

IMPLEMENTASI BACKPROPAGATION DALAM PENGOLAHAN CITRA TEKS TULISAN TANGAN MENJADI TEKS DIGITAL

Fifin Hietania, Wayan Santiyasa, Ida Bagus Gede Dwidasmara
Program Studi Teknik Informastika, Jurusan Ilmu Komputer,
Fakultas Matematika Dan Ilmu Pengetahuan Alam, Universitas Udayana
Email : fifin_hie@yahoo.co.id

ABSTRAK

Pengenalan pola adalah hal yang cukup sulit dilakukan oleh komputer. Komputer tidak dapat langsung mengenali pola setiap citra. Maka dari itu diperlukan suatu proses ataupun alat yang dapat digunakan untuk pengolahan citra di dalam suatu komputer.

Pada penelitian tugas akhir ini akan dibuat suatu aplikasi yang berhubungan dengan pengenalan pola dan pengolahan citra. Yaitu aplikasi yang dapat mengenali pola tulisan tangan dan mengubahnya menjadi teks digital. Aplikasi ini menggunakan metode backpropagation dalam proses pengenalannya. Dimana input yang berupa citra berekstensi bitmap akan diolah terlebih dahulu melalui serangkaian proses, yang disebut dengan tahap preprocessing. Dimana tahap ini terdiri dari Greyscale, thresholding, segmentasi, cropping, dan normalisasi. Kemudian setelah preprocessing barulah tahap pengenalan akan dilakukan.

Adapun keakuratan hasil yang didapat dari proses ini sebesar 67,99%. Yang didapat dari hasil uji coba 50 data uji, dengan menggunakan 660 data referensi, learning rate = 0,001, error rate = 0,01, epoch = 100, inialisasi bobot awal = -0,5 sampai 0,5, input pattern = 30 x 30 piksel, input neuron = 900, hidden layer 55, dan output neuron = 900. Dimana dapat disimpulkan bahwa aplikasi ini cukup baik dalam proses pengenalan pola tulisan tangan.

Kata kunci : Backpropagation, Pengolahan citra, Pengenalan pola, Tulisan tangan, Teks digital

ABSTRACT

Computer found it difficult enough to introduce patterns. It could not directly recognize the pattern of every image. Therefore, a process or equipment was needed to process images in computer.

In this present study, as the final assignment, an application to how to introduce patterns and to process images was created, that is, an application which could recognize handwriting and change it into digital text, using backpropagation method. The input which was in the form of bitmap extended image was firstly processed through a set of processes referred to as a preprocessing stage, which was made up of Greyscale, thresholding, segmentation, cropping, and normalization. After the preprocessing stage, the recognizing stage was done.

The degree of accuracy obtained from this process was 67.99%, which was obtained from the trial test of 50 test data, using 660 reference data, learning rate 0.001, error rate 0.01, epoch 100, initial weight initialization -0.5, input pattern 30 x 30 pixel, input neuron 900, hidden layer 55, and output neuron 900. It could be concluded that this application was good enough in the process recognizing the pattern of handwriting.

Keywords: Backpropagation, Image processing, Pattern recognition, Hand writing, Digital text

1. PENDAHULUAN

Perkembangan teknologi dewasa ini dapat dikatakan sebagai sumber dari perkembangan hampir semua aspek kehidupan. Teknologi yang berkembang demikian pesatnya tidak terlepas dari pengaruh komputerisasi. Sistem komputerisasi membuat semua berjalan dengan sangat cepat, sehingga komputer menjadi alat yang sangat penting dalam setiap proses pengolahan informasi.

Pengenalan pola adalah sesuatu yang mudah untuk dilakukan oleh manusia, namun membuat komputer agar dapat 'membaca' adalah suatu hal yang benar-benar sulit. Secara mudahnya kita dapat membaca isi tulisan pada citra tersebut dengan mudah namun tidak begitu halnya dengan komputer. Maka dari itulah diperlukan suatu perangkat lunak yang dapat mengolah citra menjadi teks digital.

Penelitian ini didasarkan kepada penelitian sebelumnya yang berjudul "Pembuatan Perangkat Lunak Identifikasi Huruf dengan Metode Bidirectional Associative Memory" (Pipin Piniman, 2008). Namun pada penelitian ini disebutkan bahwa metode BAM masih kurang stabil untuk mengenali berbagai jenis variasi karakter. Kemudian berdasarkan penelitian yang berjudul "Pengembangan Aplikasi Pengenalan Karakter Alfanumerik Dengan Menggunakan Algoritma *Neural Network Three-Layer Backpropagation*" (Emanuel, 2008). Hanya saja pengenalan

tersebut terbatas pada satu karakter saja dan tidak dapat mengkonversi tulisan yang berupa kalimat.

Berdasarkan kelemahan dari penelitian-penelitian sebelumnya maka pada penelitian kali ini dibuat suatu aplikasi yang dapat mengkonversi teks tulisan tangan menjadi teks digital menggunakan metode Backpropagation.

2. DASAR TEORI

2.1 Citra

Jika kita kembali meninjau secara matematis, Citra adalah sebuah fungsi intensitas cahaya dua dimensi $f(x,y)$ dimana x adalah posisi baris dan y adalah posisi kolom sedangkan F adalah fungsi intensitas atau kecerahan dari citra pada koordinat (x,y) . Pada umumnya citra berbentuk persegi panjang dan dimensi ukurannya dinyatakan sebagai (tinggi x lebar). Citra dengan tinggi N piksel, lebarnya M piksel, dan memiliki intensitas f dapat dipresentasikan sebagai suatu matriks yang berukuran N baris dan M kolom sebagai berikut :

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

Indeks baris (x) dan indeks kolom (y) menyatakan suatu koordinat titik pada citra. Masing-masing titik (x,y) di citra disebut *picture element* atau piksel, sedangkan $f(x,y)$ merupakan intensitas (derajat keabuan) pada titik (x,y) .

2.2 Grayscale

Dalam komputasi, suatu citra digital grayscale atau greyscale adalah suatu citra dimana nilai dari setiap pixel merupakan sample tunggal. Citra grayscale seringkali merupakan perhitungan dari intensitas cahaya pada setiap pixel pada spektrum elektromagnetik single band.

Cara yang cukup mudah untuk mengubah citra berwarna menjadi grayscale adalah dengan menggunakan rumus :

$$\text{Grayscale} = 0,299R + 0,587G + 0,114B$$

2.3 Thresholding

Thresholding merupakan teknik yang sederhana dan efektif untuk segmentasi citra. Proses *thresholding* sering disebut dengan proses binerisasi. Setiap piksel di dalam citra dipetakan dengan dua nilai, satu (1) atau nol (0) dengan fungsi pengambangan :

$$f_b(i, j) = \begin{cases} 0 & \text{if } f(i, j) \geq T \\ 1 & \text{if } f(i, j) \leq T \end{cases}$$

2.4 Segmentasi

Segmentasi merupakan teknik untuk membagi suatu citra menjadi beberapa daerah (region) di mana setiap daerah memiliki kemiripan atribut. Segmentasi bertujuan untuk memecah suatu citra ke dalam beberapa segmen dengan suatu kriteria tertentu. Proses segmentasi ini merupakan bagian yang sangat penting dan biasanya merupakan proses yang rumit dalam pemrosesan Citra Digital.

2.5 Cropping

Cropping pada pengolahan citra berarti memotong satu bagian dari citra sehingga

diperoleh citra yang diharapkan. Yang diperlukan dalam cropping suatu citra adalah menentukan koordinat dari citra tersebut. Kemudian menentukan batas dari citra yang akan di crop. Untuk menghitung suatu luasan objek yang akan dihitung maka ukuran pemotongan dapat dirumuskan sebagai berikut :

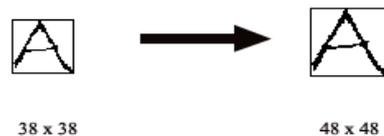
$$W' = (x_r - x_l) + 1$$

$$H' = (y_b - y_t) + 1$$

2.6 Normalisasi

Normalisasi pada pengolahan citra berarti mentransformasikan citra ke bentuk citra normal yang sesuai dengan kebutuhan. Suatu citra setelah di normalisasi, dapat dilakukan proses penskalaan (scaling) agar nilai fitur berada pada suatu interval tertentu. Penskalaan citra dirumuskan sebagai berikut :

$$\begin{aligned} x' &= S_x \cdot x \\ \dots y' &= S_y \cdot y \dots \dots \dots 2.5 \end{aligned}$$



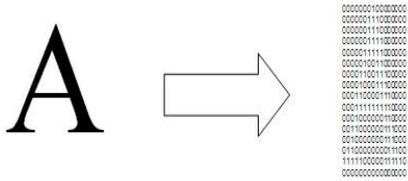
Gambar 1. Normalisasi huruf 'A'

2.7 Ekstraksi Ciri

Ekstraksi ciri pada pengolahan citra berarti mengubah nilai-nilai intensitas koordinat piksel yang terdapat dalam citra menjadi susunan kode-kode nilai pada setiap piksel.

Ekstraksi ciri dapat dirumuskan sebagai berikut :

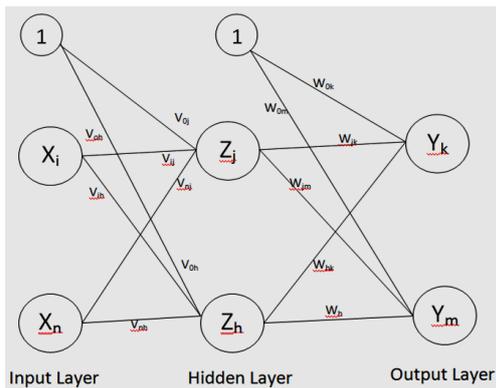
$$f(i, j) = \begin{cases} 1 & \text{if } f(i, j) = 0 \\ -1 & \text{if } f(i, j) \text{ lainnya} \end{cases}$$



Gambar 2. Hasil ekstraksi ciri huruf 'A'

2.8 Backpropagation

Backpropagation 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.



Gambar 3. Arsitektur Backpropagation

Keterangan :

X = Masukan (input).

Y = Keluaran hasil.

Z = Hidden Layer

v = Bobot pada lapisan tersembunyi.

w = Bobot pada lapisan keluaran.

i = input ke- i (1-400)

j = hidden layer ke- j (1- h)

k = output ke- k (1- o)

n = Jumlah input maksimum (400)

h = Jumlah neuron hidden layer maksimum

m = Jumlah output maksimum

V_{ij} merupakan bobot garis dari unit masukan X_i ke unit layar tersembunyi Z_j (V_{0j} merupakan bobot garis yang menghubungkan bias di unit masukan ke unit layar tersembunyi Z_j). W_{jk} merupakan bobot dari unit layar tersembunyi Z_j ke unit keluaran Y_k (W_{0k} merupakan bobot dari bias di layar tersembunyi ke unit keluaran Z_k).

Fungsi aktivasi yang digunakan dalam aplikasi ini adalah Fungsi sigmoid bipolar yang didefinisikan sebagai berikut.

$$f_2(x) = \frac{2}{1 + \exp(-x)}$$

$$f_2'(x) = \frac{1}{2} [1 + f_2(x)][1 - f_2(x)]$$

Pelatihan untuk jaringan *backpropagation* dengan satu layar tersembunyi (dengan fungsi aktivasi sigmoid bipolar) adalah sebagai berikut:

1. Langkah 0

Inisialisasi semua bobot dengan bilangan acak kecil.

2. Langkah 1

Jika kondisi penghentian belum dipenuhi, lakukan langkah 2-8.

3. Langkah 2

Untuk setiap pasang data pelatihan, lakukan langkah 3-8.

4. Langkah 3 (langkah 3-5 merupakan fase *feedforward*)

Input unit (X_i) menerima sinyal *input* (x_i) dan menyebarkan sinyal itu ke *hidden layer*.

5. Langkah 4

Setiap unit tersembunyi ($x_i, i = 1, \dots, p$) jumlahkan bobot sinyal masukannya,

$$z_{in_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij}$$

v_{0j} = bias pada unit tersembunyi j aplikasikan fungsi aktivasinya untuk menghitung sinyal keluarannya, $z_j = f(z_{in_j})$, dan kirimkan sinyal ini keseluruh unit pada lapisan diatasnya (unit keluaran). Terapkan fungsi aktivasi dan hasilnya dikirim ke *output unit*.

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

6. Langkah 5

Tiap unit keluaran ($y_k, k = 1, \dots, m$) jumlahkan bobot sinyal masukannya,

$$y_{in_k} = w_{0k} + \sum_{j=1}^p z_j w_{jk}$$

w_{0k} = bias pada unit keluaran k dan aplikasikan fungsi aktivasinya untuk menghitung sinyal keluarannya,

$$y_k = f(y_{in_k})$$

7. Langkah 6 (langkah 6-7 merupakan fase *backpropagation*)

Tiap unit keluaran ($y_k, k = 1, \dots, m$) menerima pola target yang saling berhubungan pada masukan pola pelatihan, hitung kesalahan informasinya,

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

t_k = Target keluaran

hitung koreksi bobotnya (digunakan untuk memperbaharui w_{jk} nantinya),

$$\Delta W_{jk} = \alpha \delta_k z \tag{2.13}$$

hitung koreksi biasnya (digunakan untuk memperbaharui w_{0k} nantinya), dan kirimkan δ_k ke unit-unit pada lapisan dibawahnya,

$$\Delta W_{0k} = \alpha \delta_k \tag{2.14}$$

8. Langkah 7

Setiap unit lapisan tersembunyi ($z_j, j = 1, \dots, p$) jumlahkan hasil perubahan masukannya (dari unit-unit lapisan diatasnya),

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

Kalikan dengan nilai fungsi turunan aktivasi untuk menghitung *error*,

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

Hitung nilai perbaikan bobot, (v_{ij}),

$$\Delta V_{ij} = \alpha \delta_j x_i \tag{2.16}$$

Hitung nilai perbaikan bias (v_{0j}),

$$\Delta V_{0j} = \alpha \delta_j$$

9. Langkah 8 (fase perubahan bobot)

Hitung semua perubahan bobot. Perubahan bobot garis yang menuju ke unit keluaran (Y_k), yaitu:

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

Setiap *hidden unit* (Z_j) memperbaharui bobot dan biasnya:

$$V_{ij}(baru) = v_{ij}(lama) + \Delta v_{ij}$$

Parameter α merupakan laju pemahaman yang menentukan kecepatan iterasi. Nilai α terletak antara 0 dan 1 ($0 \leq \alpha \leq 1$). Semakin besar harga α , semakin sedikit iterasi yang dipakai. Akan tetapi jika harga α terlalu besar, maka akan merusak pola yang sudah benar sehingga pemahaman menjadi lambat. Satu siklus pelatihan yang melibatkan semua pola disebut *epoch*.

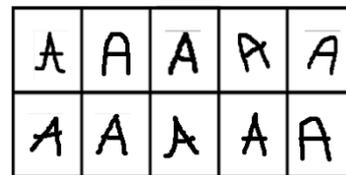
Pemilihan bobot awal sangat mempengaruhi jaringan syaraf tiruan dalam mencapai minimum global (atau mungkin lokal saja) terhadap nilai *error* (kesalahan) dan cepat tidaknya proses pelatihan menuju kekonvergenan. Apabila bobot awal terlalu besar maka *input* (masukan) ke setiap lapisan tersembunyi atau lapisan *output* (keluaran) akan jatuh pada daerah dimana turunan fungsi sigmoidnya akan sangat kecil. Apabila bobot awal terlalu kecil, maka *input* (masukan) ke setiap lapisan tersembunyi atau lapisan *output* (keluaran) akan sangat kecil. Hal ini akan menyebabkan proses pelatihan berjalan

sangat lambat. Biasanya bobot awal diinisialisasi secara random dengan nilai antara -0.5 sampai 0.5 (atau -1 sampai 1 atau interval yang lainnya).

Setelah pelatihan selesai dilakukan, jaringan dapat dipakai untuk pengenalan pola. Dalam hal ini, hanya propagasi maju (langkah 4 dan 5) saja yang dipakai untuk menentukan keluaran jaringan.

3. HASIL DAN PEMBAHASAN

Pada pengujian aplikasi ini digunakan data training (data referensi) sebanyak 660 buah data, yang terdiri dari 10 jenis variasi untuk setiap karakternya.



Gambar 4. Data referensi huruf 'A'

Sedangkan data uji yang digunakan adalah sebanyak 50 buah data uji. Berikut ini adalah hasil yang didapat dari percobaan aplikasi yang dilakukan.

NO	INPUT	HASIL	PRESENTASE PENGENALAN
1	FIFIN HIE 1202	IIFIN HIE FEE2	66,67%
2	Eddy Setiawan 2704	Edd2 EEtIeI0n t70H	56,25%
3	AGNES TASIA 2708	AGNEI T7EI7 27EI	57,14%
4	MerLyna roSalIA. 0798	MerLlna r0Salii, a708	73,68%
5	Sintya DWI 1309.	3intya DMI 13rS	69,23%
6	Septian Bagus 0835	Seltian 8asus 083s	68,75%
7	Dian TRISNA 0860	DiDn TIISia D860	57,14%

8	INDRA Prastiwi 0865	INORi PrisIlwi D86S	52,94%
9	Hendra K4r3n!?	!en3re rKr3n!?	61,54%
10	FeYIn 210566	reY!n 2!!0566	54,54%
11	Muni Adriyani 0846	Munl Adrlyoni 0B4G	62,5%
12	Ilmu Komputer o8	IImu Komeuter o8	78,57%
13	Fakultas MIPA	Fakulta5 Mlp4	66,67%
14	tEkNik iNfOrMaTiKa	tEINIK iNp0yMaTiKa	76,47%
15	Nim. 0808605044	Nim. oBo8c05044	71,42%
16	12 Februari 1990	12 Fcbruarl 1g9o	71,42%
17	Zodiak Aquarius	Zadiak A9uaRli5	64,28%
18	Universitas UDAYANA	Uniyer7itas U0AvAnA	72,22%
19	Bukit Jimbaran	8u7iL Jimbacen	61,54%
20	PROGRAM STUDI	eR06RAM gTuDI	50%
21	Back Propagation!	BacIpr0ea9tlon!	62,5%
22	KOMPUTASI	IDmpuIaS!	55,55%
23	Pengolahan CITRA	Pen90Iohan CITRA	66,67%
24	NetWork Arcitectur3	NelmorI Arilteetui3	61,11%
25	eKstraksi Ciri?	eIs7yaIsi Cili?	66,67%
26	GraysCaLE	6yay5laL3	44,44%
27	Thresholding 196..?	Tkyesh0lain9 196..!?	72,22%
28	Konversi T3k5	K0nversl T3l5	58,33%
29	Fungsi Aktivasi, x, y	Fun9sl AhtIva5i,x,x	66,67%
30	Arsitektur Jaringan	ArsiichtuR Jarln9an	72,22%
31	Jumlah Neuron Go0	Junlai neur0n go0	66,67%
32	Im4g3 Array	Im493 4rroy	70%
33	DK 1804 UC	DK 1B04 UC	87,5%

34	NetWork Arctectur3	NetW0rk ArLtectur3	83,33%
35	SMAN 4 Denpasar	5MAN 4 Deneaaar	76,92%
36	Twlllght 5464	TwIII9ii S464	50%
37	NEW MODN	NEW MODN	85,71%
38	Eclipse	EcLipse	100%
39	Breaking Dawn	8rcaKIng Dawn	75%
40	Stephenv Meyer	Stphenv Meycy	61,53%
41	HaRRy POTTer o7	HaRRv POTTer o7	75%
42	Jk. Rowling	J.T. R0wlln9	63,63%
43	Memoirs Of GEISHA	Nem0ir5 Of 6EISHA	60%
44	King Arthur !!!	Kin6 Aythur!!!	84,61%
45	Source Cod3	5ourcc c0d3	60%
46	Algoritma Pemrograman	A190tlmI QenrogramaN	65%
47	PenGolah4n Citr4...	een6ol4htn C1ty4...	72,22%
48	DiGitAl Library	Di6itAl Librarr	5,71%
49	Wisuda ke 101	Wi5ud0 ke 1o1	72,72%
50	Teks.Digital	Teh5.Digital.	84,61%
Rata-rata			67,99%

Persentase pengenalan dihitung dengan cara

:

$$P = \frac{b}{s} \times 100\%$$

Dimana :

P = Persentase pengenalan

b = Jumlah karakter benar

s = Jumlah keseluruhan karakter

Sedangkan rata-rata pengenalan karakter dihitung dengan cara :

$$R = \frac{\sum_{n=1}^{50} \frac{b}{s} \times 100\%}{n}$$

4. KESIMPULAN DAN SARAN

4.1 Kesimpulan

Kesimpulan yang didapat dari analisis aplikasi koversi tulisan tangan ke citra digital adalah :

1. Algoritma Backpropagation cukup baik digunakan dalam proses pengenalan karakter tulisan tangan. Dengan

menggunakan acuan variable learning rate = 0,001, Error rate = 0,01, Maksimum epoch = 100, inialisasi bobot awal = 0,5, input patern = 30 x 30 piksel, output patern =30 x 30 piksel, input neuron = 900, output neuron = 900, serta hidden layer = 55, data referensi =660, data citra uji = 50. Dari acuan nilai-nilai variable tersebut maka didapatkan akurasi sebesar 67,99%.

2. Proses pengenalan karakter sangat ditentukan oleh data referensi yang dimiliki oleh program. Semakin banyak data citra referensinya, semakin baik pula proses pengenalan karakter oleh program.

4.2 Saran

Saran yang dapat penulis berikan adalah:

1. Perbanyak data citra referensi agar proses pengenalan karakternya menjadi lebih baik.
2. Program ini tidak memiliki database, diharapkan untuk kemudian hari program ini dapat dikembangkan lagi. Terutama untuk pembuatan database, sehingga setiap proses pengenalan huruf dapat dilakukan tanpa harus memasukan data citra referensi terlebih dahulu.

5. DAFTAR PUSTAKA

[1] Aprijani,Dwi Astuti. 2011. "Aplikasi Jaringan Syaraf Tiruan Untuk Mengenali Tulisan Tangan Huruf A,B,C,D Pada Jawaban Soal

Pilihan Ganda". Jurnal Matematika, Sains dan Teknologi Volume 12 nomor 1.

- [2] Arief, Achmad Fauzi. 2010. "Perangkat Lunak Konversi Tulisan Tangan Menjadi Teks Digital"
- [3] Emanuel, Andi Wahyu Rahardjo dan Arie Hartono. 2008. "Pengembangan Aplikasi Pengenalan Karakter Alfanumerik Dengan Menggunakan Algoritma Neural Network Three-Layer Backpropagation". Jurnal Informatika (4), 49 – 58
- [4] Hermawan, Arief. 2006. Jaringan Syaraf Tiruan teori dan Aplikasi. Yogyakarta : ANDI
- [5] Fausett, L. 1994. Fundamentals of Neural Network. Prentice-Hall International, Inc., Florida Institut of Technology.
- [6] Fauziah, M Iwan Wahyuddin. 2009. "Metode Jaringan Saraf Tiruan Penjejukan Balik Untuk Pengenalan Huruf Cetak Pada Citra Digital" . Jurnal Artificial, ICT Research Center UNAS (3)
- [7] Murni,A.,1992.Pengantar Pengolahan Citra. Jakarta : PT.Elex Media Komputindo.
- [8] Nugraha, A.P. dan A.B. Mutiara . 2002. "Metode ekstraksi data untuk pengenalan huruf dan angka tulisan tangan dengan menggunakan

- jaringan syaraf buatan propagasi balik”
- [9] Putra, Dharma. 2010. Pengolahan Citra Digital. Yogyakarta : ANDI
- [10] Puspitaningrum,Diyah.2006. Pengantar Jaringan Syaraf Tiruan. Yogyakarta : ANDI
- [11] Piniman,Pipin.2008. ”Pembuatan Perangkat Lunak Identifikasi Huruf Menggunakan Metode Bidirectional Associative Memory”. Jurnal Tugas Akhir
- [12] Zukhri,Zainudin. 2003. Dasar-Dasar Pemrograman Visual Dengan Delphi 6.0.Yogyakarta : Graha Ilmu