

MAKALAH SEMINAR TUGAS AKHIR

Pengenalan Teks Braille Berbasis Jaringan Syaraf Tiruan Feedforward Multilayer dengan Menggunakan Metode Back Propagation

Praditya Firmansyah *, Wahyul Amien Syafei**, Iwan Setiawan**

Abstrak

Huruf braille merupakan suatu bentuk tulisan khusus yang digunakan oleh para tunanetra untuk mengetahui informasi dan ilmu pengetahuan. Dalam tugas akhir ini dibuat suatu perangkat lunak untuk mengenali teks braille agar para tunanetra dapat mengetahui informasi dan ilmu pengetahuan dengan nyaman.

Proses pengenalan karakter teks braille ini menggunakan jaringan syaraf (neuron) tiruan feedforward multilayer dengan metode backpropagation serta menggunakan adanya suatu proses pengolahan gambar.

Pada tahap pengenalan pola huruf braille, sebelumnya ada proses pengolahan gambar, dimana citra masukan teks braille diambil secara *offline* dengan menggunakan scanner terlebih dahulu sebelum dilakukan proses selanjutnya, yaitu: grayscale, deteksi tepi, thresholding, ekstraksi ciri dan kemudian selanjutnya dapat dilakukan pelatihan dan pengenalan huruf serta kemudian tulisan yang telah dikenali tersebut akan dikeluarkan oleh komputer pribadi dalam bentuk suara sesuai dengan huruf yang telah diproses.

Kata-kunci : *jaringan syaraf tiruan, huruf braille, backpropagation.*

I. PENDAHULUAN**1.1 Latar Belakang**

Dengan perkembangan teknologi komputer yang sangat pesat, maka *Personal Computer (PC)* kini dapat meniru sistem kerja dari jaringan syaraf biologi atau sistem syaraf manusia untuk diaplikasikan pada suatu jaringan syaraf tiruan (JST). Metode jaringan syaraf tiruan dikembangkan dengan berbagai cara guna mengenali pola-pola yang dimasukkan ke jaringan tersebut. Salah satu aplikasi dari jaringan syaraf tiruan dapat diterapkan pada pengenalan pola huruf braille.

Proses pembacaan melalui meraba dengan jari dapat membutuhkan waktu yang relatif lama, selain itu tidak semua tunanetra mampu membacanya karena memerlukan pendidikan di sekolah luar biasa (SLB).

1

Pengenalan teks braille dengan jaringan syaraf tiruan ini dimaksudkan untuk membantu tunanetra membaca suatu tulisan dengan mudah dan nyaman serta membutuhkan waktu yang relatif singkat dibandingkan membaca dengan meraba.

Oleh karena itu dibutuhkan suatu perangkat lunak yang dapat mengenali pola-pola huruf braille. Agar output dari perangkat lunak tersebut dapat didengar oleh tunanetra, maka output yang dihasilkan dalam bentuk suara.

Adapun metode yang digunakan yaitu mengembangkan Jaringan Syaraf Tiruan dengan metode pembelajaran *backpropagation*.

1.2 Tujuan dan Manfaat

Tujuan dari tugas akhir ini adalah membuat suatu perangkat lunak (*software*) yang mampu mengenali pola-pola huruf braille dan menghasilkan output berupa suara.

1.3 Pembatasan Masalah

1. Perencanaan dan pembuatan perangkat lunak digunakan untuk mengenali pola-pola huruf braille.
2. Proses pengambilan teks braille dilakukan secara *offline* menggunakan scanner dengan hasil citra Braille yang berformat .BMP.
3. Warna kertas yang dipakai untuk penulisan huruf braille telah ditentukan yaitu berwarna putih.
4. Tidak membandingkan unjuk kerja jumlah *node* atau titik dalam hidden layer.
5. Pola-pola huruf braille yang akan dikenali adalah huruf kecil dan spasi.
6. Pengambilan gambar harus tegak lurus dan tidak terbalik.

II. LANDASAN TEORI**2.1 Artificial Neural Network (Jaringan Syaraf Tiruan)**

Pendefinisian jaringan syaraf tiruan dilihat dari fungsi atau struktur rancangan merupakan penyederhanaan dari model otak manusia. Kinerja struktur jaringan syaraf biologi pada otak manusia adalah dengan cara menyampaikan sinyal dari satu neuron ke neuron yang lain yang berdekatan serta bersesuaian. Hal yang sama berlanjut untuk neuron yang berikutnya, sampai pada neuron terakhir yang dikehendaki sinyal tersebut.

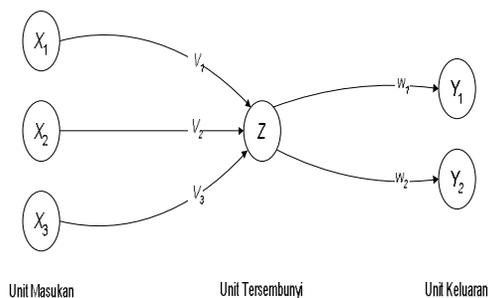
* Mahasiswa Jurusan Teknik Elektro UNDIP

** Staf Pengajar Jurusan Teknik Elektro UNDIP

JST dicirikan oleh (1) hubungan pola antar neuron (disebut sebagai arsitektur), (2) metode untuk menentukan bobot pada hubungan (disebut sebagai algoritma pelatihan atau pembelajaran) dan (3) fungsi aktivasinya.

Sebuah jaringan syaraf terdiri dari banyak elemen pengolah sederhana yang disebut *neuron*, *unit*, *cell*, atau *node*. Masing-masing neuron tersebut dihubungkan ke neuron yang lain oleh rantai komunikasi terarah dengan masing-masing bobot yang berkaitan dengannya.

Dalam aplikasi sesungguhnya tidaklah cukup hanya menggunakan satu neuron saja, biasanya terdapat *layer* tambahan. Meskipun jaringan syaraf pada gambar 1 relatif sederhana, keberadaan unit tersembunyi (*hidden unit*), memberikan kemampuan untuk memecahkan banyak permasalahan daripada yang dapat dilakukan oleh jaringan yang hanya terdiri dari input dan output unit



Gambar 1 Jaringan syaraf sederhana dengan hidden unit

2.2 Metode Pembelajaran

Pembelajaran (*learning*) bagi JST merupakan proses mengatur harga-harga dari parameter bobotnya untuk mendapatkan harga yang terbaik dengan melatih (*training*) jaringan menurut unjuk kerja sistem yang dikehendaki.

Tujuannya agar kumpulan pola input (vektor input) yang diberikan menghasilkan pola output (vektor output) yang diinginkan atau paling sedikit mendekati. Training ini dibentuk dengan menerapkan secara berurutan pada pola input dan mengatur bobot jaringan mendekati pola output mengikuti suatu algoritma belajar tertentu selama proses belajar, pembobot secara perlahan konvergen menuju harga tertentu, sehingga pola input menghasilkan pola output yang diinginkan.

Ada 2 metode pembelajaran, yaitu: Pembelajaran terbimbing (*Supervised training*) dan Pembelajaran tidak terbimbing (*Un-supervised training*)

Algoritma propagasi balik atau *Backpropagation* atau *Generalized Delta Rule* termasuk di dalam metode pembelajaran terbimbing (*supervised training*). Algoritma ini membutuhkan

pasangan untuk tiap vektor input dengan vektor target (keluaran yang diinginkan). Jika jaringan diberi *input* baru, output jaringan akan diubah seperti yang diharapkan berdasarkan pola yang sudah ada. Selisih dari kedua *output* tersebut menyatakan kesalahan (*error*) yang akan digunakan untuk mengubah pembobot sambungan. Sehingga kesalahan akan semakin kecil dalam siklus pelatihan berikutnya.

2.3 Pengolahan Citra (image processing)

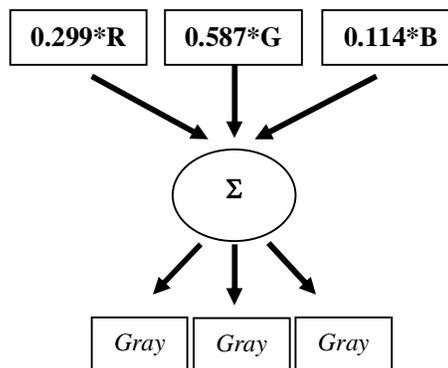
Pengolahan citra adalah suatu metode yang digunakan untuk mengolah gambar sehingga menghasilkan gambar lain yang sesuai dengan keinginan kita². Pengolahan citra berperan sebagai pra pengolahan yang bertugas untuk menyederhanakan gambar menjadi suatu data berupa matrik $m \times n$, kemudian matriks tersebut dijadikan sebagai input bagi jaringan syaraf tiruan

Komputer digital hanya dapat memproses suatu citra dalam bentuk digital. Pengambilan gambar bisa dilakukan oleh penangkap citra digital misalnya, kamera video, scanner dan kamera digital. Atau alat-alat lain yang bisa digunakan untuk mentransfer gambar.

2.3.1 Proses Grayscale

Operasi ini bertujuan untuk merubah citra 24 *bit* RGB menjadi citra abu-abu. Pemilihan pemrosesan pada tingkat abu-abu ini dipilih karena lebih sederhana karena hanya menggunakan sedikit kombinasi warna dan dengan citra abu-abu dirasakan sudah cukup untuk memproses suatu gambar.

Prinsip citra perubahan citra 24 *bit* RGB menjadi citra abu-abu adalah dengan menghitung rata-rata dari intensitas $0.299 * red$, $0.587 * green$, $0.114 * blue$ dari citra 24 *bit* RGB³ seperti ditunjukkan pada gambar 2.



Gambar 2 Operasi perubahan citra 24 *bit* ke citra abu-abu

² Agustinus Nalwan, "Pengolahan Gambar Secara Digital", Elex Media Computindo, Jakarta, 1997

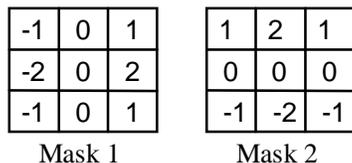
³ Ioannis Pitas, "Digital Image Processing Algorithms", Prentice Hall, Singapore, 1993

2.3.2 Proses Deteksi Tepi

Deteksi Tepi adalah perubahan nilai intensitas derajat keabuan yang besar dalam jarak yang dekat. Biasanya, tepi terdapat pada batas antara dua daerah berbeda pada citra.

Tepi (*edge*) merupakan ciri dasar dari citra, mereka membawa informasi yang sangat berguna tentang pinggiran dari suatu objek yang mana dapat digunakan untuk menganalisa suatu *image* (citra), pengidentifikasian suatu objek maupun untuk aplikasi penyaringan (*filtering*) dari citra. Deteksi tepi ini sangat penting dan merupakan hal paling mendasar dalam pemrosesan citra secara digital.

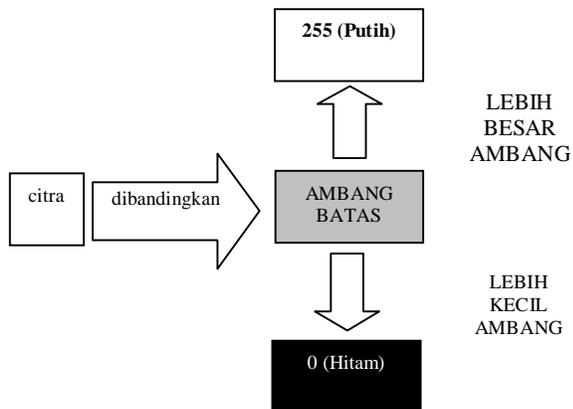
Tujuan deteksi tepi adalah untuk meningkatkan penampakan garis batas objek dalam citra. Ada beberapa metode deteksi tepi yang banyak dipakai dalam pendeteksian tepi. Dalam makalah ini kami memakai deteksi tepi turunan gradien pertama dengan sobel mask operator seperti ditunjukkan pada gambar 3.



Gambar 3 Sobel Mask

2.3.3 Proses Tresholding atau Binarisasi

Proses thresholding atau binarisasi pada prinsipnya adalah melakukan pengubahan nilai intensitas warna piksel menjadi 2 nilai yaitu 0 atau warna hitam dan 255 atau warna putih.



Gambar 4 Ilustrasi pengambilan keputusan pada proses thresholding

Pengubahan diharapkan akan menghasilkan citra yang hanya terdiri dari dua warna saja, sehingga untuk proses selanjutnya menjadi lebih sederhana.

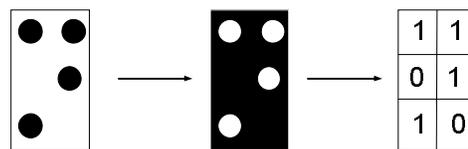
Tujuan utama dari proses thresholding ini adalah untuk memisahkan dan membedakan antara

objek dengan background (latar belakang) dari suatu citra.

2.3.4 Proses Ekstraksi Ciri

Setelah objek-objek yang ada berhasil disederhanakan pada tahap sebelumnya, tahap berikutnya adalah mengekstraksi ciri-ciri tertentu dari setiap objek atau karakter huruf braille. Pada tahap ini mencari posisi gambar paling kiri, paling kanan, paling atas, dan paling bawah.

Kemudian gambar dibagi-bagi menjadi baris M dan kolom N. Setiap kotak dilakukan *scanning pixel*. Jika melebihi jumlah yang ditentukan maka kotak tersebut diberi nilai 1, jika tidak diberi nilai 0. kemudian melakukan penyimpanan matriks MxN sebagai data referensi. Proses ekstraksi ciri dapat dilihat pada gambar 5



Gambar 5 Proses ekstraksi ciri

Dalam makalah ini kami melakukan ekstraksi ciri dengan membagi gambar karakter atau huruf braille menjadi 3 baris dan 2 kolom atau matriks 3x2.

2.4 Algoritma Propagasi balik

Secara garis besar algoritma ini menjelaskan, yaitu ketika jaringan diberikan pola input sebagai pola pelatihan maka pola tersebut menuju ke simpul-simpul pada lapisan tersembunyi untuk diteruskan ke simpul-simpul lapisan output. Kemudian simpul-simpul lapisan output memberikan respon disebut output jaringan.

Saat output jaringan tidak sesuai dengan output yang diharapkan maka output akan menyebar mundur pada lapisan tersembunyi diteruskan ke simpul pada lapisan input. Oleh karena itu maka mekanisme pelatihan tersebut disebut propagasi balik.

Algoritma propagasi balik terdiri dari dua proses yaitu *feed forward* dan propagasi balik dari errornya⁴. Untuk jelasnya dapat dijelaskan sebagai berikut:

- Step 0. Inisialisasi faktor penimbang dengan nilai random yang kecil
- Step 1. Ulangi langkah 2 hingga 9 hingga kondisi stop terpenuhi
- Step 2. Lakukan langkah 3 hingga 8 untuk masing-masing pasangan training Feedforward.

⁴ Adeli H, S. Hung, "Machine Learning, Neural Network, Genetic Algorithm, and Fuzzy Systems, John Willey & Sons Inc., Singapore, 1994

Step 3. Masing-masing unit input (X_i , $i=1, \dots, n$) menerima sinyal input X_i dan sinyal tersebut disebar ke unit bagian atas lapisan tersembunyi (unit hidden).
Step 4. Masing-masing hidden menjumlahkan faktor penimbang :

$$Z_{in_j} = V_{oj} + \sum_{i=1}^n X_i V_{ij}$$

dan menghitung sinyal output sesuai dengan fungsi aktivasi:

$$Z_j = f(Z_{in_j})$$

Karena yang digunakan fungsi sigmoid maka:

$$Z_j = \frac{1}{1 + \exp(-Z_{in_j})}$$

kemudian mengirim sinyal tersebut ke semua unit di atasnya (output unit).

Step 5. Masing-masing unit output (Y_k , $k=1, 2, 3, \dots, m$) dijumlahkan faktor penimbang:

$$Y_{in_k} = W_{ok} + \sum_{j=1}^p Z_j W_{jk}$$

Menghitung sesuai dengan fungsi aktifasi:

$$Y_k = f(Y_{in_k})$$

Propagasi Balik dari Errornya

Step 6. Masing-masing unit output (Y_k , $k=1, 2, 3, \dots, m$) menerima pola target sesuai dengan pola masukan saat training dan menghitung errornya:

$$\delta_k = (t_k - Y_k) f'(Y_{in_k})$$

Menghitung pembedulan faktor penimbang (untuk memperbaiki W_{jk})

$$\Delta W_{jk} = \alpha \delta_k Z_j$$

Menghitung pembedulan koreksi:

$$\Delta W_{ok} = \alpha \delta_k$$

dan mengirimkan nilai δ_k ke unit lapisan di bawahnya.

Step 7. Masing-masing unit hidden (Z_j , $j=1, 2, 3, \dots, p$) menjumlahkan delta inputnya (dari unit layer di atasnya)

$$\delta_{in_j} = \sum_{k=1}^m \delta_k W_{jk}$$

selanjutnya dikalikan dari fungsi aktivasinya untuk menghitung error.

$$\delta_j = \delta_{in_j} f'(Z_{in_j})$$

Kemudian menghitung pembedulan penimbang (digunakan untuk memperbaiki V_{ij})

$$\Delta V_{ij} = \alpha \delta_j X_i$$

kemudian menghitung pembedulan bias (untuk memperbaiki V_{oj})

$$\Delta V_{oj} = \alpha \delta_j$$

Memperbaiki Penimbang dan Bias

Step 8. Masing-masing output unit (Y_k , $k=1, 2, 3, \dots, m$) diperbaiki bias dan penimbangannya ($j=0, 1, 2, \dots, p$)

$$W_{jk}(\text{baru}) = W_{jk}(\text{lama}) + \Delta W_{jk}$$

Masing-masing unit hidden (Z_j , $j=1, 2, 3, \dots, p$) diperbaiki bias dan penimbangannya ($j=0, 1, 2, \dots, n$)

$$V_{jk}(\text{baru}) = V_{jk}(\text{lama}) + \Delta V_{jk}$$

Step 9. Proses berhenti

2.5 Tahap Pelatihan/Pembelajaran

Tahap pelatihan/pembelajaran ini merupakan langkah bagaimana suatu jaringan syaraf itu berlatih, yaitu dengan cara melakukan perubahan bobot sambungan. Sedangkan fase pemecahan masalah akan dilakukan jika proses belajar tersebut selesai, fase tersebut adalah proses pengujian atau testing.

Proses pembelajaran dilakukan dengan membandingkan pasangan output jaringan dengan vektor target. Selisih dari keduanya merupakan error yang akan digunakan untuk mengubah pembobot sambungan. Sehingga kesalahan akan semakin kecil pada siklus pembelajaran berikutnya atau mencapai konvergen.

2.6 Laju pembelajaran (Learning rate)

Learning rate (α) adalah salah satu parameter yang turut mempengaruhi kecepatan belajar. *Learning rate* sangat berpengaruh pada intensitas pada proses belajar, efektivitas dan konvergensi dari pelatihan. Harga optimum, dari *learning rate* tergantung masalah yang dihadapi.

Intinya dipilih sedemikian rupa sehingga dicapai konvergensi yang optimal dalam proses pelatihan. Harga α yang cukup kecil akan menjadikan proses training berjalan dengan lambat, sehingga konvergensi akan tercapai pada iterasi yang besar.

III. PERANCANGAN SISTEM

3.1 Sistem Pengenalan Pola Huruf Braille

Perangkat lunak (*software*) dibuat dengan menggunakan bahasa Pascal, dalam hal ini *Delphi 7.0*, yang dijalankan pada sistem operasi *Windows XP*

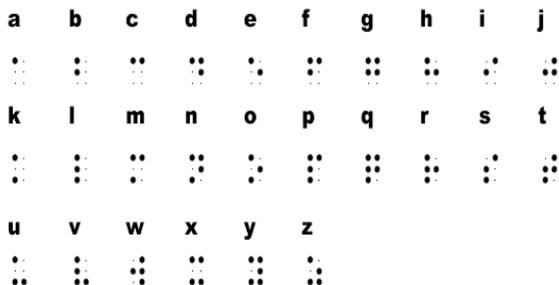
Secara garis besar, ada beberapa proses dalam sistem pengenalan pola-pola huruf braille yaitu:

1. Pengambilan gambar (*file*)
2. Pemrosesan awal (*grayscale, deteksi tepi, thresholding, ekstraksi ciri*)
3. Proses pembelajaran
4. Proses pengenalan atau pengambil keputusan.

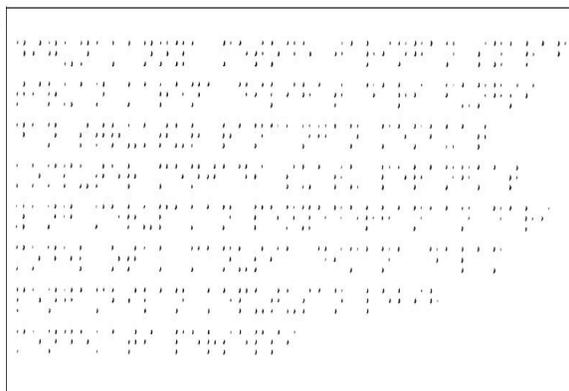
Tujuan dari pembuatan perangkat lunak ini adalah untuk mengenali pola-pola huruf braille yang telah disimpan dalam sebuah file gambar. Gambar pola-pola huruf braille yang akan dikenali adalah gambar yang telah diambil gambarnya dengan menggunakan *scanner*, dalam hal ini pengambilan gambar dilakukan secara *offline*.

Pengambilan gambar tersebut harus lurus dan tidak dalam posisi terbalik. Pola-pola huruf braille dalam hal ini berbentuk suatu teks.

File gambar huruf braille yang akan diproses dapat berformat **.bmp* dengan 24 bit *perpixel*.

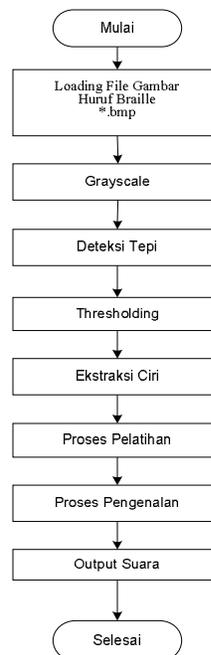


Gambar 6 Pola Huruf Braille

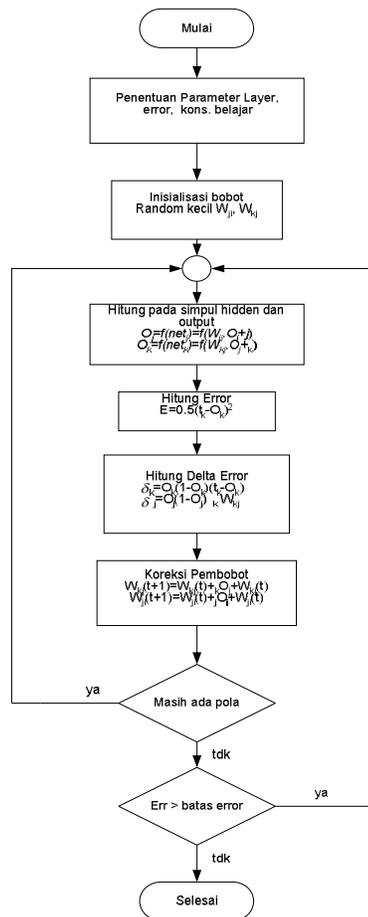


Gambar 7 Teks Braille Hasil Scan

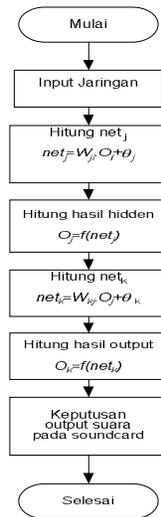
3.2 Diagram Alir Sistem



Gambar 8 Diagram Alir Perangkat Lunak Utama



Gambar 9 Diagram Alir Tahap Pembelajaran



Gambar 10 Diagram Alir Proses Pengambil Keputusan

3.3 Desain Arsitektur

Pada tugas akhir tersebut kami menggunakan layer input sebanyak 6 titik, layer hidden sebanyak 25 titik, dan layer output sebanyak 27 titik. Mengapa demikian? Karena kami mempunyai target 26 karakter huruf abjad dan 1 karakter spasi yang akan dikenali. Sedangkan banyaknya jumlah titik dalam hidden layer dipilih berdasarkan kesesuaian dengan hasil error rate dalam jaringan.

IV. PENGUJIAN DAN ANALISA

4.1 Pengujian Perangkat Lunak

Pengujian perangkat lunak memiliki langkah (step) yang telah ditentukan secara urut. Urutan dalam pengujian langkah demi langkah harus secara teratur, tidak dapat melewati salah satu langkah untuk menuju langkah lainnya.

Pengujian perangkat lunak meliputi:

1. Uji coba proses loading data
2. Uji coba proses grayscale
3. Uji coba proses deteksi tepi
4. Uji coba proses threshold
5. Uji coba proses ekstraksi ciri
6. Uji coba proses pembelajaran
7. Uji coba proses pengenalan

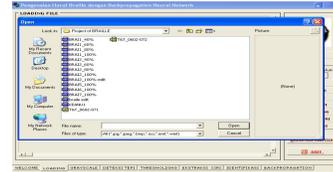
4.1.1 Hasil Uji Coba Loading Data

Pada proses yang paling awal adalah loading file, dimana data yang dapat diambil adalah file dengan ekstensi *.bmp. Untuk membuka file gambar dapat menggunakan pilihan pada menu utama seperti ditampilkan pada gambar 10



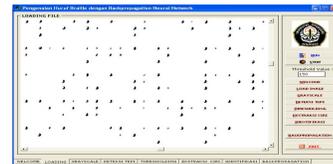
Gambar 11 Tampilan Menu Utama Program

Kemudian membuka Load Image dengan tampilan dialog seperti ditampilkan pada gambar 11



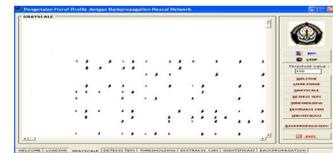
Gambar 12 Proses loading file BRAII_100%.BMP

Hasil loading file nya ditunjukkan pada gambar 12



Gambar 13 Hasil Loading FileBRAII_100%.BMP

4.1.2 Hasil Uji Coba Proses Grayscale



Gambar 14 Hasil Uji Coba Proses Grayscale

4.1.3 Hasil Uji Coba Proses Deteksi tepi



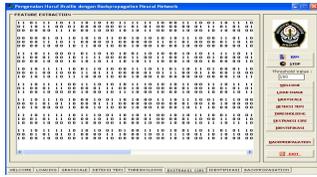
Gambar 15 Hasil Uji Coba Proses Deteksi Tepi

4.1.4. Hasil Uji Coba Proses Thresholding



Gambar 16 Hasil Uji Coba Proses Thresholding

4.1.5. Hasil Uji Coba Proses Ekstraksi Ciri

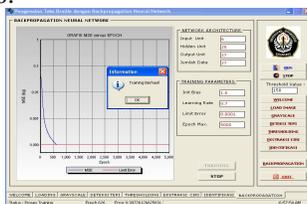


Gambar 17 Hasil Uji Coba Proses Ekstraksi Ciri

4.1.6 Uji Coba Pembelajaran

Jumlah pola huruf braille yang ditraining sebanyak 27 data yaitu meliputi karakter a sampai z dan ditambah karakter untuk spasi. Pada uji coba pembelajaran mencapai 626 kali iterasi dengan batasan error sebesar 0.0001

Gambar 10 merupakan gambar grafik error dari proses pembelajaran jaringan syaraf tiruan. Pada gambar terdapat garis merah yaitu menunjukkan batasan error (error limited) sebesar 0.0001, dan garis biru adalah menunjukkan perubahan error selama proses pembelajaran berlangsung. Pembelajaran akan berhenti jika hasilnya mencapai konvergen yaitu pada iterasi ke-626.

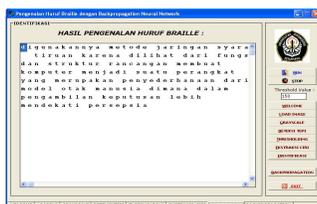


Gambar 18 Grafik Uji Pembelajaran

4.1.7 Uji Pengenalan

Tahap terakhir dari perangkat lunak ini adalah tahap pengenalan. Setelah dilakukan langkah-langkah sebelumnya yaitu grayscale, deteksi tepi, threshold, serta ekstraksi ciri dengan secara berurutan dan teratur, kemudian dilakukan tahap pengenalan yang merupakan implementasi dari jaringan syaraf tiruan.

Hasil dari pengenalan dikeluarkan dalam bentuk suara melalui soundcard pada PC. Suara yang dikeluarkan akan mengeja huruf demi huruf yang dikenali. Jumlah pola yang akan dikenali berjumlah 27 pola yaitu mulai a sampai z termasuk spasi. Selain output suara yang dikeluarkan melalui soundcard, juga terdapat tampilan pada layar seperti pada gambar 18.



Gambar 19 Hasil Pengenalan Dengan Jaringan Syaraf Tiruan File Gambar BRAII_100%.bmp

4.2 Analisa

Jika dilihat dari pola-polanya, huruf Braille berbeda dengan bentuk abjad biasa. Satu karakter huruf Braille berbentuk seperti kode-kode tertentu yaitu berupa bulatan-bulatan kecil sebanyak 3x2, serta dibuat secara timbul agar karakter tersebut dapat dikenali oleh tunanetra

Setelah melakukan beberapa pengujian pengenalan pola huruf Braille sebanyak 10 kali dengan menggunakan 2 buah gambar hasil pengambilan dengan menggunakan scanner, didapatkan hasil pengenalan seperti ditunjukkan pada tabel 1 dan dan tabel 2 .

Dari percobaan yang dilakukan sebanyak 10 kali dengan gambar yang sama, hasilnya sama seperti pada pertama kali percobaan pengenalan dilakukan. Artinya jaringan tetap dapat mengenali sesuai dengan apa yang telah jaringan pelajari pada saat pembelajaran

Tabel 1 Hasil Pengenalan BRAII_100%.BMP

No	Pengujian ke-	Jumlah Karakter	Dikenali	Tidak Dikenali	Prosentase Pengenalan
1	1	217	216	1	99,54%
2	2	217	216	1	99,54%
3	3	217	216	1	99,54%
4	4	217	216	1	99,54%
5	5	217	216	1	99,54%
6	6	217	216	1	99,54%
7	7	217	216	1	99,54%
8	8	217	216	1	99,54%
9	9	217	216	1	99,54%
10	10	217	216	1	99,54%

Tabel 2 Hasil Pengenalan BRAII2_100%.BMP

No	Pengujian ke-	Jumlah Karakter	Dikenali	Tidak Dikenali	Prosentase Pengenalan
1	1	306	303	3	99,02%
2	2	306	303	3	99,02%
3	3	306	303	3	99,02%
4	4	306	303	3	99,02%
5	5	306	303	3	99,02%
6	6	306	303	3	99,02%
7	7	306	303	3	99,02%
8	8	306	303	3	99,02%
9	9	306	303	3	99,02%
10	10	306	303	3	99,02%

Tabel 3 Total Hasil Pengenalan BRAII_100%.BMP

Pengujian Gambar	Jumlah Karakter	Dikenali	Tidak Dikenali	Prosentase Pengenalan
1	217	216	1	99,54%
2	306	303	3	100 %
Total	523	519	1	99,28%

Pada tabel 3 total karakter yang dapat dikenali adalah 99,28 %. Masih adanya kesalahan dalam pengenalan pola dapat terjadi karena pemisahan karakter yang kurang akurat. Hal lain juga dapat disebabkan kertas yang dipakai untuk penulisan huruf braille tersebut dalam keadaan kusut (lungset) atau kertas dalam keadaan kotor. Keadaan tersebut akan menjadikan proses ekstraksi ciri dapat menghasilkan data yang berbeda dengan gambar.

Hasil akhir pengenalan huruf Braille dengan menggunakan jaringan syaraf tiruan ini berupa suara yang mengeja huruf demi huruf. Perekaman suara dilakukan dengan memanfaatkan fasilitas Microsoft Sound Recorder. Suara hasil rekaman memiliki format (*.wav) yang disimpan dalam file c:\voice sebanyak 26 huruf yaitu huruf A-Z. Pemanggilan prosedur atau fungsi untuk menjalankan suara sesuai dengan output hasil identifikasi perhurufnya telah disediakan oleh Delphi yaitu SndPlaySound pada unit MMSsystem.

V. KESIMPULAN

5.1. Kesimpulan

1. Dari hasil pengujian 2 buah gambar dengan menggunakan scanner yang dilakukan percobaan masing-masing gambar sebanyak 10 kali, didapatkan bahwa unjuk kerja dari perangkat lunak ini dapat memberikan rata-rata hasil pengenalan sebesar 99,28%.
2. Pemilihan nilai parameter learning rate dapat mempengaruhi dalam lamanya proses pembelajaran. Semakin tinggi learning rate semakin cepat pula proses iterasi berlangsung dan begitu juga sebaliknya. Dengan learning rate sebesar 0,7 yang kami pakai dalam percobaan, iterasi yang berlangsung sebanyak 626 kali pembelajaran
3. Kesalahan pengenalan disebabkan pemisahan karakter yang kurang akurat dan juga kualitas kertas yang digunakan untuk penulisan huruf braille tersebut kurang baik sehingga output jaringan lebih mirip dengan target lain yang diharapkan
4. Hasil pengenalan akan dikeluarkan melalui soundcard dalam bentuk suara per huruf sehingga dapat membantu tunanetra dalam membaca huruf braille tanpa meraba dengan tangan.

5.2. Saran

Berikut adalah saran-saran yang berkaitan dengan penelitian yang telah dilakukan.

1. Perangkat lunak dapat dikembangkan lebih lanjut untuk menghasilkan suara yang mengeja hasil pengenalan per kata.
2. Perkembangan selanjutnya diharapkan menggunakan video camera secara online.
3. Agar pendeteksian karakter lebih akurat, pembagian gambar atau karakter menjadi baris dan kolom jumlahnya lebih banyak lagi.

DAFTAR PUSTAKA

- [1] Ir. Son Kuswadi, Dr. Mohammad Nuh, DEA, Ir. Anang Tjahjono, Riyanto Sigit, ST, "Elektronika Terapan", Diktat Mata Kuliah, PENS-ITS, Surabaya, 1998.
- [2] Freeman, JA, Davis Shapura, "Neural Network: Algorithm, Application and Tecnique Programming", Addison Wesley Publishing Company, Massachussets, 1991
- [3] Purnomo, Mauridhi Hery, "Pengantar Jaringan Syaraf Tiruan (Neural Networks)", Diktat Pelatihan Neural Network, Surabaya, 1999
- [4] Fausett, Laurence, "Fundamentals of Neural Network Architecture, Algorithms, and Application", Prentice Hall International Inc., Englewood Cliffs N.J, 1994
- [5] Adeli H, S. Hung, "Machine Learning, Neural Network, Genetic Algorithm, and Fuzzy Systems", John Willey & Sons Inc., Singapore, 1994
- [6] Agustinus Nalwan, "Pengolahan Gambar Secara Digital", Elex Media Komputindo, Jakarta, 1995
- [7] Ioannis Pitas, "Digital Image Processing Algorithms", Prentice Hall, Singapore, 1993
- [8] Tuesday T.A.S. Rondonuwu, "Kontrol Kualitas Keragaman Warna Produk Industri Dengan Metode Neural Network (Algoritma Backpropagation)", Tugas Akhir Program D IV Teknik Elektronika Instrumentasi Jurusan Elektro ITB, Surabaya, 2000.



Praditya Firmansyah [L2F303467]
Lahir di Pamekasan, 14 Juli 1980
Mahasiswa Teknik Elektro Ekstensi
2003, Bidang Konsentrasi Elektronika
dan Telekomunikasi,
Universitas Diponegoro Semarang
Email : DLT_PDF@yahoo.com

Semarang,.....Maret 2006

Menyetujui dan Mengesahkan

Pembimbing I,

Pembimbing II,

Wahyul AS ST, MT.
NIP. 132 137 934

Iwan Setiawan ST, MT.
NIP. 132 283 183