

PENGENALAN POLA ALPHABET TULISAN TANGAN SECARA ON-LINE DENGAN JST-BP

Tole Sutikno, Andhy Fathurrakhman

Program Studi Teknik Elektro Universitas Ahmad Dahlan
Kampus III Jln. Prof Soepomo, Janturan, Yogyakarta
Telp. (0274) 379418 Ext. 220, HP. 08562862810, Fax. (0274) 381523
email: tholes2000@yahoo.com, andhyf@yahoo.com

Abstract

The intelligent system based in the backpropagation artificial neural network and image processing with GUI Matlab 7.0 was developed. This system is designed for on-line pattern recognition. In this experiments, system are trained dan tested using capital alfabet handwriting patterns that captured from on-line webcam automatically. Generally, the results of the experiments show that the system could recognize with a high accuracy.

Keywords: Neural Network, Matlab, Pattern Recognition, Webcam

Abstrak

Telah dibangun sistem cerdas berbasis jaringan syaraf tiruan dan pengolahan citra dengan GUI Matlab 7.0. Sistem ini dirancang untuk pengenalan pola on-line. Pada eksperimen ini, sistem dilatih dan diuji dengan pola tulisan tangan kapital, yang di-capture otomatis oleh webcam secara on-line. Secara umum, hasil eksperimen menunjukkan bahwa sistem dapat mengenali pola dengan akurasi yang tinggi.

Kata kunci: Jaringan Syaraf Tiruan, Matlab, Pengenalan pola, Webcam

1. Pendahuluan

Penelitian tentang pengenalan pola dengan jaringan syaraf tiruan *backpropagation* (JST-BP) telah banyak dilakukan. Beberapa diantaranya dipadukan dengan teknik pengolahan citra (TPC) sebagai pengkondisi input JST. Lallican, cs (2000) pada papernya yang diseminarkan pada "sevent international workshop on frontiers in handwriting recognition" mendeskripsikan migrasi pengenalan pola tulisan tangan *off-line* ke *on-line*.

Publikasi hasil penelitian serupa dilakukan oleh Plamondon dan Srihari (2000), yang memberikan kajian komprehensif tentang konsep pengenalan pola *off-line* dan *on-line*. Penelitian lain dilakukan Fadlil (2004), yang mengembangkan JST-BP untuk mengenali tulisan tangan dinamis secara *on-line*. Sistem dilatih dan diuji menggunakan sebagian *database standart international IRONOFF*, tetapi sistem ini tidak menggunakan TPC untuk praproses.

Sigit, dkk (2001) dalam penelitiannya mengembangkan JST-BP dan TPC untuk pengenalan angka. Praproses yang digunakan adalah deteksi tepi *prewitt* dan ekstraksi citra ke bentuk matrik 16 x 16 piksel. Arsitektur JST-BP menggunakan 3 lapisan, yaitu 256 neuron untuk lapisan input, 18 neuron pada lapisan tersembunyi (*hidden*) dan 10 neuron untuk lapisan output. Banyak jenis pola pelatihan adalah 10 yaitu angka 1 sampai dengan angka 10. Dari pelatihan yang dilakukan, untuk mencapai *performance* yang diinginkan yaitu sebesar 0,01 dibutuhkan sebanyak 1351 epoch.

Pada penelitian ini dirancang suatu sistem pengenalan pola tertentu secara *on-line* dengan JST-BP dan TPC. Pada tahap eksperimen, sebagai pola adalah tulisan tangan alfabet kapital. Praproses (TPC) yang digunakan ialah konversi citra RGB ke citra biner, eliminasi derau, pen-center-an citra (pemfokusan pola) dan penskalaan ke matrik 20 x 20 piksel. Citra diinput dengan menggunakan suatu kamera yang telah dikondisikan. Arsitektur yang diteliti adalah 1 sampai dengan 3 *hidden layer*, yaitu untuk 1 *hidden layer* : 400 – 213 – 26, untuk 2 *hidden layer* : 400 – 275 – 151 – 26 dan untuk 3 *hidden layer* : 400 – 307 – 213 – 120 – 26. Banyak jenis pola pelatihan adalah 26 (alfabet A–Z). Untuk mengetahui yang terbaik antara 3

bentuk arsitektur JST-BP tersebut digunakan perhitungan *Mean Square Error* (MSE) dan untuk mengetahui tingkat level pengenalan tiap hurufnya digunakan perhitungan *Standard Estimate of Error* (SEE).

2. METODE PENELITIAN

Keseluruhan perangkat keras membentuk suatu sistem dengan susunan blok diagram seperti Gambar 1, dan kamera diletakkan dalam suatu tempat tertutup agar memperoleh kondisi yang tetap seperti Gambar 2.

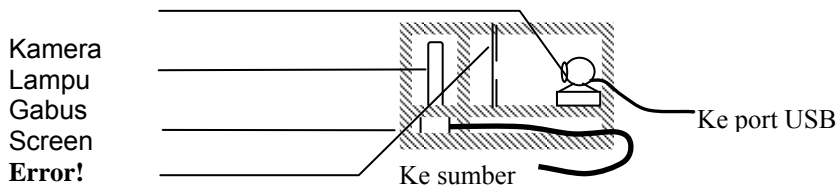
Secara keseluruhan, hardware dan software yang digunakan untuk penelitian adalah:

- Personal Computer* (PC) dengan klasifikasi : Pentium III 1,2 GHz, RAM 128 dan Hd 40 Gbyte.
- Perangkat lunak Matlab ver. 7 dan sistem operasi Windows XP Sp 1.
- Kamera *Webcam Logitech* dengan resolusi 640x840 piksel dan kabel USB.
- Pencahayaan lampu berkekuatan 65 watt, kabel dan konektor listrik.
- Kotak kamera

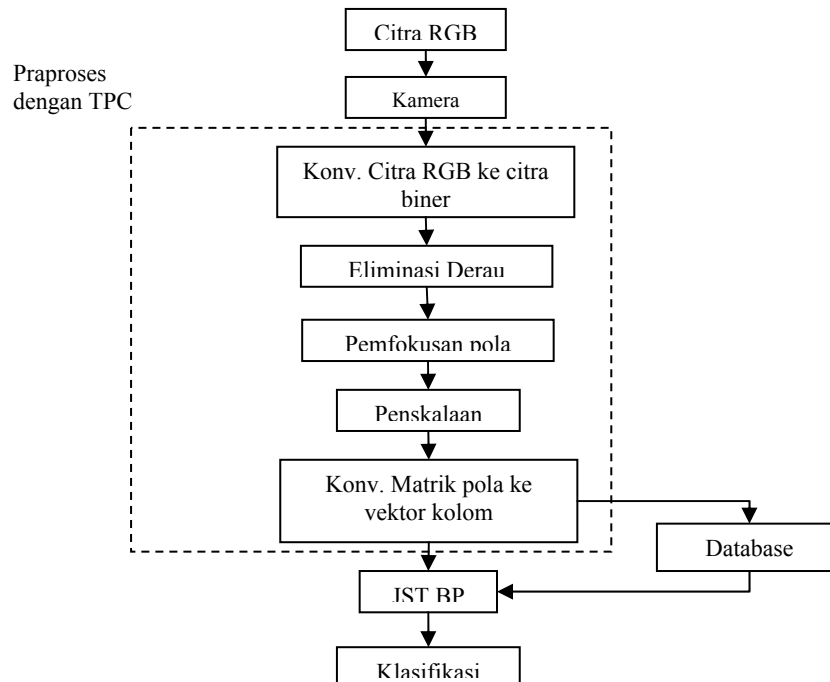


Objek Gambar

Gambar 1. Skema *Hardware* sistem pengenalan pola



Gambar 2. Skema alat pengambilan input



Gambar 3. Skema perangkat lunak sistem pengenalan pola

Perangkat lunak dibangun dengan GUI Matlab versi 7 dan terintegrasi langsung dengan kamera *webcam*. Secara diagram, skema perangkat lunak sistem ditunjukkan pada Gambar 3.

Konversi citra RGB ke citra biner melewati 2 tahapan, yaitu citra RGB ke citra keabuan (*gray level*) kemudian dibinerkan dengan threshold sebesar 100. Eliminasi derau yang digunakan adalah eliminasi 9 titik bertetangga. Pen-*center*-an (pemfokusan) adalah bagian dimana citra bagian tepi (atas, bawah, kiri dan kanan) yang memiliki nilai 0 dibuang, sehingga citra yang diperoleh benar-benar citra yang memiliki informasi. Penskalaan adalah perubahan ukuran citra menuju ukuran yang dapat ditempatkan dalam suatu *frame* yang berpixel 20x20. Hasil akhir dari praproses adalah perubahan citra yang telah diskalakan kedalam vektor kolom yang siap dipelajari oleh JST-BP.

Pada pembelajarannya JST-BP menggunakan sistem *Batch Mode* (train) dilakukan modifikasi momentum dan laju pembelajaran terhadap algoritma standard pelatihan JST-BP sebagai berikut:

Perubahan laju pembelajaran:

$$\alpha_{kj}(t+1) \begin{cases} \alpha_{kj}(t) + k & \text{Jika } \bar{\Delta}_{kj}(t-1)\Delta_{kj}(t) > 0 \\ (1-\gamma)\alpha_{kj}(t) & \text{Jika } \bar{\Delta}_{kj}(t-1)\Delta_{kj}(t) < 0 \\ \alpha_{kj}(t) & \text{Lainnya} \end{cases} \quad (1)$$

Perubahan bobot:

$$\text{dan } W_{kj}(t+1) = W_{kj}(t) + \alpha\delta_k Z_j + \mu[W_{kj}(t) - W_{kj}(t-1)] \quad (2)$$

$$V_{ji}(t+1) = V_{ji}(t) + \alpha\delta_j X_i + \mu[V_{ji}(t) - V_{ji}(t-1)] \quad (3)$$

Keterangan :

$W_{kj}(t+1), V_{ji}(t+1)$: bobot baru berikutnya

$W_{kj}(t), V_{ji}(t)$: bobot sekarang

$W_{kj}(t-1), V_{ji}(t-1)$: bobot sebelumnya

k : Learning rate increasing (kenaikan laju pembelajaran)

γ : learning rate decreasing (penurunan laju pembelajaran)

μ : Konstanta Gradien

Penentuan jumlah neuron tiap lapisan (*layer*) digunakan fungsi *linear Space* untuk membentuk jumlah neuron yang mengkrucut stabil dari lapisan input ke lapisan output. Fungsi *Linear Space* yaitu:

$$L_i = \text{floor} \left(d_1 + \left(\frac{n_i * (d_2 - d_1)}{\text{floor}(\text{layer} - 1)} \right) \right) \quad (4)$$

Keterangan:

n_i = Bilangan array dari 0 sampai dengan jumlah layer -1

d_1 = Jumlah neuron lapisan input

d_2 = Jumlah neuron lapisan output

Dari fungsi *Linear Space* diperoleh nilai untuk arsitektur JST-BP yaitu untuk 1 *hidden layer*: 400 – 213 – 26, untuk 2 *hidden layer*: 400 – 275 – 151 – 26 dan untuk 3 *hidden layer*: 400 – 307 – 213 – 120 – 26. Untuk penelitian ini variabel pelatihan JST-BP diset sebagai berikut:

Maksimal epoh	=	2000
Performan goal	=	1×10^{-5}
Minimum Gradien	=	1×10^{-20}
Konstanta Gradien	=	0,9
Learning Rate	=	0,01
Learning Rate Increasing	=	1,05
Learning Rate Decreasing	=	0,7

Pola yang diinputkan sebanyak 78 pola yang terdiri dari jenis pola A–Z, yaitu 1 sampai dengan 26, 27 sampai dengan 52 dan 53 sampai dengan 78. citra dilewatkan melalui praproses yang kemudian diubah ke dalam vektor kolom. Baris paling bawah pada vektor kolom ditambah 26 bit berisi target dari pola yang berkesesuaian, sehingga total ukuran vektor adalah 426. Pola-pola yang telah diubah ke vektor kolom disimpan ke dalam database. Setelah terkumpul sebanyak 78, pola dilatihkan ke JST-BP dengan membagi vektor sebagai input pelatihan (dari baris ke 1 sampai dengan 400 semua kolom) dan target (baris 401 sampai dengan 426 semua kolom).

Untuk analisa, data yang akan diambil adalah output simulasi ketiga JST-BP terhadap pola pelatihan, jumlah epoch dan *performance* maksimum yang diperoleh. Penentuan klasifikasi kelas terbaik terhadap ketiga arsitektur JST-BP digunakan perhitungan *Mean Square Error* (MSE) dari total output tiap arsitektur, yang dirumuskan:

$$MSE = \frac{1}{Q} \sum_{k=1}^Q (t_k - a_k)^2 \quad (5)$$

Keterangan :

Q = Jumlah pola yang dihitung

t_k = vektor target

a_k = vektor keluaran jaringan

Untuk pelevelan tingkat pengenalan tiap alphabet masing – masing arsitektur, digunakan perhitungan *Standard Estimate of Error* yang dirumuskan:

$$SEE = \sqrt{\frac{\sum (T - Y)^2}{n - 2}} \quad (6)$$

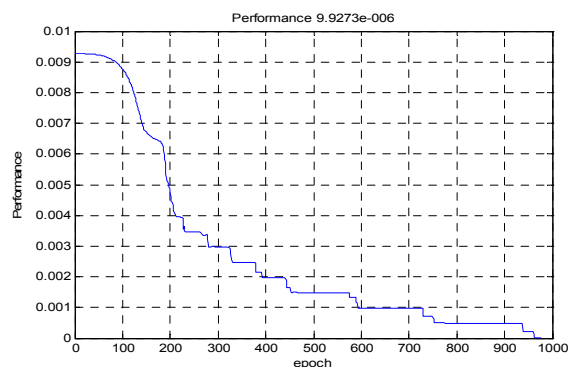
Se = *Standard Estimate of Error*

T = Nilai yang diharapkan

N = Jumlah elemen data

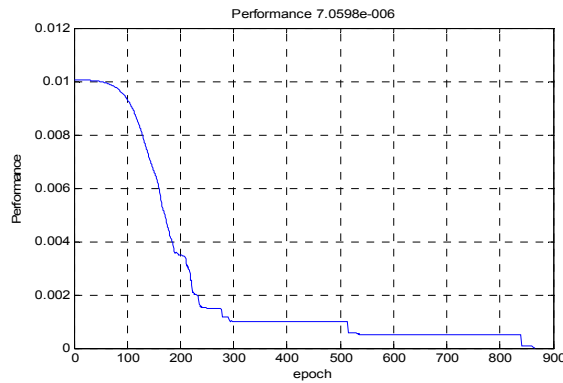
3. HASIL DAN PEMBAHASAN

Untuk 1 lapisan tersembunyi, pelatihan berhenti pada epoch ke 977, performance yang dicapai 9,9273e-006, seperti ditunjukkan pada Gambar 4.



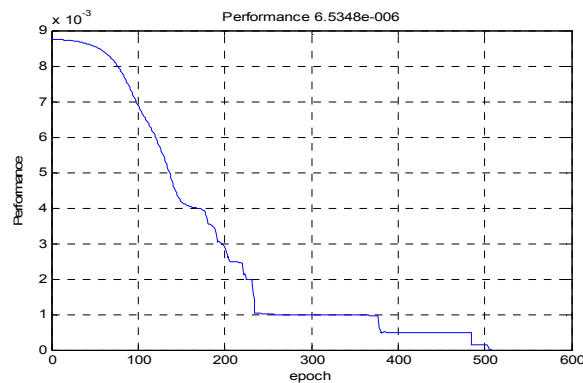
Gambar 4. Grafik pelatihan jaringan dengan 1 lapisan tersembunyi

Untuk 2 lapisan tersembunyi, pelatihan berhenti pada epoch ke 867, performance yang dicapai 7,0598e-006, seperti ditunjukkan pada Gambar 5.



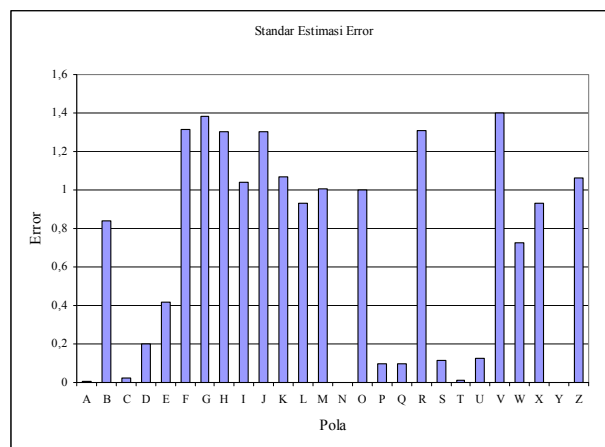
Gambar 5. Grafik pelatihan jaringan dengan 2 lapisan tersembunyi

Untuk 3 lapisan tersembunyi, pelatihan berhenti pada epoch ke 501, performance yang dicapai 6,5348e-006, seperti ditunjukkan pada Gambar 6.

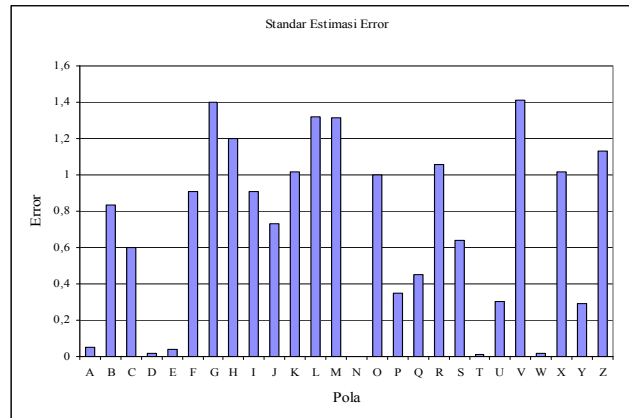


Gambar 6. Grafik pelatihan jaringan dengan 3 lapisan tersembunyi

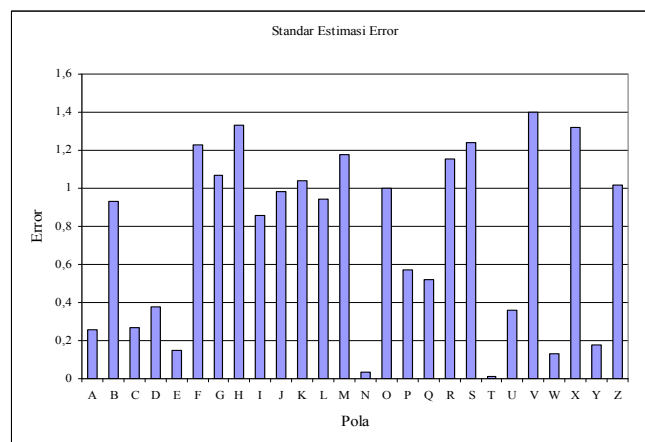
Dari ketiga arsitektur untuk mencapai *performance* yang diinginkan (1×10^{-5}) dalam mengenali 26 alphabet kapital, hanya memerlukan kurang dari 1000 iterasi. Perhitungan MSE dari 1 lapisan tersembunyi adalah $1,5594 \times 10^{-6}$, 2 lapisan tersembunyi adalah $9,94278 \times 10^{-7}$ dan 3 lapisan tersembunyi adalah $1,98841 \times 10^{-6}$. Dari perhitungan MSE ini terlihat bahwa arsitektur dengan menggunakan 2 lapisan tersembunyi mempunyai tingkat akurasi pengenalan pola lebih baik dari pada 1 dan 3 lapisan tersembunyi. Perhitungan SEE tiap jaringan diperlihatkan oleh Gambar 7, 8 dan 9.



Gambar 7. SEE 1 lapisan tersembunyi



Gambar 8. SEE 2 lapisan tersembunyi



Gambar 9. SEE 3 lapisan tersembunyi

Pelevelan tingkat pengenalan tiap alphabet untuk arsitektur JST-BP dapat dibuat sebagai berikut:

Tabel 1. Level pengenalan alphabet kapital

Level	1 Lapisan Tersembunyi	2 Lapisan Tersembunyi	3 Lapisan Tersembunyi
SBS	A, C, N, P, Q, S, T, U, Y	A, D, E, N, T, W	E, N, T, W, Y
SB	D	P, U, Y	A, C, D, E, U
B	E	C, Q	P, Q
S	W	J, S	
J	B, L, O, X	B, F, I	B, I, J, L,
SJ	I, K, M, Z	K, O, R, X, Z	G, K, O, R, Z
SJS	F, G, H, J, R, V	G, H, L, M, V	F, S, H, V, X

Keterangan:

- SBS : Sangat Baik Sekali
- SB : Sangat Baik
- B : Baik
- S : Standar
- J : Jelek
- SJ : Sangat Jelek
- SJS : Sangat Jelek Sekali

4. KESIMPULAN

Dari hasil dan pembahasan dapat ditarik kesimpulan sebagai berikut:

1. Susunan praproses (TPC) dari konversi citra RGB ke citra biner, eliminasi derau dengan 9 titik bertetangga, pemfokusan citra dan penskalaan, mampu meningkatkan kinerja sistem pengenalan pola alphabet kapital hasil tulisan tangan.

2. Ditinjau dari perhitungan MSE, arsitektur yang terbaik dihasilkan oleh 2 lapisan tersembunyi.
3. Dilihat dari nilai SEE tiap huruf untuk masing-masing arsitektur, alphabet paling sulit untuk dikenali adalah H, V, dan K, sedangkan yang paling mudah dikenali adalah N, T, dan A.
4. Keberhasilan perangkat lunak sangat tergantung kondisi piranti pendukung seperti kamera video dan intensitas cahaya.

DAFTAR PUSTAKA

- [1] Fadlil, 2004, "**Pengaruh Kompleksitas Ekstraksi Ciri Pada Sistem Pengenalan Tulisan Tangan Dinamis**" Jurnal Ilmiah Telkomnika, Vol.2, No.1.
- [2] Fausett, L., 1994, "**Fundamentals Of Neural Networks : Architecture, algorithms, And Applications**" Practise Hall, New Jersey .
- [3] Jain, A. K., 1995, "**Fundamentals of Digital Image Processing**", Practice Hall, New Jersey.
- [4] Mudjirahardjo, P., 2001, "**Penerapan JST-BP untuk Pengenalan Kode Pos Tulisan Tangan**" Tesis S-2, Universitas Gadjah Mada, Yogyakarta.
- [5] Plamondon, R. and Srihari, S. N., 2000, "**On-line and Off-line Handwriting Recognition : A Comprehensive Survey**" IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No.1.
- [6] Raph, G., 1997, "**Run-On Recognition in an On-line Handwriting Recognition System**" Report University of Karlsruhe, Germany.
- [7] Sigit, R., 2001, "**Aplikasi Jaringan Syaraf Tiruan Untuk Mengenali Angka**" Institut Teknologi Surabaya, Surabaya.
- [8] Tappert, C.C., Suen, C. Y. and Wakahara, T., 1990, "**The State of the Art in On-Line Handwriting Recognition**" IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 12, No.8.