Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

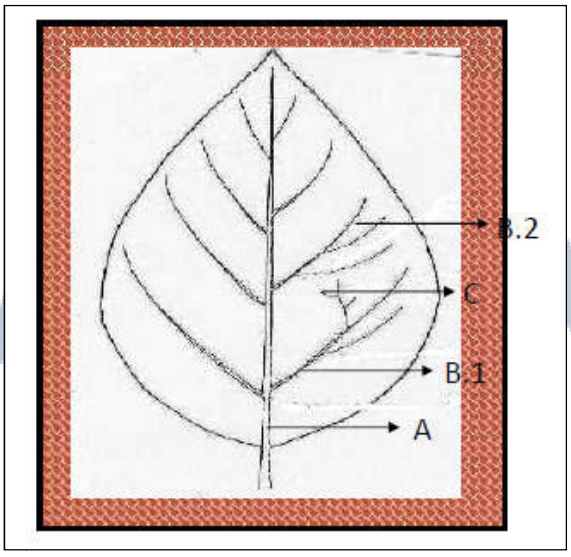
Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II
LANDASAN TEORI

2.1 Morfologi Daun

Morfologi merupakan salah satu cabang ilmu yang mempelajari bentuk, susunan, dan fungsi dari suatu aspek ilmu. Pada morfologi daun sebenarnya dapat dibedakan menjadi dua yaitu morfologi luar yang mempelajari dan mengamati bentuk fisik, maupun morfologi dalam, yang mempelajari bagian terdalam dari daun dengan menggunakan alat bantu mikroskopik (Warnita dkk., tanpa tahun). Daun merupakan bagian terpenting pada tumbuhan, yang umumnya melekat pada batang dan dahan. Fungsi utama daun pada tumbuhan adalah sebagai organ yang berperan sebagai penyerap, pengangkut, pengolah, dan penimbunan zat-zat makanan.



Gambar 2.1 Morfologi tulang daun
(Sumber : Warnita dkk., tanpa tahun)

Pada Gambar 2.1 terlihat bahwa daun tersusun atas berbagai macam tulang daun. A menunjukkan ibu tulang daun (*costa*), yang merupakan tulang utama yang langsung berhubung dengan pangkal daun.

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

B merupakan tulang-tulang cabang (*nervus lateralis*), B.1 dan B.2 menunjukkan tingkatan cabangnya. C adalah urat-urat daun (*vena*) yang merupakan bagian terkecil dari struktural tulang daun (Warnita dkk, tanpa tahun). Tulang daun tidak hanya berfungsi sebagai rangka daun, tetapi juga sebagai wadah untuk pengangkut unsur hara dari dalam tanah dan hasil fotosintesis.

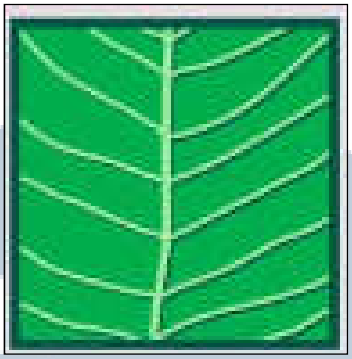
Namun, ternyata berbagai tanaman tidak selalu mempunyai rupa tulang yang sama, walau pada dasarnya mempunyai tiga macam tulang dan fungsi yang sama pula. Maka dari itu muncul istilah "*Leaf venation*" atau pola tulang daun. Pola tulang daun dapat dibedakan menjadi lima macam (Geneve, tanpa tahun), yaitu:

2.1.1 Pinnate

Pola tulang daun *pinnate* memiliki ciri tulang cabang bersumber dari ibu tulang daun. Di kedua sisi ibu tulang daun terdapat tulang cabang dari pangkal hingga ujung daunnya. Gambar 2.2 menunjukkan salah satu contoh daun *pinnate* dan Gambar 2.3 menunjukkan struktur pola tulang daun *pinnate* jika diperbesar.

Gambar 2.2 Contoh tulang daun *pinnate*
(Sumber: Geneve, tanpa tahun)

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016



Gambar 2.3 Pola tulang daun *pinnate*
(Sumber : Nix, tanpa tahun)

2.1.2 Palmate

Pola tulang daun *palmate* memiliki ciri ditandai dengan adanya suatu titik yang merupakan awal mula percabangan dari beberapa tulang ibu daun yang akan menyebar dengan tulang cabangnya masing-masing hingga ke ujung daun. Gambar 2.4 menunjukkan salah satu contoh daun *palmate* dan Gambar 2.5 menunjukkan struktur pola tulang daun *palmate* jika diperbesar.

Gambar 2.4 Contoh tulang daun *palmate*
(Sumber: Geneve, tanpa tahun)

UNIVERSITAS MULTIMEDIA NUSANTARA

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

Gambar 2.5 Pola tulang daun *palmate*
(Sumber: Nix, tanpa tahun)

2.1.3 Reticulate

Pola *reticulate* dapat terlihat dari tulang cabang daun dan di sekitar permukaan daunnya terdiri dari serabut-serabut tipis yang berjumlah sangat banyak dan terkadang saling terhubung. Gambar 2.6 menunjukkan salah satu contoh daun *reticulate* dan Gambar 2.7 menunjukkan struktur pola tulang daun *reticulate* jika diperbesar.

Gambar 2.6 Contoh tulang daun *reticulate*
(Sumber: Geneve, tanpa tahun)

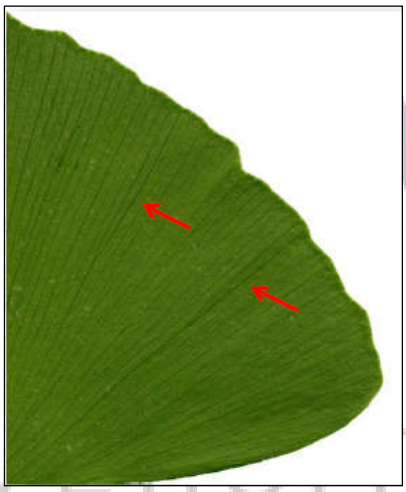
UNIVERSITAS MULTIMEDIA NUSANTARA

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

Gambar 2.7 Pola tulang daun *reticulate*
(Sumber: Nix, tanpa tahun)

2.1.4 Dichotomous

Pola *dichotomous* biasanya hanya terdiri dari banyak tulang ibu yang menjulur lurus ke atas dan akan bercabang berbentuk Y (terbelah) saat mendekati ujung daun. Gambar 2.8 menunjukkan salah satu contoh daun *dichotomous* dan Gambar 2.9 menunjukkan struktur pola tulang daun *dichotomous* jika diperbesar.



Gambar 2.8 Contoh tulang daun *dichotomous*
(Sumber: Geneve, tanpa tahun)

UNIVERSITAS MULTIMEDIA NUSANTARA

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

Gambar 2.9 Pola tulang daun *dichotomous*
(Sumber: Nix, tanpa tahun)

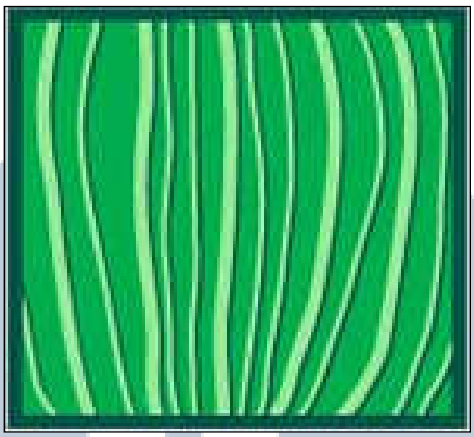
2.1.5 Parallel

Pola *parallel* ditandai dengan tulang cabang yang sejajar ke atas tanpa adanya percabangan lagi pada tulang daun cabang tersebut. Gambar 2.10 menunjukkan salah satu contoh daun *parallel* dan Gambar 2.11 menunjukkan struktur pola tulang daun *parallel* jika diperbesar.

Gambar 2.10 Contoh tulang daun *parallel*
(Sumber: Geneve, tanpa tahun)

UNIVERSITAS MULTIMEDIA NUSANTARA

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016



Gambar 2.11 Pola tulang daun *parallel*
(Sumber : Nix, tanpa tahun)

2.2 Canny Edge Detection

Edge detection adalah salah satu metode *image processing* yang digunakan untuk mengambil suatu ciri khusus dari citra. Banyak algoritma dalam *edge detection*, salah satunya adalah yang dikembangkan oleh John F. Canny pada tahun 1986 yang dikenal dengan nama *canny edge detection*. Tujuan utama dari *edge detection* adalah mengurangi jumlah data pada *image*, namun tetap menjaga sifat struktural yang akan digunakan untuk pemrosesan gambar lebih lanjut (Canny, 1986). Kriteria pengembangan *Canny Edge detection* sebagai algoritma yang optimal adalah sebagai berikut:

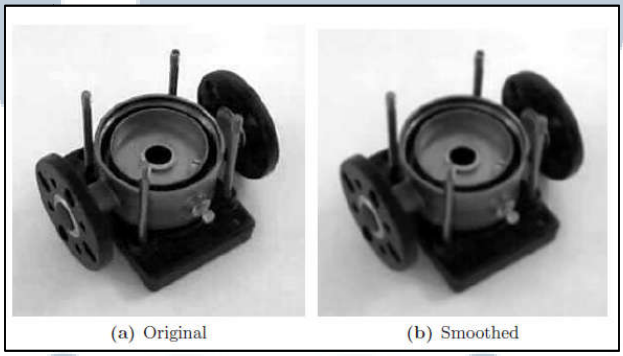
1. *Detection*: Probabilitas dalam mendeteksi titik tepi harus maksimal, sedangkan probabilitas titik non-tepi palsu harus diminimalkan.
2. *Localization*: Tepi yang dideteksi harus mempunyai kemiripan dengan tepi yang asli.
3. *Number of responses:* Satu tepi asli tidak boleh dideteksi dengan hasil yang berbeda atau lebih dari satu deteksi tepi.

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

Algoritma *canny edge detection* terdiri dari lima langkah (Moeslund, 2009) yaitu:

1. *Smoothing*

   Image dari hasil potret foto terkadang terdapat *noise* yang tentunya dapat salah dikenali sebagai tepi. *Smoothing* dilakukan dengan cara melakukan *blurring* untuk menghilangkan *noise*. *Smoothing* dilakukan dengan menggunakan *Gaussian filter*. Gambar 2.12 di bawah menunjukkan *image* sebelum dilakukan proses *smoothing* (kiri) dan setelah dilakukan proses *smoothing* (kanan).



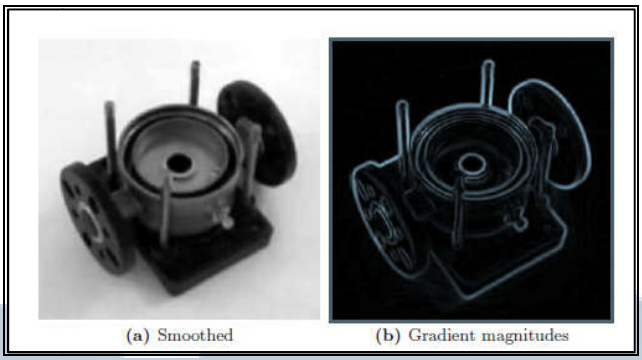
Gambar 2.12 *Smoothing image*
(Sumber: Moeslund, 2009)

2. *Finding gradients*

   Algoritma *canny edge detection* menemukan tepi dengan cara mencari nilai intensitas *grayscale* pada *image* dengan perubahan paling drastis. Hal tersebut dapat dilakukan dengan mencari *gradients* dari *image* dengan menggunakan *sobel-operator*. Gambar 2.13 menunjukkan proses perubahan *image* dari *smoothed image* ke dalam *gradient* dengan *sobel-operator*.

UNIVERSITAS MULTIMEDIA NUSANTARA

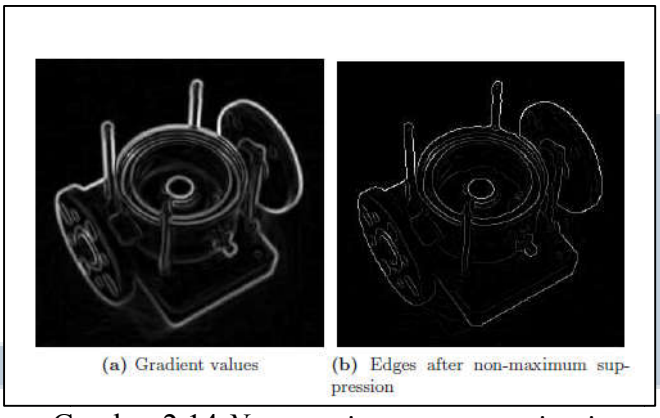
Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016



Gambar 2.13 *Gradient image*
(Sumber: Moeslund, 2009)

3. *Non-maximum suppression*

   Pada langkah ini dilakukan perubahan sisi yang *blur* pada *image* menjadi lebih tajam dan detail. Pada dasarnya, hal ini dilakukan dengan cara mempertahankan semua *local maxima* pada *image* dan menghapus bagian lainnya. Gambar 2.14 menunjukkan proses perubahan *image* dari *gradient image* ke *non-maximum suppression image*.



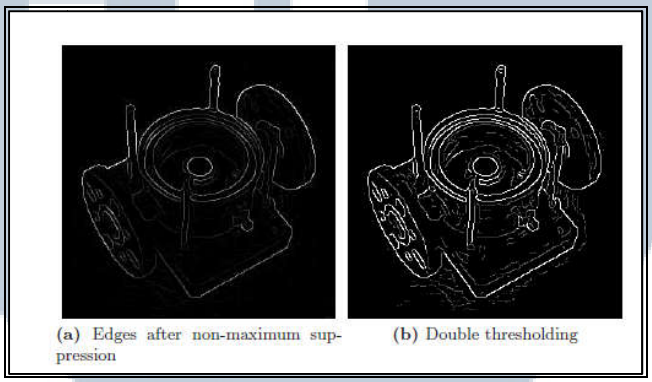
Gambar 2.14 *Non-maximum suppression image*
(Sumber: Moeslund, 2009)

4. *Double thresholding*

   Pixel pada *image* yang tersisa masih ditandai berdasarkan kekuatan tiap *pixel*. Banyak dari *pixel* tersebut adalah tepi yang asli, namun masih memiliki beberapa *noise* atau variasi warna yang dikenali sebagai tepi. Oleh karena itu digunakan *tresholding* untuk membedakan keduanya. *Canny edge detection*

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

menggunakan *double tresholding* yaitu tepi *pixel* dengan nilai lebih besar dari *high threshold* akan ditandai sebagai titik kuat (warna putih), tepi *pixel* dengan nilai di bawah *low threshold* akan dihilangkan, dan tepi *pixel* yang bernilai di antara *high* dan *low* akan ditandai sebagai titik lemah (warna abu). Gambar 2.15 menunjukkan proses perubahan *image* dari *non-maximum suppression image* ke *double tresholding image*.

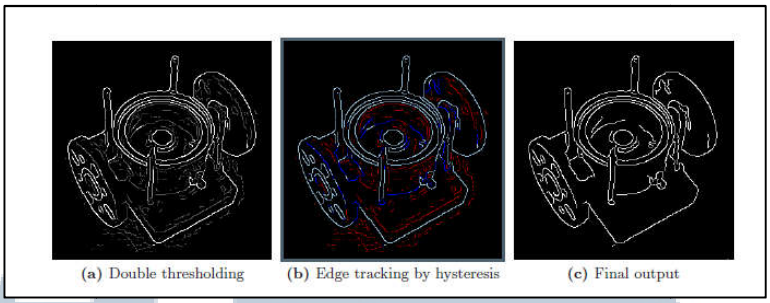


Gambar 2.15 *Double tresholding image*
(Sumber: Moeslund, 2009)

5. *Edge tracking by hysteresis*

Pixel tepi yang kuat akan dinyatakan sebagai "tepi nyata", dan dapat digunakan sebagai *output image*. Sedangkan *pixel* dengan tepi yang lemah akan dimasukkan ke dalam *output image* jika terhubung dengan tepi *pixel* yang kuat dan dinyatakan sebagai "tepi nyata". Gambar 2.16 menunjukkan proses perubahan *image* dari *double tresholding image* ke *edge tracking hysteresis image* (gambar tengah). Garis putih menunjukkan *pixel* tepi yang kuat, garis biru menunjukkan *pixel* tepi lemah yang terhubung dengan *pixel* tepi kuat, dan warna merah menunjukkan *pixel* tepi lemah yang tidak terhubung. Gambar kanan merupakan *output image canny edge detection*.

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016



Gambar 2.16 *Final output image*
(sumber: Moeslund, 2009)

2.3 Jaringan saraf tiruan

Jaringan saraf tiruan dipahami sebagai sistem pemrosesan informasi yang bekerja berdasarkan prinsip kerja dari sistem saraf biologis. Jaringan saraf tiruan dibentuk untuk memecahkan suatu masalah tertentu seperti pengenalan pola atau klasifikasi karena adanya proses pembelajaran (Ginting, 2014).

Berdasarkan Haykin (1994), jaringan saraf tiruan memberikan manfaat dan kelebihan sebagai berikut:

1. Pemetaan masukan-keluaran

   Proses pelatihan dalam paradigma umum adalah proses dengan seorang pengajar atau pelatihan yang terarah. Pelatihan pada jaringan saraf tiruan memberikan sebuah *set* pelatihan yang sudah ditentukan. Tiap *node* dalam jaringan saraf tiruan memiliki sinyal masukan yang unik dan target keluaran yang diharapkan.

2. Kemampuan adaptasi

   Jaringan saraf tiruan yang telah dilatih pada suatu kondisi, dapat dengan mudah menyesuaikan diri dengan sebuah perubahan kecil atas kondisi operasinya.

UNIVERSITAS MULTIMEDIA NUSANTARA

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

3. Respon yang jelas

   Sebuah jaringan saraf tiruan dapat dirancang untuk menyediakan informasi tidak hanya pola mana yang dipilih, melainkan juga tingkat keyakinan dari keputusan yang dibuat.

4. Analogi saraf biologi

   Layaknya jaringan saraf biologis, jaringan saraf tiruan juga memiliki toleransi kesalahan yang bekerja secara cepat dan dapat diandalkan.

2.3.1 Jaringan saraf tiruan Backpropagation

Backpropagation neural network terdiri atas beberapa proses yang harus dilakukan. Fase pelatihan adalah tahap awal yang digunakan untuk melatih jaringan saraf tiruan. Data diberikan dengan *output* pola yang sudah ditentukan. Jaringan akan menyesuaikan antara *output* dan pola yang telah ditentukan. Dalam tiap kali jaringan belajar dikenal dengan istilah *epoch*. Satu *epoch* akan terjadi jika telah melewati satu *training cycles*. Banyaknya *training cycles* atau *epoch* sangat mempengaruhi kemampuan belajar dari jaringan (Young, 1995).

Setelah melalui fase pelatihan, maka jaringan akan diambil tingkat akurasinya dengan menggunakan kembali data *input* pada fase pelatihan, tanpa melalui proses pembelajaran jaringan. *Output* akan langsung dikeluarkan oleh jaringan dan *developer* dapat membandingkan sendiri dengan target *output* yang diharapkan. Melalui proses ini maka *developer* dapat menentukan, apakah jaringan telah dapat mempelajari dan mengenali pola.

Setelah mendapatkan akurasi dari jaringan dan dianggap telah cukup, maka fase *testing* dapat dimulai. Fase *testing* dilakukan dengan cara memasukkan

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

data yang belum pernah dikenali oleh jaringan namun tetap mempunyai asal yang serupa dengan data *training*.

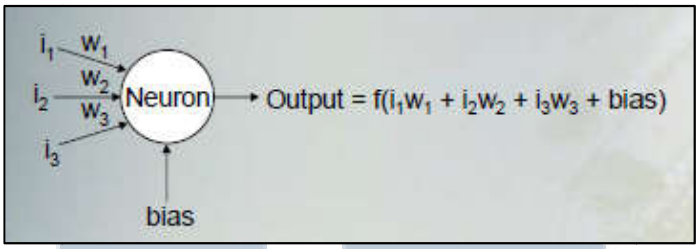
2.3.2 Parameter pembangun backpropagation

Berikut adalah parameter yang diperlukan dalam pembuatan jaringan saraf tiruan:

1. Inisialisasi *range* bobot (r): *range* bobot biasanya berkisar [-r,r]
2. Banyaknya *node* dalam *hidden layer*: Menentukan jumlah *node* dalam setiap lapisan tersembunyi. Memilih jumlah *node* dapat ditentukan dengan cara *trial and error*.
3. Banyaknya *epoch*: Satu *epoch* adalah satu putaran dalam proses *training* data. Memperbesar banyaknya *epoch* akan meningkatkan akurasi, namun akan memakan waktu yang cukup lama. Begitu juga sebaliknya, jumlah *epoch* yang terlalu sedikit akan mengurangi akurasi namun memiliki waktu pengerjaan yang cepat.
4. Penjumlahan nilai pada *node*: Penjumlahan *node* dari satu *layer* ke *layer* berikutnya digunakan dengan menjumlahkan *value node* dikalikan dengan bobotnya masing-masing lalu dijumlahkan dengan *bias*. *Bias* bernilai satu dan memiliki bobotnya sendiri. Nilai *bias* digunakan untuk menghindari *value* dari *node* agar tidak bernilai nol. Gambar 2.17 menunjukkan perhitungan *value* pada suatu *node*.

UNIVERSITAS MULTIMEDIA NUSANTARA

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016



Gambar 2.17 Penumlahan *node*
(Sumber: Cheung dan Cannons, 2002)

5. Fungsi aktivasi: Dalam *backpropagation*, fungsi aktivasi yang dipakai harus memenuhi beberapa syarat yaitu: kontinu, terdiferensial dengan mudah dan merupakan fungsi yang tidak turun (Siang, 2005). Salah satu yang paling sering dipakai adalah fungsi *sigmoid* yang memiliki *range* (0,1) yang ditunjukkan pada rumus 2.1.

$$f(x) = \frac{1}{1 + e^{-x}} \quad \ldots(2.1)$$

6. Besarnya *learning rate*: Adalah faktor pengali untuk kesalahan dalam perbaikan propagasi balik. Nilai *learning rate* yang rendah, akan menghasilkan pembelajaran yang lambat namun cenderung stabil. Nilai *learning rate* yang besar menghasilkan pembelajaran yang lebih cepat namun fase pembelajaran yang kurang stabil. Nilai *learning rate* berkisar antara 0.1 sampai 0.9.
7. *Critical error*: Merupakan batasan *error* yang diterima. Jika *error* yang keluar sama atau lebih kecil dari batasan *error* maka pelatihan dihentikan. Cara paling umum untuk menghitung *error* adalah menggunakan *Mean Square Error* (MSE).

$$MSE = (target - output)^2 \quad \ldots(2.2)$$

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

8. Perhitungan *error*: Dalam fase *backpropagation* perlu dilakukan perhitungan *error* pada tiap *node* dengan menggunakan rumus perhitungan *error* dengan cara mengkalikan turunan dari fungsi aktivasi dengan target yang diharapkan dikurangi dengan *output* dari *node* seperti yang terlihat pada Rumus 2.3.

$$\delta_j = f'(net_j)(target_j - output_j) \quad \ldots(2.3)$$

9. Perubahan bobot: Perubahan bobot dilakukan setelah *error* diperoleh untuk menghitung bobot yang baru. Nilai perubahan bobot akan diperoleh dengan mengalikan *learning rate, error*, dan *value node* seperti terlihat pada rumus 2.4.

$$\Delta w_{ji} = \alpha \delta_j x_{ji} \quad \ldots(2.4)$$

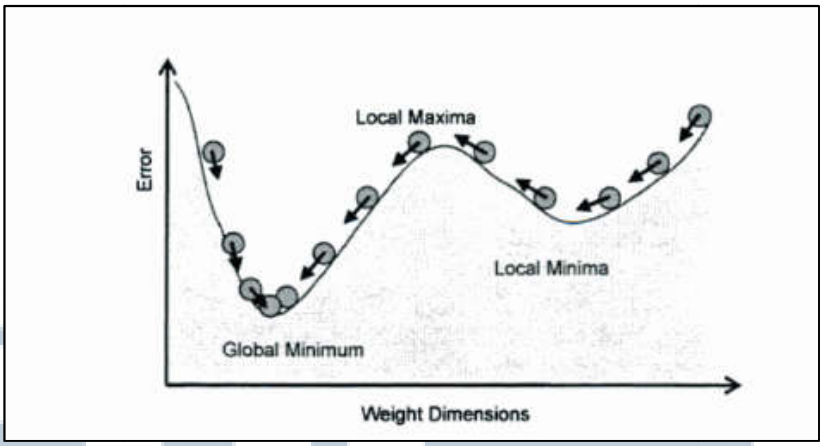
Setelah diperoleh nilai perubahan bobot, maka akan dihitung bobot baru dengan menambahkan bobot lama dan bobot baru seperti pada Rumus 2.5

$$w_{new} = w_{old} + \Delta w_{new} \quad \ldots(2.5)$$

10. *Momentum*: Merupakan parameter tambahan yang dapat dimasukkan kedalam perhitungan perubahan bobot *backpropagation*. *Momentum* digunakan untuk menghindari terjebak pada *local minimum* dan mengarahkan bergerak kedalam *global minimum* agar memperoleh *value error* yang lebih kecil seperti yang terlihat pada Gambar 2.18.

UNIVERSITAS MULTIMEDIA NUSANTARA

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016



Gambar 2.18 *Momentum*
(Sumber: Priddy dan Keller, 2005)

Momentum dapat diterapkan dengan menambahkan perubahan bobot sebelumnya (n-1) dikalikan dengan *value momentum* yang telah ditetapkan. Rumus perubahan bobot menggunakan *momentum* dapat dilihat pada Rumus 2.6.

$$\Delta w_{ji}^{n} = \alpha \delta_j x_{ji} + \mu \Delta w_{ji}^{(n-1)} \quad \ldots(2.6)$$

2.3.3 Kriteria penghentian data training

Berikut adalah kriteria dari penghentian data *training*:

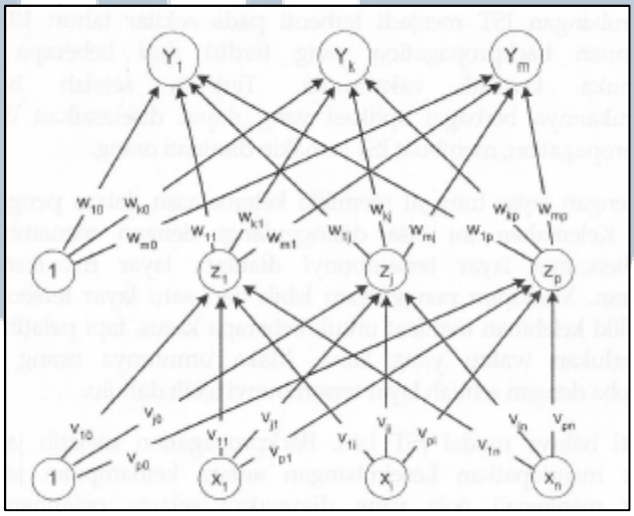
1. *The epoch/cycle control strategy*: *Training* akan terus dilakukan hingga batas *epoch* yang telah ditentukan.
2. *The error control strategy*: *Training* data akan dihentikan jika *error* yang diperoleh berada di bawah batas toleransi *error* yang didefinisikan. Biasanya menggunakan perhitungan MSE.
3. *The user control strategy*: *User* dapat menghentikan secara paksa suatu pelatihan data, jika memang tidak diperlukan pelatihan data lebih lanjut.

UNIVERSITAS MULTIMEDIA NUSANTARA

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

2.3.4 Arsitektur jaringan saraf tiruan

Jaringan saraf tiruan memiliki 3 *layer* utama, yaitu *input layer*, *hidden layer*, dan *output layer*. Gambar 2.19 adalah arsitektur dari jaringan saraf tiruan dengan n *input* (ditambah *bias*), *hidden layer* terdiri atas p *node* (ditambah *bias*), serta m *node output layer*. V_{ji} merupakan bobot dari *input node* X ke unit tersembunyi Z. W_{jk} merupakan bobot dari *hidden layer* ke *output layer*.

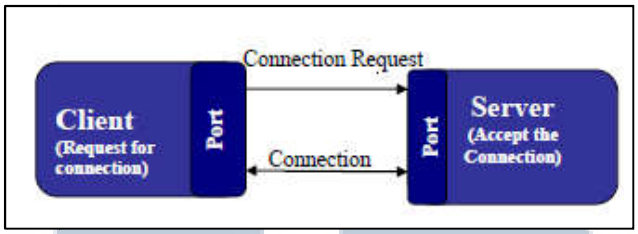


Gambar 2.19 Arsitektur jaringan saraf tiruan
(Sumber: Siang, 2005)

2.4 Socket programming

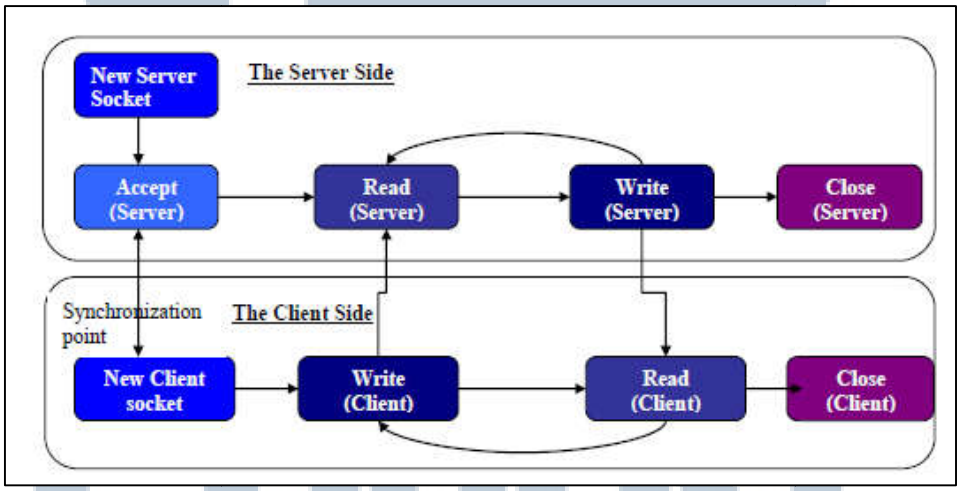
Socket merupakan cara melakukan komunikasi antar program dalam suatu jaringan yang sama. Socket digunakan untuk mewakili konektivitas antar *client* dan *server*. Untuk membuat suatu *socket connection* dibutuhkan IP dan port yang digunakan. *Server* yang sedang berjalan menggunakan suatu *port* yang telah ditetapkan. *Server* menunggu client untuk membuat *connection*. *Client* yang mengetahui IP dan *port server* akan membuat permintaan koneksi. Setelah koneksi diterima *server* maka *socket* dapat digunakan sebagai media perpindahan data (Aamir, 2010).

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016



Gambar 2.20 *Socket connection*
(Sumber: Aamir, 2010)

Socket connection melakukan proses perpindahan data dengan cara melakukan *write* dan *read* untuk proses komunikasi data (Aamir, 2010). Komunikasi data antar *client* dan *server* terjalin dua arah seperti yang terlihat pada Gambar 2.21.



Gambar 2.21 Arsitektur *socket*
(Sumber: Aamir, 2010)

2.5 Technology acceptance model (TAM)

Model TAM diadopsi dari model TRA (*Theory of Reasoned Action*) yaitu teori tindakan yang beralasan dengan satu premis bahwa reaksi dan persepsi seseorang terhadap suatu hal akan menentukan sikap dan perilaku orang tersebut (Wibowo, 2008). Model TAM dikembangkan dari teori psikologis yang berlandaskan pada kepercayaan, sikap, keinginan, dan hubungan perilaku

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

pengguna. Model ini menempatkan faktor sikap dari tiap-tiap perilaku pengguna dengan dua variabel yaitu:

1. Kemudahan penggunaaan (*ease of use*)
2. Kemanfaatan (*usefulness*)

Model TAM dapat menjelaskan bahwa persepsi pengguna akan menentukan sikapnya dalam penggunaan TI. Model ini mengambarkan penerimaan penggunaan TI dipengaruhi oleh kemanfaatan (*usefulness*) dan kemudahan penggunaan (*ease of use*).

2.6 Accidental sampling (Convenience sampling)

Sampling merupakan metode pengambilan data dari bagian suatu populasi dan memproses data tersebut menjadi sebuah informasi (Latham, 2007). Keterbatasan uang dan waktu merupakan alasan kenapa perlunya dilakukan pengambilan sampel dari populasi untuk memperoleh suatu informasi (Wilumila, 2002).

Accidental sampling merupakan bagian dari *non probability sample*. Cara ini tidak menghiraukan prinsip-prinsip *probability*. Hasil yang diharapkan hanya merupakan gambaran kasar tentang suatu keadaan. Cara ini digunakan bila biaya sangat sedikit, hasil diminta segera, tidak memerlukan ketepatan tinggi, dan hanya menggambarkan gambaran umum saja. *Accidental sampling* digunakan atas dasar populasi yang digunakan dalam penelitian terlalu besar. *Accidental sampling* dilakukan secara *accidental* dengan mengambil responden yang kebetulan ada sesuai dengan konteks penelitian.

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016

2.7 Likert Scale

Merupakan skala pengukuran yang biasa digunakan dalam kuesioner (Bertram, tanpa tahun). Dikembangkan oleh Dr. Rensis Likert, seorang *sociologis*t di *University of Michigan* dengan *report* yang berjudul "*A Technique for the Measurement of Attitudes*" yang di-*publish* pada tahun 1932. *Likert scale* dibuat dengan tujuan untuk mengukur aspek psikologis tanggapan menjadi suatu hal yang *scientific*. Biasanya *Likert scale* menggunakan lima poin skala penilaian, seperti Gambar 2.22.

Gambar 2.22 Skala *Likert*
(Sumber: Bertram, tanpa tahun)

Tiap *level* dalam skala memiliki nilai masing-masing yang berbeda dengan bertambahnya tiap satu poin dengan naiknya *level*.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Rancang bangun..., Alvin Hanjaya Tandrian, FTI UMN, 2016