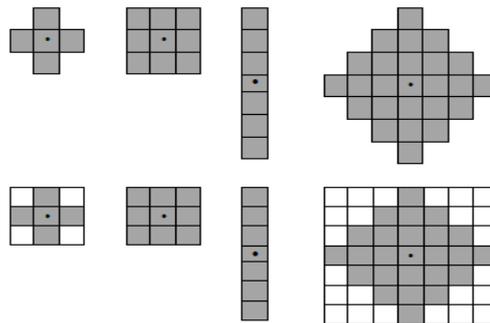
Algoritma ini berjalan dengan lebih cepat meskipun menggunakan formulasi iteratif. (Prasetyo,2011)

2. Structure Element

Struktur Element adalah himpunan sub-image kecil yang digunakan untuk meneliti citra dalam pembelajaran propertinya.

Untuk elemen yang menjadi anggota strel, original strel, juga harus ditetapkan.

Origin dari strel ditandai dengan tanda titik hitam. jika tidak ada titik hitam maka diasumsikan origin berada di pusat simetri. karena origin tidak harus berada di pusat, tetapi juga bisa berada di pinggir strel.



Gambar 2.13 Contoh Gambar strel

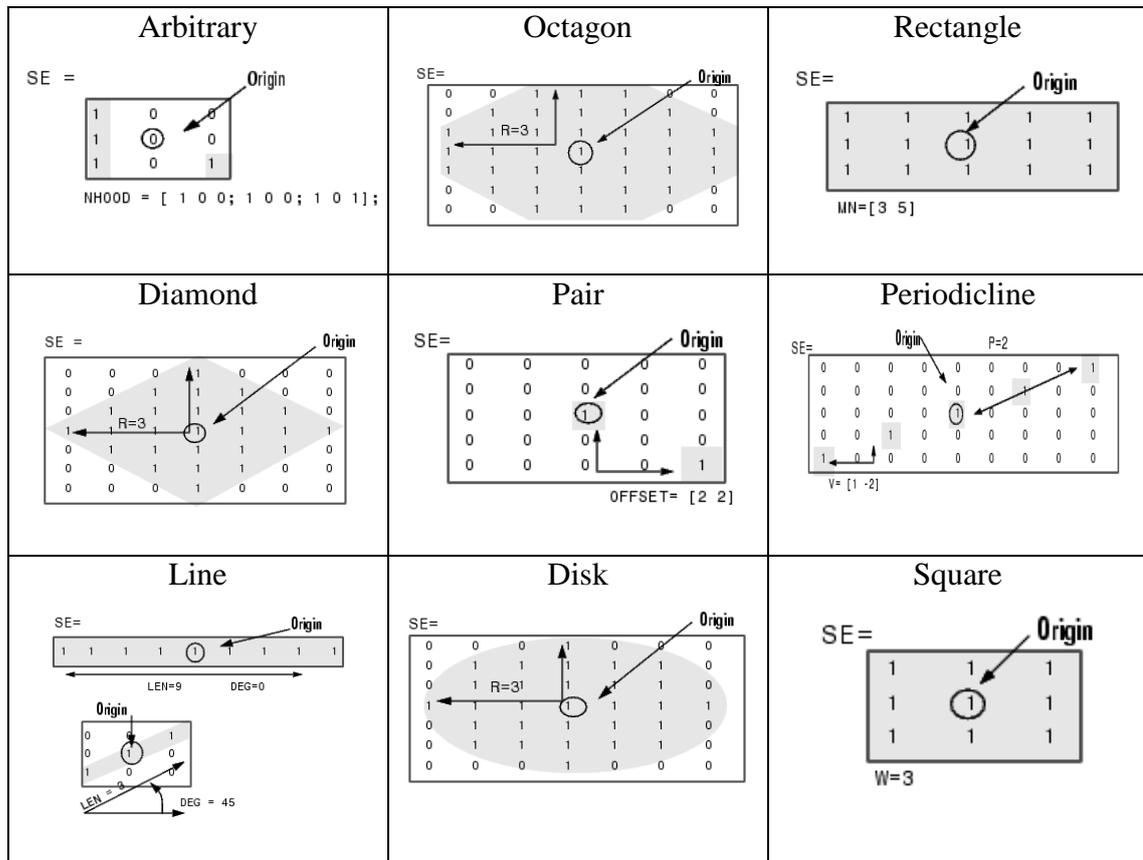
Pada gambar 2.14 menunjukkan berbagai macam type yang dapat digunakan, dan pada gambar 2.14 menjelaskan dari berbagai macam type tersebut.

SE = strel(tipestrel, parameter)

Toolbox MATLAB untuk membuat strel :

<u>Tipe</u>	<u>Format fungsi</u>
Arbitrary	SE = strel('arbitrary', NHOOD)
Diamond	SE = strel('diamond', R)
Disk	SE = strel('disk', R, N)
Line	SE = strel('line', LEN, DEG)
Octagon	SE = strel('octagon', R)
pair	SE = strel('pair', OFFSET)
periodicline	SE = strel('periodicline', P, V)
rectangle	SE = strel('rectangle', MN)
square	SE = strel('square', W)

Gambar 2.14 Tipe dari Structure Element (SE)



Gambar 2.15 Penjelasan dari masing-masing SE

2.5. Deskriptor Bentuk

Deskriptor bentuk adalah teknik untuk mempresentasikan bentuk objek, sebuah representasi yang baik akan dapat menggambarkan karakteristik intrinsik dari sebuah shape secara eksplisit. Representasi sebuah shape juga harus invariant terhadap rotasi, scaling dan transformasi.

Klasifikasi yang digunakan dalam skripsi ini adalah descriptor bentuk, terdapat 4 point yang dianalisis pada descriptor bentuk ini, yaitu:

1. Area

Area adalah jumlah piksel dalam S, sehingga bila dalam suatu citra terdapat lebih dari satu komponen, S_1, S_2, \dots, S_n maka ada A_1, A_2, \dots, A_n . Jadi nilai area suatu objek adalah jumlah dari piksel-piksel penyusun objek tersebut dan unit yang umum digunakan adalah piksel karena sejumlah piksel membentuk suatu luasan. Area dapat

mencerminkan ukuran atau berat objek sesungguhnya pada beberapa benda dengan bentuk yang hampir seragam (misalnya buah mangga) tetapi tidak demikian untuk benda yang berongga (misalnya paprika)

2. Perimeter

Perimeter merupakan bagian terluar dari suatu objek yang bersebelahan dengan piksel-piksel dari latar belakang. Nilai perimeter suatu objek dapat dicari dengan menghitung banyaknya piksel yang merupakan piksel-piksel yang berada pada perbatasan dari objek tersebut.

3. Indeks Kebulatan

Faktor kebulatan dapat digunakan untuk menggolongkan bentuk objek yang dihubungkan dengan bentuk bundar dan memanjang seperti banyak dijumpai pada jenis buah-buahan. Lingkaran adalah bentuk rasio paling bundar dengan nilai resiko sekitar 1, untuk bujur sangkar nilainya $\pi/4$ semakin mengecil ketika objek berbentuk memanjang (elips atau kotak). Untuk mendapatkan luas suatu objek dilakukan dengan cara mencari indek kebulatan dengan rumus :

$$\frac{4 \cdot \pi A}{p^2} \quad (2.16)$$

Luas lingkaran dengan panjang perimeter P adalah dapat dinyatakan dengan rumus :

$$\frac{p^2}{4\pi} \quad (2.17)$$

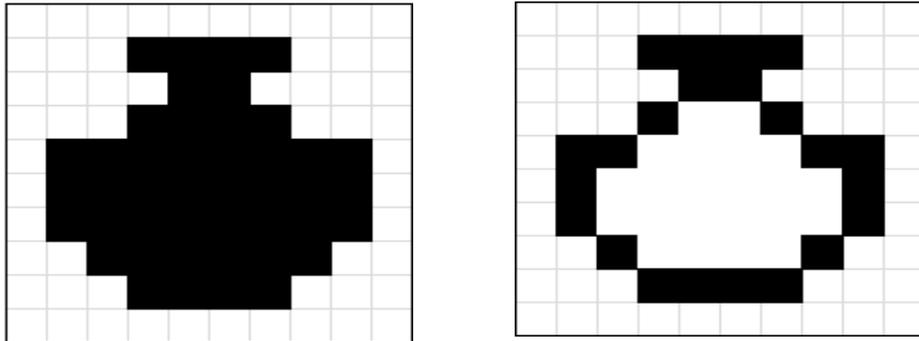
4. Compactness

Indeks kepadatan (Compactness indeks/CI) adalah pengukuran bentuk pembatas yang paling populer yang mengetimasi kebulatan obejek 2D. namun pengukuran ini sangat sensitive terhadap noise disepanjang pembatasan atau tepi area daun yang nilainya dikuatkan oleh kuadrat.

$$\frac{p^2}{4 \pi A} \quad (2.18)$$

Hasil perhitungan compactness digunakan sebagai acuan dalam uji coba implementasi software. Gambar 2.16 merupakan contoh image yang mempunyai area, perimeter, dan kebularan:

(Rifa, 2014)



Gambar 2.16 Bentuk objek mempunyai area, perimeter, dan kebularan.

A = Jumlah piksel dibaris ke-1 + baris ke-2 + ... + baris ke-8

$$= 4+2+4+8+8+8+6+4$$

$$= 44 \text{ piksel}$$

P = jumlah piksel dari batas area

$$= 24 \text{ piksel}$$

$$R = \frac{4 \cdot \pi A}{p^2} = \frac{4 \cdot 3,14 \cdot 44}{24^2} = 0,96$$

$$C = \frac{p^2}{4 \pi A} = \frac{24^2}{4 \cdot 3,14 \cdot 44} = 1,04$$

Ket :

A = Area

P = Perimeter

R = Kebularan

C = Compactness

2.6. Jaringan Syaraf Buatan

2.6.1. Pengertian Jaringan Syaraf Buatan

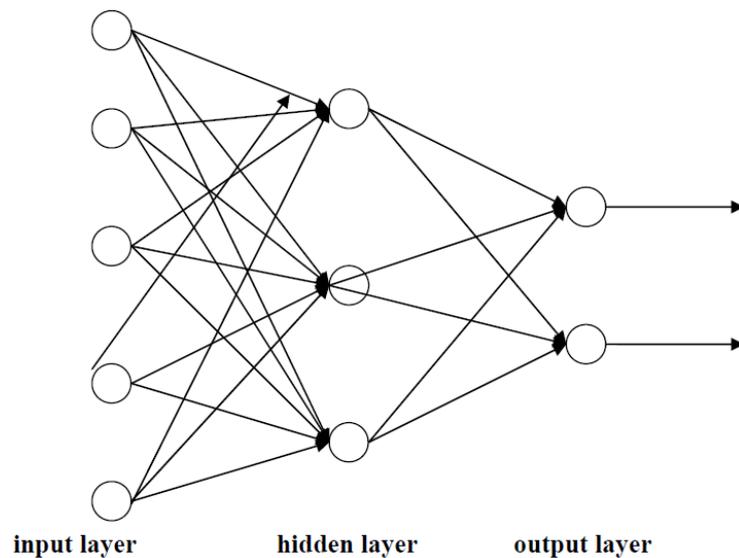
Jaringan syaraf tiruan dibangun berdasarkan jaringan syaraf manusia. Bagaimana canggihnya jaringan syaraf tiruan yang dibangun, masih jauh dari setara dengan jaringan syaraf manusia, mungkin hanya dapat dibandingkan oleh sistem jaringan hewan sederhana.

Jaringan syaraf buatan adalah sekelompok elemen pengolah sedemikian rupa sehingga sebuah sub kelompok membuat komputasi yang berbeda dan melewatkan hasilnya kepada sub kelompok berikutnya, demikian seterusnya hingga tiba pada sebuah sub kelompok akhir yang terdiri dari sebuah atau lebih elemen pengolah yang akan menentukan keluaran (output) dari jaringan syaraf tersebut.

Setiap elemen pengolah membuat perhitungannya didasarkan pada jumlah terbobot (*weighted sum*) dari inputnya. Elemen pengolah tersebut merupakan satuan yang mirip dengan neuron dalam otak manusia, yang sering disebut sebagai neuron buatan, yang seterusnya akan disebut neuron saja.

Sebuah sub kelompok pengolah disebut sebuah layer dalam jaringan, dimana lapisan pertama disebut input layer, lapisan terakhir disebut sebagai output layer, sedangkan lapisan-lapisan diantara lapisan masukan dengan lapisan keluaran disebut sebagai hidden layers. Terdapat pula sebuah fungsi aktivasi (*activation function*) yang digunakan untuk menentukan keluaran sebuah neuron menuju lapisan selanjutnya. Sinapsis akan menghubungkan setiap neuron, sinapsis direpresentasikan sebagai yang simpul-simpulnya merupakan neuron-neuron.

Gambar 2.17 menunjukkan jaringan syaraf buatan sederhana dalam bentuk grafik. Dalam gambar ditunjukkan representasi jaringan syaraf buatan yang terdiri dari 3 lapisan yaitu satu lapisan masukan, satu lapisan tersembunyi dan satu lapisan keluaran.



Gambar 2.17 Jaringan syaraf buatan

Jaringan syaraf buatan ini dapat dilatih untuk menyimpan, mengingat, mencocokkan suatu pola tertentu. Pola yang digunakan mengandung digit biner (0 dan 1) untuk kasus dimana masalah berupa discrete atau bilangan-bilangan real yang mewakili signal analog untuk kasus yang kontinu.

Jaringan syaraf buatan ini dapat dibuat sebagai perangkat keras atau dapat pula direpresentasikan dengan bentuk perangkat lunak. Namun jaringan syaraf buatan direpresentasikan dalam bentuk perangkat keras lebih cepat daripada bentuk perangkat lunak. Dalam skripsi ini tidak akan digunakan bentuk representasi dalam perangkat keras, namun cukup berupa perangkat lunak dengan algoritma pembelajaran backpropagation. (Kusumadewi, 2003)

2.6.2. Fungsi Aktivasi dan Signal-signal

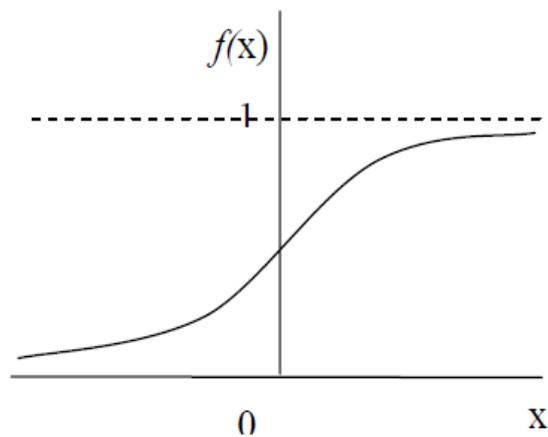
Neuron akan berkelakuan analog seperti fungsi yaitu menerima masukan dan menghasilkan signal keluaran. Pada bagian ini akan dibahas fungsi aktivasi dan signal-signal yang akan dihasilkan.

Aktivasi dari neuron adalah jumlah terbobot dari masukan sebuah neuron dalam jaringan syaraf buatan. Sebagai gambaran, pada saat t , aktivasi masukannya dinotasikan sebagai $x(t)$, maka dari aktivasi tersebut neuron akan menghasilkan signal keluaran $S(t)$, dimana untuk melakukan hal tersebut dibutuhkan fungsi aktivasi. Banyak macam fungsi aktivasi, mulai dari fungsi linear, sigmoid, step,

ramp, linear, dll. Fungsi aktivasi yang sering digunakan adalah fungsi sigmoid yang berbentuk seperti huruf S. fungsi sigmoid diantaranya adalah fungsi tangen hyperbolic yang range nilainya diantara -1 dan 1, dan fungsi logistic yang range nilainya antara 0 dan 1 (Kusumadewi, 2003). Kedua fungsi tersebut dapat ditulis sebagai berikut:

Fungsi logistic 1

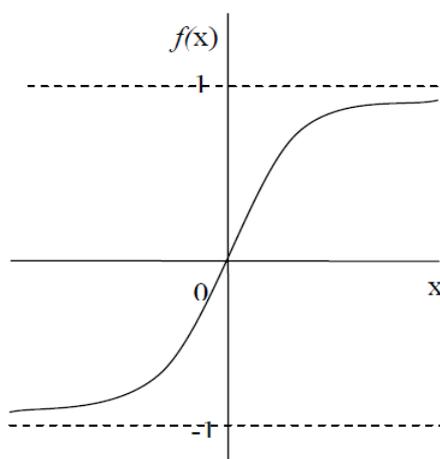
$$f(x) = \frac{1}{1+e^{-x}} \quad (2.19)$$



Gambar 2.18 Fungsi Sigmoid Logistik

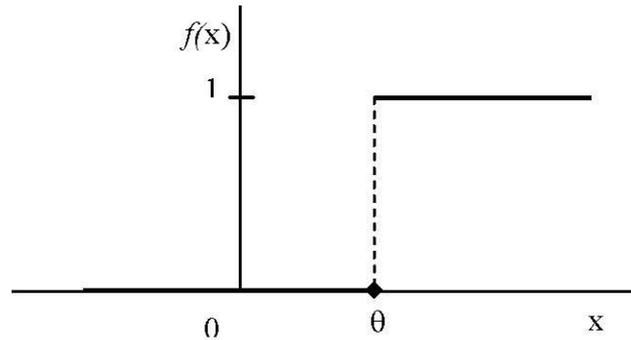
Fungsi tangen hiperbolik

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.20)$$



Gambar 2.19 Fungsi Sigmoid Hiperbolik

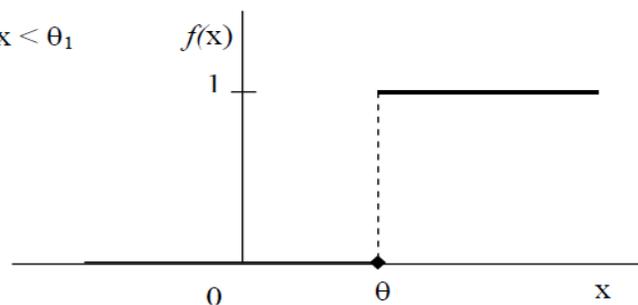
Fungsi step bekerja sebagai berikut, fungsi diawali dengan 0 dan terus berada posisi 0 hingga mencapai nilai ambang θ . Lompatan (step) pertama terjadi untuk nilai-nilai yang berada di sebelah kanan θ , dan nilainya terus bertahan di posisi 1. Untuk aktivasi bernilai sama dengan θ , signal keluaran ($S(t)$) dapat bernilai 1 atau 0, tergantung dari jaringan syaraf buatan yang digunakan



Gambar 2.20 Fungsi Step

Fungsi Ramp mempunyai kemiripan dengan fungsi step, perbedaannya adalah perubahan antara 0 menuju 1 tidak dilakukan dengan tiba-tiba pada satu titik, namun dilakukan secara perlahan. Fungsi tersebut dapat dituliskan sebagai berikut:

$$f(x) = \begin{cases} 1 & \text{jika } x > \theta_2 \\ \alpha x - \theta_1 & \text{jika } \theta_1 \leq x \leq \theta_2 \\ 0 & \text{jika } x < \theta_1 \end{cases}$$



Gambar 2.21 Fungsi Ramp

Fungsi linear juga dapat digunakan sebagai fungsi aktivasi, yang mempunyai bentuk umum seperti:

$$f(x) = \alpha \cdot x + \beta \quad (2.21)$$

Jika $\alpha = 1$ berarti ambang fungsi ini akan sama dengan menambahkan nilai bias yang sama dengan nilai beta untuk jumlah masukannya.

(Kusumadewi, 2003)

2.7. Jaringan Backpropagation

2.7.1. Sejarah Perkembangan Backpropagation

Jaringan *backpropagation* merupakan jaringan perceptron multi lapisan atau multi layer perceptrons tetapi dengan algoritma pembelajaran yang berbeda.

Jaringan *backpropagation* dibuat karena perceptron mempunyai kelemahan. Kejayaan perceptrons berakhir setelah beredar buku berjudul Perceptrons yang ditulis oleh Minsky dan Papert pada tahun 1969. Dalam buku tersebut dikatakan bahwa sebagian besar masalah tidak memberikan klasifikasi yang konvergen secara linier, bahkan perceptron terkadang tidak mampu memecahkan masalah-masalah yang sederhana, seperti operasi XOR, dimana diketahui 2 masukan biner, dan akan dihasilkan nilai 1 jika dan hanya jika satu masukan hidup (bernilai 1), selain itu akan dihasilkan nilai 0. Perceptron tidak mampu memecahkan masalah tersebut karena masalah tersebut tidak terpisah secara linier.

Ternyata masalah ini dapat dipecahkan dengan menggunakan perceptron multi lapisan, yaitu rangkaian perceptron dimana sedemikian sehingga keluaran perceptron pertama berlaku sebagai masukan perceptron pada lapisan berikutnya. Tetapi algoritma pembelajaran perceptron dianggap tidak mencakup perceptron multi lapisan. Pada saat buku tersebut diterbitkan, cara pembelajaran multi lapisan pada perceptron belum ditemukan, sampai akhirnya pada tahun 1980-an ditemukan suatu cara pembelajaran pada jaringan syaraf buatan multilapisan.

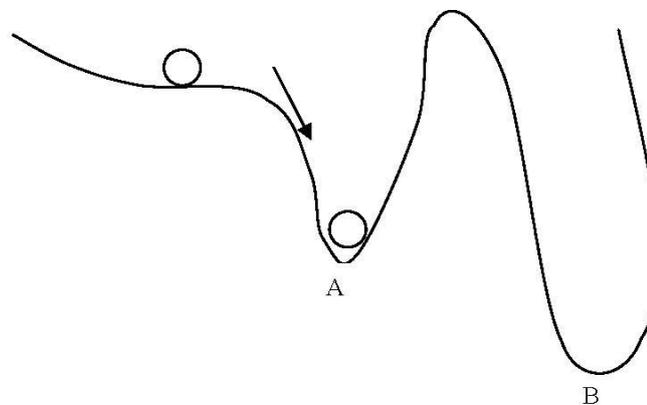
Rumelhart et al. mempopulerkan algoritma *backpropagation* dalam volume *Parallel Distributed Processing* pada akhir 1980-an. Mereka menyarankan penggunaan algoritma ini, yang mereka sebut aturan delta tergeneralisasi (*generalized delta rule*) untuk mengatasi keterbatasan algoritma perceptron. Backpropagation adalah keberhasilan bagi *connectionism*, yang merupakan istilah yang digunakan oleh ilmuwan komputer dan kognitif untuk teori dan aplikasi jaringan syaraf tiruan.

Backpropagation segera mendominasi dunia jaringan syaraf buatan karena kemudahannya dalam pelatihan sistem. Diperkirakan lebih dari 80% dari semua proyek sistem syaraf buatan dikembangkan dengan menggunakan algoritma pelatihan backpropagation.

Popularitas backpropagation tidak luput dari kritikan karena dikatakan algoritma ini belajar dari informasi nonlokal, sehingga sering mencapai kegagalan konvergensi, dan cenderung membawa ke arah kesalahan minima lokal, bukan global. Algoritma ini memodifikasi bobot-bobot koneksinya dengan informasi kesalahan nonlokal, dimana nonlokalitas ini menyebabkan pelatihan menjadi sinkron, lama dan menyimpang dari kemasuk-akalan biologis.

Beberapa ilmuwan mengatakan algoritma backpropagation pasti membawa informasi ke arah yang konvergen dengan argumen karena algoritma ini mengimplementasikan gradient descent pada error dalam vektor ruang weight antar koneksi lapisannya. Oleh karena itu mereka mengatakan pasti akan mencapai titik konvergen dimana tercapai kesalahan minimal. Pada kenyataannya pada simulasi diskret, algoritma ini sering kali tidak konvergen, bahkan terjadi osilasi, dan hanya mencapai kesalahan minimal pada keadaan lokal (*local minima*) bukan global.

Ilmuwan memperkenalkan beberapa metode untuk mengatasi *local minima* ini. Salah satunya adalah dengan memberikan faktor momentum m dimana akan menjaga supaya perubahan weight bergerak pada arah yang sama sedemikian sehingga algoritma akan melompat *local minima*. (Aminuddin, 2011)



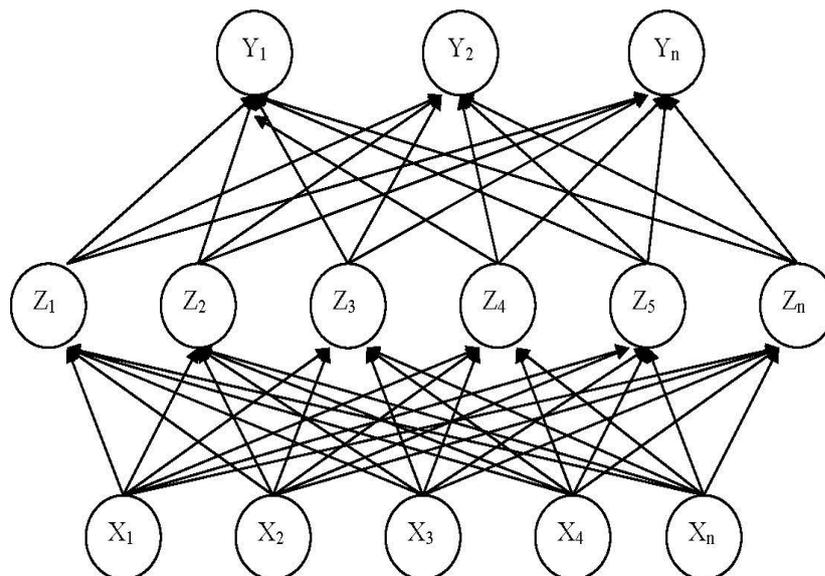
Gambar 2.22 *Local Minima (A), Global Minima (B)*

Menurut rich and Knight (1001, p507). Algoritma backpropagation jarang mencapai local minima karena pada jaringan yang relatif besar, vektor ruang weight yang berdimensi tinggi menyediakan banyak tingkat kebebasan bagi algoritma sehingga teorema berkurangnya konvergensi bukanlah masalah dalam praktek nyatanya. Keistimewaan yang melegakan ini ditemukan baru-baru ini, ketika komputer digital cukup cepat untuk mendukung proses simulasi jaringan syaraf buatan berskala besar. (Aminuddin, 2011)

2.7.2. Arsitektur Jaringan Backpropagation

Backpropagation adalah sistem syaraf buatan yang berkoneksi penuh, berlapis, dan berumpan maju (*feedforward*). Contoh jaringan backpropagation yang sederhana terdapat pada gambar 2.23.

Jaringan *backpropagation* dapat mempunyai banyak lapisan tersembunyi (*hidden layer*), namun pada contoh pada gambar 2.23 hanya dilukiskan arsitektur jaringan backpropagation dengan satu hidden layer. Jumlah neuron masukan dan neuron keluaran berturut-turut ditentukan dari dimensi pola masukan dan pola keluaran. Sedangkan untuk lapisan tersembunyi jumlah neuron tidak dapat ditentukan dengan mudah.



Gambar 2.23 Jaringan Backpropagation Sederhana

Jaringan ini mengalami pelatihan dengan pengawasan (*supervised*), dengan jumlah tertentu pasangan pola masukan dan pola keluaran yang diharapkan. Pola masukan dimasukkan pada lapisan masukan, yang kemudian akan melewati digit-digit pola ke neuron-neuron pada lapisan selanjutnya, yang merupakan lapisan tersembunyi. Keluaran dari lapisan tersembunyi diperoleh dengan menggunakan nilai bias dan fungsi aktivasi dengan nilai masukan ke aktivasi ditentukan oleh bobot-bobot dan masukan. Keluaran tersebut menjadi masukan bagi lapisan keluaran, yang akan mengolah keluaran akhir jaringan dengan menggunakan nilai bias yang berbeda dengan fungsi aktivasi yang sama.

Pola yang telah dihitung dan pola masukan dibandingkan dengan suatu fungsi kesalahan untuk setiap komponen pola, kemudian dilakukan penyesuaian bobot koneksi (*weight*) di antara lapisan tersembunyi (*hidden layer*) dan lapisan keluaran (*output layer*). Perhitungan yang sama dilakukan untuk bobot-bobot koneksi di antara lapisan masukan (*input layer*) dan lapisan tersembunyi. Proses ini diulangi sebanyak yang diperlukan hingga kesalahan berada dalam toleransi kesalahan yang diinginkan. Setiap prosedur tersebut diulangi untuk setiap pasangan pola yang diberikan ke jaringan. (Aminuddin, 2011)

2.7.3. Algoritma Backpropagation

- Objektif : untuk menyesuaikan bobot-bobot koneksi sehingga diperoleh hasil keluaran yang sesuai dengan toleransi kesalahan dari user.
- Diketahui : himpunan pasangan vektor masukan dan keluaran.
- Dihitung : himpunan bobot jaringan syaraf buatan dengan tiga lapisan yang akan memetakan vektor masukan ke vektor keluaran yang bersesuaian.

Algoritma Backpropagation:

1. Inialisasi bobot koneksi
2. Lakukan selama kondisi berhenti belum terpenuhi,

Feedforward:

3. Untuk setiap unit masukan (X_i , $i = 1, \dots, n$) akan menerima signal masukan yang akan diteruskan menuju lapisan berikutnya (*hidden layer*).
4. Untuk setiap unit tersembunyi (Z_j , $j = 1, \dots, p$) akan menjumlahkan setiap bobot koneksi dengan signal dari lapisan masukan.

$$z_inj = V_0j + \sum_{i=1}^n X_i V_{ij} \quad (2.22)$$

Masukan hasil perhitungan ke dalam fungsi aktivasi guna menghasilkan nilai keluaran:

$$z_j = \int(z_inj) \quad (2.23)$$

Lalu hasil perhitungan fungsi aktivasi akan dikirim sebagai signal masukan bagi semua unit di lapisan keluaran (*output layer*).

5. Untuk setiap unit keluaran (Y_k , $k = 1, \dots, m$) akan menjumlahkan setiap bobot koneksi dengan masukan signal dari lapisan tersembunyi.

$$y_in_k = W_{0k} + \sum_{j=1}^p Z_j W_{jk} \quad (2.24)$$

Lalu masukkan hasil perhitungan ke dalam fungsi aktivasi untuk mendapatkan signal keluaran:

$$y_k = \int(y_in_k) \quad (2.25)$$

Backpropagation of Error:

6. Untuk setiap unit keluaran (y_k , $k = 1, \dots, m$) akan menerima pola keluaran yang sebenarnya (t_k) sesuai dengan pola masukan yang telah diproses, lalu akan dihitung kesalahan antara pola keluaran sebenarnya dengan pola keluaran dari sistem syaraf buatan.

$$\delta_k = (t_k - y_k) \cdot f'(y_in_k) \quad (2.26)$$

Lalu dihitung nilai bobot antara lapisan tersembunyi dengan lapisan keluaran yang baru:

$$w_{jk} = w_{jk}(\text{old}) + (\alpha \cdot \delta_k \cdot z_j) \quad (2.27)$$

Lalu dihitung nilai bias lapisan keluaran yang baru:

$$w_{0k} = w_{0k}(\text{old}) + (\alpha \cdot \delta_k) \quad (2.28)$$

7. Untuk setiap unit tersembunyi ($Z_j, j = 1, \dots, p$) akan menjumlahkan setiap delta masukan dari lapisan keluaran.

$$\delta_{inj} = \sum_{k=1}^m \delta_k W_{jk} \quad (2.29)$$

Untuk mendapatkan nilai kesalahan dari lapisan tersembunyi kalikan hasil di atas dengan turunan dari fungsi aktivasi.

$$\Delta_j = \delta_{inj} \cdot f'(z_{inj}) \quad (2.30)$$

Lalu dihitung nilai bobot antara lapisan tersembunyi dengan lapisan masukan yang baru:

$$v_{ij} = v_{ij}(\text{old}) + (\alpha \cdot \delta_j \cdot x_i) \quad (2.31)$$

Lalu dihitung nilai bias lapisan tersembunyi yang baru:

$$v_{0j} = v_{0j}(\text{old}) + (\alpha \cdot \delta_j) \quad (2.32)$$

8. Uji, apakah hasil kesalahan sudah memenuhi kriteria toleransi kesalahan.

Algoritma tersebut digunakan apabila jaringan backpropagation yang dipakai adalah jaringan dengan 3 lapisan. Sedangkan untuk jaringan lebih dari 3 lapisan masih diperlukan beberapa langkah tambahan, yaitu mengulang langkah feedforward dan langkah backpropagation untuk menyesuaikan bobot pada langkah no.7.

Fungsi aktivasi untuk algoritma backpropagation berbentuk sigmoid. Fungsi ini dibutuhkan karena turunan aturan penyesuaian bobot koneksi membutuhkan fungsi aktivasi yang bersifat kontinu dan bisa terdiferensiasi. Dengan fungsi sigmoid bobot yang dibutuhkan agar jaringan dapat mencapai 0 dan 1 adalah tak terhingga, maka untuk hasil biner, vektor keluarannya hanya akan memberikan nilai yang mendekati 0 atau 1, tidak tepat sama dengan nilai-nilai tersebut.

(Kusumadewi, 2003)