

Berikut adalah standar isyarat tangan pada sistem isyarat bahasa indonesia dapat dilihat pada Gambar 2.1.



Gambar 2.1 Abjad isyarat SIBI

Gambar 2.1 merupakan bentuk isyarat tangan pada sistem isyarat bahasa Indonesia. Pada gambar terdapat 26 bentuk huruf abjad tapi yang di gunakan hanya 24 huruf abjad yaitu a, b, c, d, e, f, g, h, i, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y sementara huruf j dan z tidak diterjemahkan karena dalam bentuk gerakan.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2.2 Citra

Menurut Munir (2004), citra adalah gambar pada bidang dwimatra (dua dimensi) yang memegang peranan sangat penting sebagai bentuk visual kaya informasi. Citra merupakan objek fisik misalnya lukisan, foto dan berwujud tiga dimesi seperti patung. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh alat-alat optik, misalnya mata pada manusia, kamera, pemindai (*scanner*) dan sebagainya, sehingga bayangan objek yang disebut citra terekam (Wardani, 2011).

2.2.1 Citra Berwarna

Citra berwarna atau biasanya dikenal dengan citra RGB (*Red-Green-Blue*) merupakan jenis citra yang menyajikan warna dalam komponen R (*red* atau merah), G (*green* atau hijau) dan B (*blue* atau biru). Setiap komponen warna menggunakan delapan bit, yang mempunyai nilai berkisar antara 0 sampai dengan 255. Kemungkinan warna yang dapat disajikan oleh citra RGB ini adalah $255 \times 255 \times 255$ atau sama dengan 16.581.375 warna (Kadir dan Susanto, 2013).



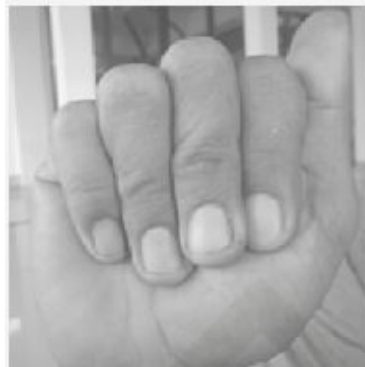
Gambar 2.2 Citra berwarna

2.2.2 Citra Berskala Keabuan

Jenis citra berskala keabuan menangani gradasi warna hitam dan putih yang menghasilkan efek warna abu-abu atau keabuan. Pada jenis citra ini, warna dinyatakan dengan intensitas. Dalam hal ini, intensitas berkisar antara 0 sampai dengan 255. Nilai 0 menyatakan hitam dan nilai 255 menyatakan putih (Kadir dan Susanto, 2013).

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



Gambar 2.3 Citra Grayscale

2.2.3 Citra Biner

Citra biner adalah citra dengan setiap piksel hanya dinyatakan dengan sebuah nilai dengan dua kemungkinan, yaitu nilai 0 dan 1. Nilai 0 menyatakan warna putih dan nilai 1 menyatakan warna hitam (Kadir dan Susanto, 2013).



Gambar 2.4 Citra Binner

Pengolahan citra adalah pemrosesan citra, khususnya dengan menggunakan komputer, menjadi citra yang kualitasnya lebih baik. Umumnya, operasi-operasi pada pengolahan citra diterapkan pada citra bila :

1. Perbaikan atau memodifikasi citra perlu dilakukan untuk meningkatkan kualitas penampakan atau untuk menonjolkan beberapa aspek informasi yang terkandung di dalam citra.
2. Elemen di dalam citra perlu di kelompokkan, dicocokkan atau diukur.
3. Sebagian citra perlu digabung dengan bagian citra yang lain.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Pengenalan citra mengelompokkan data *numeric* dan simbolik (termasuk citra) secara otomatis oleh mesin (*computer*). Tujuan pengelompokan adalah untuk mengenali suatu objek di dalam citra. Manusia bisa mengenali objek yang dilihatnya karena otak manusia telah belajar mengklasifikasi objek-objek di alam sehingga mampu membedakan suatu objek dengan objek lainnya. Kemampuan system visual manusia inilah yang dicoba ditiru oleh mesin.

2.3 Pembentukan Citra (*Data Acquisition*)

Citra merupakan gambaran tentang karakteristik suatu obyek menurut kondisi variabel tertentu. Citra yang diperoleh tergantung pada karakteristik objek yang direkam dan kondisi variabel dari sistem perekaman. Secara umum ada empat komponen dalam proses pembentukan citra, yaitu (Gonzales, 2008) : *digitizer*, komputer digital, piranti tampilan dan media penyimpanan.

Digitizer merupakan sistem penangkap citra digital yang melakukan penjelajahan citra dan mengkonversinya ke representasi numerik sebagai masukan bagi komputer digital hasil dari *digitizer* adalah matriks yang elemen-elemennya menyatakan nilai intensitas cahaya pada suatu titik. Contoh *digitizer* adalah kamera digital, *scanner*. Piranti tampilan berfungsi mengkonversi matriks intensitas yang merepresentasi citra ke tampilan yang dapat diinterpretasikan oleh mata manusia. Media penyimpanan adalah piranti yang mempunyai kapasitas memori besar sehingga gambar dapat disimpan secara permanen agar dapat diproses lagi pada waktu lain.

Menentukan data yang diperlukan dan memilih metode perekaman citra digital. Dalam tahap ini, menentukan data yang akan menjadi objek penelitian kemudian diambil datanya berupa gambar dari objek penelitian tersebut. Dalam penelitian ini objek yang diambil datanya berupa gambar dari bentuk isyarat tangan pada bahasa isyarat tunawicara.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2.4 Pre-processing

Pre-processing adalah tahap proses perubahan warna pada data citra yang dimasukkan dan tahap pemisahan objek dan background dari citra bentuk isyarat tangan yang akan dikenali. Proses *pre-processing* dilakukan untuk mendapatkan merubah ciri warna pada suatu citra sehingga dapat menjadi data inputan untuk tahap selanjutnya yaitu tahap *processing*. Pada pengenalan bentuk isyarat tangan pada bahasa isyarat tunawicara, proses yang dilakukan adalah *cropping* dan *resize*, konversi RGB ke *grayscale*, konversi *grayscale* ke biner, ekstraksi citra dengan metode deteksi tepi sobel.

Tahapan *pre-processing* yang dilakukan adalah *cropping* dan *resize*., konversi RGB ke *grayscale* , dan konversi *grayscale* ke biner, ekstraksi citra dengan metode deteksi tepi sobel.

2.4.1 Cropping dan Resize

Setelah mendapatkan nilai biner dari pola yang dimasukkan, maka selanjutnya yang harus dilakukan adalah pemetaan pola karakter pada area pola dan proses perubahan ukuran dari ukuran asli. Proses pemotongan citra yang telah diinputkan sebelumnya atau *cropping* dan proses perubahan ukuran dari ukuran asli adalah proses *resize*. Pada proses *cropping*, citra yang telah dimasukkan akan dipotong sehingga objek yang diinginkan akan terpotong sesuai dengan pemotongan citra yang dilakukan. Setelah dilakukan proses *cropping* dan akan dilakukan proses *resize* yang diuji. Melakukan proses *resize* bertujuan untuk mempercepat proses perhitungan dengan mengubah menjadi 300x300 piksel. Serta melakukan proses peningkatan citra dengan melakukan pencerahan atau menaikkan kontras dan berformat PNG.

2.4.2 Konversi RGB ke Grayscale

Salah satu persamaan dari konversi RGB ke *grayscale* yang digunakan adalah (Kadir dan Susanto, 2013).

$$Grayscale = (0,2989 * R) + (0,5870 * G) + (0,1141 * B) \dots\dots (2.1)$$

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Keterangan :

R = merepresentasikan nilai warna merah pada citra

G = merepresentasikan nilai warna hijau pada cira

B = merepresentasikan nilai warna biru pada citra

2.4.3 Konversi *Grayscale* ke Biner

Proses *thresholding* citra *grayscale* untuk menghasilkan citra biner adalah sebagai berikut.

$$g(x,y) = \begin{cases} 1, & \text{jika } f(x,y) \geq T \\ 0, & \text{jika } f(x,y) < T \end{cases} \dots\dots\dots (2.2)$$

Dengan $g(x,y)$ adalah citra biner dari citra *grayscale* $f(x,y)$ dan T menyatakan nilai ambang (*threshold*). Nilai T memegang peranan yang sangat penting dalam proses *thresholding*. Kualitas citra biner sangat bergantung pada nilai T yang digunakan.

Terdapat dua jenis *thresholding*, yaitu *global thresholding* dan *locally adaptive thresholding*. Pada *global thresholding*, seluruh *pixel* pada citra dikonversikan menjadi hitam atau putih dengan satu nilai ambang T. Kemungkinan besar pada *global thresholding* akan banyak informasi hilang karena hanya menggunakan satu nilai T untuk keseluruhan *pixel*. Untuk mengatasi masalah ini dapat menggunakan *locally adaptive thresholding*. Pada *locally adaptive thresholding*, suatu citra dibagi menjadi blok-blok kecil dengan nilai T yang berbeda (Putra, 2010). *Thresholding* pada pengenalan bentuk isyarat tangan pada bahasa isyarat tunawicara adalah tahap untuk mendapatkan citra biner yang bernilai 0 dan 1. 0 menandakan *pixel* berwarna hitam dan 1 menandakan *pixel* berwarna putih.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2.5 Ekstraksi Ciri

Ekstraksi ciri adalah proses pengambilan ciri-ciri yang terdapat pada objek didalam citra. Beberapa proses ekstraksi ciri mungkin perlu mengubah citra masukan sebagai citra biner, melakukan penipisan pola, dan sebagainya (Munir, 2004).

Tujuan ekstraksi ciri adalah untuk mereduksi data sebenarnya dengan melakukan pengukuran terhadap ciri tertentu yang membedakan pola masukan (input) satu dengan yang lainnya. Pada penelitian ini, proses ekstrasi ciri yang digunakan adalah ekstrasi ciri bentuk yaitu deteksi tepi.

Penentuan tepian suatu objek dalam citra merupakan salah satu wilayah pengolahan citra digital yang paling awal dan paling banyak diteliti. Proses ini seringkali ditempatkan sebagai langkah pertama dalam aplikasi segmentasi citra, yang bertujuan untuk mengenali objek-objek yang terdapat dalam citra ataupun konteks citra secara keseluruhan.

Banyak deteksi tepi yang digunakan untuk mendeteksi tepi. Deteksi tepi berdasarkan turunan pertama diantaranya deteksi Robert, sobel, prewitt, dan canny. Metode yang banyak digunakan untuk proses deteksi tepi adalah metode sobel

2.5.1 Deteksi Sobel

Deteksi tepi Sobel merupakan pengembangan metode robert dengan menggunakan filter HPF yang diberi satu angka nol penyangga. Metode ini mengambil prinsip dari fungsi laplacian dan gaussian yang dikenal sebagai fungsi untuk membangkitkan HPF. Kelebihan dari metode sobel ini adalah kemampuan untuk mengurangi noise sebelum melakukan perhitungan deteksi tepi.

Metode sobel menggunakan matrik 3 x 3 dan susunan pikselnya disekitar piksel (x, y) seperti kernel matrix sebagai berikut : 2. Menghitung nilai dari y-gradient. Dalam menghitung nilai dari y-gradient dapat menggunakan kernel

sebagai berikut : 3. Menghitung nilai gradient magnitudo. Diambil satu sampel objek lukisan untuk dilakukan bagan berikut : Dari diagram alir yang telah dijabarkan sebelumnya, maka dapat dijelaskan proses deteksi tepi dengan metode sobel adalah sebagai berikut : 1. Menghitung nilai dari x-gradient. Dalam menghitung nilai dari x-gradient dapat menggunakan penghitungan matrik dengan menggunakan kernel metode sobel 3 X 3

Peninjauan pengaturan pixel di sekitar pixelnya (x,y) adalah :

$$\begin{matrix}
 a_0 & a_1 & a_2 \\
 a_7 & (x, y) & a_3 \\
 a_6 & a_5 & a_4
 \end{matrix} \dots\dots\dots(2.3)$$

Operator sobel adalah magnitudo dari gradien yang dihitung dengan :

$$M = \sqrt{S_x^2 + S_y^2} \dots\dots\dots(2.4)$$

dalam hal ini turunan parsial dihitung dengan :

$$\begin{aligned}
 s_x &= (a_2 + ca_3 + a_4) - (a_0 + ca_7 + a_6) \\
 s_x &= (a_0 + ca_1 + a_2) - (a_6 + ca_5 + a_4) \dots\dots\dots(2.5)
 \end{aligned}$$

Dengan konstanta c = 2. Dalam bentuk mask, sy dan sx dapat dinyatakan sebagai :

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \text{ dan } S_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \dots\dots\dots(2.6)$$

Arah tepi dihitung dengan persamaan :

$$a(x, y) = \tan^{-1} \frac{S_y}{S_x} \dots\dots\dots(2.7)$$

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2.6 Jaringan Syaraf Tiruan

Menurut (Maharani, dkk, 2012) pada jurnalnya yang berjudul Perbandingan Metode Jaringan Syaraf Tiruan Backpropagation Dan Learning Vector Quantization Pada Pengenalan Wajah mendefinisikan sebagai berikut, “Jaringan Syaraf Tiruan (JST) merupakan suatu sistem pemrosesan informasi yang mempunyai karakteristik menyerupai jaringan syaraf biologi (JSB).” JST tercipta sebagai suatu generalisasi model matematis dari pemahaman manusia (human cognition) yang didasarkan atas asumsi sebagai berikut :

1. Pemrosesan informasi terjadi pada elemen sederhana yang disebut neuron
2. Sinyal mengalir diantara sel saraf/neuron melalui suatu sambungan penghubung.
3. Setiap sambungan penghubung memiliki bobot yang bersesuaian. Bobot ini akan digunakan untuk menggandakan atau mengalikan sinyal yang dikirim melaluinya.
4. Setiap sel syaraf akan menerapkan fungsi aktivasi terhadap sinyal hasil penjumlahan berbobot yang masuk kepadanya untuk menentukan sinyal keluarannya.

Menurut (Halim, dkk, 2000) pada jurnalnya yang berjudul Penerapan Jaringan Saraf Tiruan Untuk Peramalan, mendefinisikan sebagai berikut : “Sistem saraf tiruan atau jaringan saraf tiruan adalah sistem selular fisik yang dapat memperoleh, menyimpan dan menggunakan pengetahuan yang didapatkan dari pengalaman”.

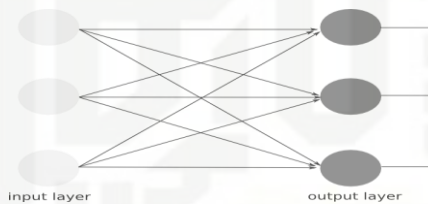
2.6.1 Arsitektur Jaringan

Arsitektur jaringan syaraf tiruan merupakan pengaturan neuron dalam *layer* dan hubungan-hubungannya. Arsitektur sebuah jaringan akan menentukan keberhasilan sebuah pola target yang akan dicapai, karena tidak semua permasalahan dapat diselesaikan dengan arsitektur yang sama.

Adapun jenis-jenis arsitektur pada jaringan syaraf tiruan adalah sebagai berikut :

1. *Single-Layer Feedforward Network*

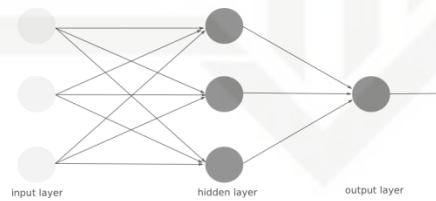
Susunan jaringan *single layer*, neuron-neuron dapat dikelompokkan menjadi dua bagian, yaitu unit-unit input dan unit-unit output. Unit input menerima masukan dari luar sedangkan unit output akan mengeluarkan respon dari jaringan sesuai dengan masukannya. Contoh jaringan syaraf dengan *single layer* adalah sebagai berikut :



Gambar 2.5 Single Layer Feed forward Net

2. *Multi-Layer Feed forward Network*

Susunan jaringan *multi layer*, selain ada unit-unit input dan unit-unit output, juga terdapat unit-unit tersembunyi (*hidden layer*). Jumlah unit *hidden* tergantung pada kebutuhan. Semakin kompleks jaringan, unit *hidden* yang dibutuhkan semakin banyak, demikian pula jumlah *layernya*. Contoh jaringan syaraf dengan *multi layer* adalah sebagai berikut :

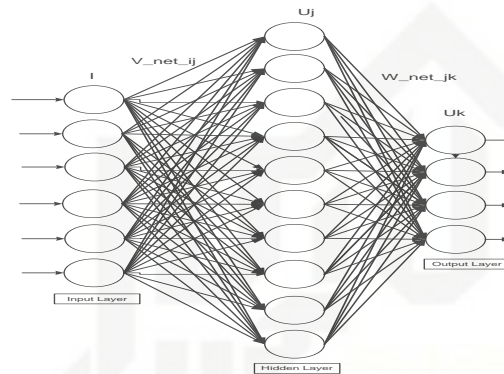


Gambar 2.6 Multi Layer Feed forward Net

2.6.2 Back Propagation Neural Network (BPNN)

Backpropagation neural network (BPNN) merupakan salah satu algoritma pembelajaran dalam jaringan saraf tiruan. Proses pembelajaran dalam backpropagation dilakukan dengan penyusuan bobot-bobot jaringan saraf tiruan dengan arah mundur berdasarkan nilai error dalam proses pembelajaran.

BPNN melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang serupa (tapi tidak sama) dengan pola yang dipakai selama pelatihan (Riztyan, 2007).



Gambar 2.7 Arsitektur jaringan BPNN

Keterangan gambar :

Input layer = lapisan yang menerima data masukan

Hidden layer = lapisan tersembunyi yang mengolah data inputan

Output layer = lapisan keluaran yang akan mengeluarkan informasi dari lapisan tersembunyi

I = inputan

U_j = neuron pada layer tersembunyi

U_k = neuron pada layer keluaran

V_{net_ij} = bobot masukan yang di olah pada layer masukan (i) dan tersembunyi (j)

W_{net_jk} = bobot keluaran yang diolah pada layer tersembunyi (j) dan keluaran (k)

Penggunaan *Backpropagation* terdiri dari 2 tahap :

1. Tahap belajar atau pelatihan, dimana pada tahap ini BPNN diberikan sejumlah data pelatihan dan target.
2. tahap pengujian atau penggunaan, pengujian dan penggunaan dilakukan setelah BPNN selesai melakukan pembelajaran.

Hak Cipta Dilindungi Undang-Undang
 1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2.6.3 Fungsi Aktivasi Pada *Backpropagation*

Dalam jaringan saraf tiruan, fungsi aktivasi dipakai untuk menentukan keluaran suatu neuron. Pada BPNN, fungsi aktivasi yang dipakai harus memenuhi beberapa syarat yaitu : kontinu, terdiferensial dengan mudah, fungsi yang tidak turun (Prahesti, 2013).

Beberapa fungsi diantaranya adalah sebagai berikut :

1. *Sigmoid biner*

Fungsi *sigmoid biner* memiliki nilai pada *range* 0 sampai 1. Fungsi ini sering digunakan untuk jaringan saraf yang membutuhkan nilai output yang terletak pada interval 0 sampai 1. Definisi *sigmoid biner* adalah sebagai berikut :

$$f(x) = \frac{1}{1+e^{-x}} \text{ dengan turunan } f'(x) = f(x)(1 - f(x)) \dots \dots \dots (2.8)$$

2. *Sigmoid Bipolar*

Fungsi *sigmoid bipolar* hampir sama dengan *sigmoid biner*, hanya saja output dari fungsi ini memiliki range 1 sampai -1. Definisi fungsi ini adalah sebagai berikut :

$$f(x) = \frac{2}{1+e^{-x}} - 1 \text{ dengan turunan } f'(x) = \frac{(1+f(x))(1-f(x))}{2} \dots \dots \dots (2.9)$$

3. Fungsi Linier (Identitas)

Fungsi linier memiliki nilai *output* yang sama dengan inputnya. Sehingga nilai *input* dan *output* menghasilkan satu garis lurus jika dihubungkan pada suatu grafik. Definisi fungsi ini adalah sebagai berikut :

$$Y = X \dots \dots \dots (2.10)$$

2.6.4 Pelatihan *Backpropagation*

Sebelum melakukan pelatihan, terlebih dahulu lakukan transformasi data untuk menyesuaikan nilai data dengan range fungsi aktivasi yang digunakan dalam jaringan yang dalam hal ini adalah fungsi sigmoid biner dan fungsi sigmoid bipolar. Keluaran fungsi aktivasi sigmoid adalah [0.1] dan [-1.1]. Oleh karena itu,

data juga harus ditransformasikan ke interval [0.1] dan [-1.1]. Namun, akan lebih baik jika ditransformasikan ke interval yang lebih kecil, misalnya pada interval [0.1,0.9], karena mengingat fungsi sigmoid nilainya tidak pernah mencapai 0 ataupun 1.

Untuk mentransformasikan data ke interval [0.1,0.9] dilakukan dengan transformasi linier sebagai berikut (Nurmila dan Sugiharto, 2005) :

$$x' = \frac{(b-a)(x-x_{\min})}{(x_{\max}-x_{\min})} + a \dots \dots \dots (2.11)$$

Keterangan :

- x' : Hasil transformasi data
- x_{\max} : Nilai terbesar
- x_{\min} : Nilai terkecil
- a : Batas bawah interval
- b : Batas atas interval

Terdapat 3 fase dalam pelatihan BPNN, yaitu :

1. Data masukan ke input jaringan (*feed forward*)
2. Perhitungan dengan propagasi balik dari error yang bersangkutan
3. Pembaharuan bobot dan bias

Langkah-langkah pelatihan dalam jaringan saraf tiruan BPNN adalah sebagai berikut :

Langkah 0 : Input data latih,

Inisialisasi bobot awal (ambil nilai random yang cukup kecil)

Tentukan epoch dan error.

Langkah 1 : Lakukan langkah dibawah ini selama epoch (max epoch) target error.

Fase I : *feed forward* :

Langkah 2 : Setiap unit input ($X, i=1,\dots,n$) menerima sinyal inputan x dan menyebarkan sinyal tersebut pada seluruh unit *hidden layer*. Input x yang digunakan adalah input training data.

Langkah 3 : Setiap hidden unit ($Z_j=1,\dots,p$) akan menjumlahkan sinyal-sinyal input yang sudah berbobot, termasuk bias.

$$z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ji} \dots \dots \dots (2.12)$$

Dengan : z_{net_j} = total sinyal masukan pada lintasan j

v_{j0} = bobot antara masukan unit i dan lapisan unit j

v_{ji} = bobot bias masukan unit i dan lapisan unit j.

Hitung semua keluaran pada lapisan unit j (lapisan tersembunyi) menggunakan fungsi aktivasi :

$$z_j = f(z_{net_j} = \frac{1}{1+e^{-z_{net_j}}}) \dots \dots \dots (2.13)$$

Dengan: z_j = keluaran pada lapisan unit j

z_{net_j} = total sinyal pada lintasan j

Langkah 4 : Jumlah semua sinyal yang masuk kekeluaran unit k (*output layer*)
 Tiap-tiap unit keluaran j meneruskan sinyal tersebut kesemua unit lapisan yang ada di atasnya (unit k / *output layer*) dengan melewati lintasan k dengan menjumlahkan sinyal keluaran pada unit j (z_j) dengan bobot keluaran (w_{kj}).

$$y_{net_k} = w_{k0} + \sum_j^p z_j w_{kj} \dots \dots \dots (2.14)$$

Dengan : y_{net_k} = total sinyal masukan pada keluaran unit k

z_j = nilai masukan pada lapisan unit j

w_{kj} = bobot antara lapisan unit j dan keluaran unit k

Hitung keluaran pada unit k dengan menggunakan fungsi aktivasi :

$$y_k = f(y_{net_k}) = \frac{1}{\alpha + e^{-y_{net_k}}} \dots \dots \dots (2.15)$$

Dengan : y_{net_k} = Keluaran pada unit k

y_{net_k} = total sinyal pada lintasan k

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Fase II : *back forward* :

Langkah 5 : Hitung Keluaran pada unit k Tiap-tiap unit k ($y_k, k=1,2,3,\dots,m$) menerima target pola yang berhubungan dengan pola masukan.

Hitung kesalahan :

$$\delta_k = (t_k - y_k)y_k(1 - y_k)\dots\dots\dots(2.16)$$

Dengan: δ_k = Faktor kesalahan pada keluaran unit k

y_k = keluaran pada keluaran unit k

Kemudian hitung koreksi bobot (masukan) pada unit k yang nantinya akan digunakan untuk memperbaiki nilai y_{jk} (masukan / bobot pada lintasan j dan k).

$$\Delta w_{kj} = \alpha \delta_k z_j \dots\dots\dots(2.17)$$

Dengan: Δw_{kj} = jumlah koreksi bobot / masukan ($\Delta w_{kj}, j = 1,2,3..m$)

α = learning rate / nilai pembantu

z_j = keluaran pada unit j

Langkah 6 : Hitung penjumlahan kesalahannya penjumlahan kesalahan dengan menjumlahkan faktor kesalahan dengan koreksi bobot dari unit-unit yang berada pada lapisan di atasnya :

$$\delta_{net\ j} = \sum_{k=1}^m \delta_k w_{kj} \dots\dots\dots(2.18)$$

Kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi error pada unit j :

$$\delta_j = \delta_{net\ j} z_j (1 - z_j) \dots\dots\dots(2.19)$$

Kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai v_{jt}).

$$\Delta v_{jt} = \alpha \delta_j x_t \dots\dots\dots(2.20)$$

Fase III : Perubahan bobot:

Langkah 7 : Ubah bobot yang menuju keluaran lapisan. Jumlahkan bobot masukan (lama) dengan jumlah koreksi bobot pada unit j dan i :

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

$$v_{ji}(\text{baru}) = v_{kj}(\text{lama}) + \Delta v_{ji} \quad \dots\dots\dots(2.21)$$

Ubah bobot yang menuju lapisan tersembunyi jumlahkan bobot keluaran (lama) dengan jumlah koreksi bobot pada unit j dan i :

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + \Delta w_{kj} \dots\dots\dots(2.22)$$

Langkah 8 : Hitung MSE pada setiap epoch

Pelatihan ini dilakukan secara berulang-ulang dengan menggunakan data pelatihan, dan parameter yang telah ditentukan. Tujuan dari pelatihan yang berulang-ulang ini adalah untuk mendapatkan karakteristik BPNN yang terbaik sehingga BPNN tersebut dapat mempelajari motif yang diberikan dengan benar.

2.6.5 Optimalisasi Arsitektur *Backpropagation*

Masalah utama yang dihadapi dalam BPNN adalah lamanya iterasi yang harus dilakukan. BPNN tidak dapat memberikan kepastian tentang berapa epoch yang harus dilalui untuk mencapai kondisi yang diinginkan. Oleh karena itu orang berusaha meneliti bagaimana parameter-parameter jaringan dibuat sehingga menghasilkan jumlah iterasi yang relatif lebih sedikit.

2.6.6 Pemilihan Bobot dan Bias Awal

Pemilihan Bobot awal akan mempengaruhi apakah jaringan mencapai titik minimum lokal atau global, dan seberapa cepat konvergensinya. Bobot yang menghasilkan nilai turunan aktivasi yang kecil sedapat mungkin dihindari karena akan menyebabkan perubahan bobotnya menjadi sangat kecil. Demikian pula nilai bobot awal tidak boleh terlalu besar karena nilai turunan fungsi aktivasinya menjadi sangat kecil juga. Oleh karena itu dalam “Standar BPNN”, bobot dan bias diisi dengan bilangan acak kecil.

Hak Cipta Dilindungi Undang-Undang
 1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2.6.7 Lama Iterasi

Tujuan utama penggunaan BPNN adalah mendapatkan keseimbangan antara pengenalan bentuk isyarat tangan secara benar dan respon yang baik untuk bentuk isyarat lain yang sejenis (data pengujian). Jaringan dapat dilatih terus menerus hingga semua bentuk isyarat tangan pelatihan dikenali dengan benar. Akan tetapi hal itu tidak menjamin jaringan akan mampu mengenali bentuk pengujian dengan tepat. Jadi tidaklah bermanfaat untuk meneruskan iterasi hingga semua kesalahan motif pelatihan = 0.

Umumnya data dibagi menjadi dua bagian (saling asing), yaitu motif data pelatihan dan data pengujian. Perubahan bobot dilakukan berdasarkan bentuk pelatihan. Akan tetapi selama pelatihan (misalkan setiap 10 epoch), kesalahan yang terjadi dihitung berdasarkan semua data (pelatihan dan pengujian).

2.6.8 Akurasi

Penghitungan akurasi sebagai tolak ukur evaluasi dalam sistem dapat diperoleh dengan menggunakan *recognition rate*. Dimana dari *recognition rate* tersebut akan dihitung presentase total pengenalannya.

$$Recognition Rate = \frac{\sum Correct}{\sum Sample} \times 100\% \dots \dots \dots (2.23)$$

2.7 Studi Penelitian Terkait

Tabel 2.1 Penelitian Terkait

Penulis	Judul penelitian	Kesimpulan
Farida dan Susilawati, 2010	Pengenalan Isyarat Tangan Statis Pada Sistem Isyarat Bahasa Indonesia Berbasis Jaringan	Dari uji coba yang dilakukan penelitian ini mempelajari tentang pengenalan isyarat tangan statis pada system isyarat bahasa indonsia berbasis jaringan syaraf tiruan perambatan balik. Pada penelitian ini, jumlah kelas isyarat

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Penulis	Judul penelitian	Kesimpulan
	Saraf Tiruan Perambatan Balik	kata yang dikenali adalah 15 kelas kata. Arsitektur yang diterapkan terdiri dari 4096 neuron pada lapisan input dengan 15 neuron pada lapisan output. Hasil pengujian verifikasi yang dicapai adalah 69%. Ditinjau dari arsitekturnya, penelitian ini memiliki jumlah neuron yang jauh lebih kecil. Pada penelitian yang dilakukan oleh Klimes isyarat yang dikenali terdiri atas 8 kelas kata dengan menerapkan 65536 neuron pada lapisan input dan 8neuron pada lapisan tersembunyi. Tingkat keberhasilan pengujian verifikasi yang dicapai adalah 80%.Sehingga dalam hal waktu komputasinya hasil penelitian ini lebih cepat dibandingkan dengan Klimes. Hasil pengujian verifikasi pada penelitian ini lebih rendah dibandingkan Klimes karena jumlah kelas yang dikenali lebih banyak dan tingkat variasi datanya lebih tinggi. Pada Klimes, posisi tangan terhadap wajah relatif konstan sehingga cukup mengambil informasi citra tangannya saja, sedangkan pada penelitian ini ada variasi antara posisi tangan dengan wajah, sehingga dalam pengolahan datanya melibatkan citra tangan dan wajah.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Penulis	Judul penelitian	Kesimpulan
Khairun, dkk, 2014	Implementasi Pengenalan Bahasa Isyarat Tangan Secara Real Time Menggunakan Metode <i>Two Dimensional Linear Discriminant Analysis</i>	Menampilkan keadaan <i>error</i> hasil pengujian dari metode <i>Two Dimensional Linear Discriminant Analysis</i> . Pengujian terhadap sistem untuk pengenalan bahasa isyarat tangan secara real time menggunakan metode <i>Two Dimensional Linear Discriminant Analysis (2D-LDA)</i> dapat diambil kesimpulan, bahwa faktor pencahayaan dan jarak pengambilan gambar dengan webcam sangat mempengaruhi untuk mendapatkan kualitas citra tangan yang lebih baik. Beberapa huruf yang memiliki kesamaan postur tangan adalah huruf “A”, “E”, “M”, “N”, dan “S”, huruf “K” dan “V”, huruf “R” dan “U”. Hal ini mengakibatkan citra tangan dari beberapa huruf tersebut sering tidak teridentifikasi sebagai huruf yang benar. Pada penelitian tersebut, tingkat akurasi paling tinggi yakni skenario lima mencapai 95.38%.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Penulis	Judul penelitian	Kesimpulan
Dika Swadan, dkk, 2013	Rancang Bangun Modul Pengenalan Bahasa Isyarat Menggunakan Teknologi Kinect	Dalam penelitian rancang bangun modul pengenalan bahasa isyarat menggunakan teknologi kinect terdapat 20 isyarat kata yang diuji. Masing-masing isyarat diperagakan oleh pengguna sebanyak 5 kali. Khusus penderita tunarungu hanya akan memperagakan setiap gerakan isyarat sebanyak 2 kali. Sehingga terdapat 540 data uji yang digunakan. Hasil pengujian yang menunjukkan bahwa persentase aplikasi dapat mengenali suatu gerakan isyarat kata dengan benar adalah 72%.
Imroatun Hasanah, 2016	Pengenalan Motif Songket Riau Berdasarkan Tekstur dan Warna Menggunakan Metode <i>Backpropagation Neural Network</i>	Dalam penelitian Pengenalan Motif Songket Riau Berdasarkan Tekstur dan Warna Menggunakan Metode <i>Backpropagation Neural Network</i> Setelah melalui tahap analisa dan pengujian aplikasi pengenalan motif songket Riau dengan menggunakan HSV dan GLCM dengan metode <i>Backpropagation Neural Network</i> , maka dapat diambil beberapa kesimpulan tercapainya sebuah aplikasi untuk mengenali songket Riau dengan menerapkan warna Hue, Saturation dan Value (HSV), tekstur Gray Level Co-occurrence Matrix (GLCM) dan <i>Backpropagation Neural Network</i>

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Penulis	Judul penelitian	Kesimpulan
		<p>(BPNN). Untuk percobaan menggunakan metode <i>HSV</i> dengan klasifikasi <i>Backpropagation Neural Network</i> maka didapatkan akurasi yaitu 84% dengan data latih 80 dan data uji 20 dengan epoch 1000 dengan <i>Learning Rate</i> 0,9 dan parameter Momentum 0,9. Untuk percobaan menggunakan metode <i>GLCM</i> dengan klasifikasi <i>Backpropagation Neural Network</i> maka didapatkan akurasi yaitu 92% dengan data latih 80 dan data uji 20 dengan epoch 1000 dengan <i>Learning Rate</i> 0,9 dan parameter Momentum 0,9. Metode HSV dan GLCM klasifikasi dengan metode <i>Backpropagation Neural Network</i> didapatkan akurasi tertinggi yaitu 99% untuk data latih 80 dan data uji 20. Akurasi terendah dari data latih 70 dan 30 data uji didapatkan akurasi 97% dengan epoch 1000 dengan <i>Learning Rate</i> 0,9 dan parameter Momentum 0,9.</p>
Titis, dkk, 2014	Pengenalan Karakter Teks Menggunakan Metode Neural Network Backpropagation	Dalam penelitian ini karakter teks menggunakan metode neural network backpropagation menampilkan sistem memiliki tingkat keberhasilan pengenalan rata-rata sebesar 80.12% dengan kemampuannya mengenali 647 data dari

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Penulis	Judul penelitian	Kesimpulan
		800 data yang terdiri atas data <i>training</i> dan data uji.
Putri Cholidhazia, 2018	Implementasi jaringan syaraf tiruan Backpropagation untuk prediksi harga crude palm oil (cpo)	Penelitian ini menggunakan metode backpropagation untuk prediksi harga <i>crude palm oil</i> (CPO) mendapat hasil yang baik. Parameter yang digunakan pada proses pengujian yaitu jumlah neuron <i>hidden</i> (6),(7),(8) dan <i>learning rate</i> (0,1-0,9) dengan jumlah <i>epoch</i> 1000 – 5000. Dengan perbandingan data latih 70% dan data uji 30%, data latih 80% dan data uji 80%, dan data latih 90% data uji 10%. Penelitian ini menghasilkan akurasi terbaik yaitu 97.80% pada presentasi pembagian data 90% data latih dan 10% data uji dengan jumlah <i>hidden</i> 8 dan <i>learning rate</i> 0.9 dan jumlah <i>epoch</i> 5000.
Achmad Hidayatno, R. Rizal Isnanto, Bahrun Niam	Analisis Deteksi Tepi Pada Citra Berdasarkan Perbaikan Kualitas Citra	Dari hasil penelitian dan pembahasan, maka telah dapat dikembangkan sebuah program aplikasi untuk menganalisis deteksi tepi Sobel, deteksi tepi Prewitt, deteksi tepi Roberts, dan deteksi tepi Canny pada citra menggunakan bahasa pemrograman Matlab versi 7.0. Nilai indeks kualitas tertinggi yang diperoleh citra Jupiter adalah sebesar 0.81182, dan terendah sebesar 0.44142. Nilai indeks kualitas tertinggi yang diperoleh citra Lena adalah sebesar 0.80763, dan terendah

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Penulis	Judul penelitian	Kesimpulan
		<p>sebesar 0.42131. Nilai indeks kualitas tertinggi yang diperoleh citra Pear adalah 0.81182, dan terendah sebesar 0.44142. Nilai indeks kualitas tertinggi yang diperoleh pada citra Tape adalah 0.77776, dan sebesar 0.31204. Nilai indeks kualitas tertinggi yang diperoleh citra Kids adalah 0.70873, dan terendah sebesar 0.41064. Nilai indeks kualitas kedua memiliki nilai yang lebih tinggi dari pada nilai indeks kualitas pertama, karena citra mengalami perbaikan. Hasil deteksi tepi suatu citra dengan jenis derau tertentu, tapis tertentu, dan deteksi tepi tertentu memiliki indeks kualitas yang berbeda dibandingkan dengan hasil deteksi tepi citra yang lain karena pada elemen matrik antara citra yang satu dengan citra yang lain berbeda</p>