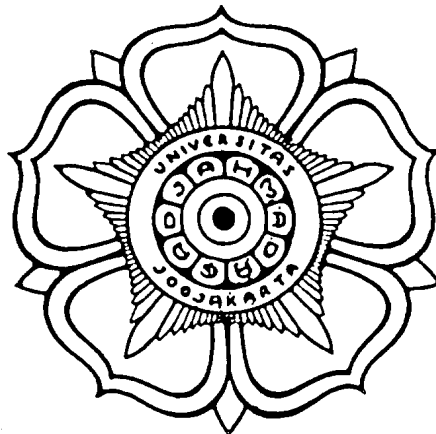


PENERAPAN JARINGAN PERAMBATAN-BALIK UNTUK PENGENALAN KODE POS TULISAN TANGAN

Tesis
untuk memenuhi sebagian persyaratan
mencapai derajat Sarjana S-2

Program Studi Teknik Elektro
Jurusan Ilmu-ilmu Teknik



diajukan oleh:
Panca Mudjirahardjo
12791/I-1/1191/99

kepada
PROGRAM PASCASARJANA
UNIVERSITAS GADJAH MADA
YOGYAKARTA

2001

Naskah Tesis

**PENERAPAN JARINGAN PERAMBATAN-BALIK
UNTUK PENGENALAN KODE POS TULISAN TANGAN**

yang diajukan oleh
Panca Mudjirahardjo
12791/I-1/1191/99

telah disetujui oleh:

Pembimbing Utama

Dr. Ir. Thomas Sri Widodo, DEA.

Tanggal

Pembimbing Pendamping

Prof. Adhi Susanto, M.Sc., Ph.D.

Tanggal

PERNYATAAN

Dengan ini saya menyatakan bahwa dalam tesis ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 22 Nopember 2001

Panca Mudjirahardjo

KATA PENGANTAR

Alhamdulillah segala puji bagi Allah. Akhirnya selesai sudah tesis ini. Tesis ini adalah laporan hasil penelitian di bidang pengolahan citra dan jaringan syaraf tiruan pada aplikasi pengenalan kode pos tulisan tangan, yang merupakan syarat untuk mencapai derajat sarjana S2 di program studi teknik elektro, Universitas Gadjah Mada, Yogyakarta.

Seperti halnya karya ilmiah lainnya, tesis ini jauh dari sempurna dan perlu adanya perbaikan dan tambahan di sana-sini seperti tertulis dalam saran-saran di belakang, untuk dapat diterapkan dalam dunia nyata.

Dalam penyelesaian tesis ini, penulis tidak lupa mengucapkan banyak terima kasih kepada:

1. Koordinator proyek DUE, Ditjen DIKTI-DEPDIKNAS, dan Rektor Universitas Brawijaya Malang, yang telah memberikan beasiswa dan kesempatan kepada penulis untuk belajar di program Pasca Sarjana, Universitas Gadjah Mada.
2. Dr. Ir. Thomas Sri Widodo, DEA., selaku pembimbing pertama, yang telah memberikan bimbingan, pengarahan, masukan dan koreksinya terhadap tesis ini, dan selaku Ka. Lab. Elektronika Lanjut yang telah memberi ijin dan keleluasaan penulis untuk menggunakan pemayar, selama penyelesaian tesis ini.
3. Prof. Adhi Susanto, M.Sc., Ph.D., selaku pembimbing pendamping, yang telah memberikan koreksi, saran-saran dan nasehatnya terhadap tesis ini, dan masa depan penulis,

4. Istriku Peni atas dukungan moral dan kesabarannya menemani dalam suka duka selama dua tahun di Yogya, Mas Nonot atas bantuan dan bimbingan spiritualnya, Alm. ibuku Sri Aminah dan Alm. Bapak Soewarno atas do'a restu, serta dorongan morilnya, Widyadi Setiawan (UNUD) atas pinjaman paper *download*-nya, serta Pak Aris selaku laboran Lab. Elektronika Lanjut yang memberikan keleluasaan dalam menggunakan pemayar.
5. Ibu Soemilatin, Ka. SDN Banyudono II Ponorogo. Mbak Dewi Iriani, BA, guru SMPN 5, Jombang. Bpk. Sumarno, BA, Ka. SDN Jongkang, Sleman. Rekan-rekan serta semua pihak yang telah membantu menyediakan data-data tulisan tangan,
6. Serta seluruh rekan angkatan 1999 minat SIE, atas semangat belajar bersama, diskusi serta guyonannya dalam suka duka menempuh studi di UGM.

Akhir kata ibarat pepatah tak ada gading yang tak retak, penulis selalu mengharap kritik dan saran demi perbaikan tulisan ini. Semoga tesis ini berguna bagi penulis dan para pembaca.

Yogyakarta, 22 Nopember 2001

Kupersembahkan buat
Almarhumah ibuku Sri Aminah
dan istriku Peni Hapsari

فان مع العسر يسرا

(Qs. Al Insyirah: 5)
" Sesungguhnya setelah kesulitan itu ada kemudahan"

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
HALAMAN PERNYATAAN	iii
KATA PENGANTAR	iv
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR GRAFIK	x
DAFTAR TABEL	xi
INTISARI	xii
ABSTRACT	xiii
I. PENGANTAR	1
I.1 Latar Belakang	1
I.1.1 Perumusan Masalah	2
I.1.2 Batasan Masalah	3
I.1.3 Keaslian Penelitian	3
I.1.4 Manfaat	3
I.2 Tujuan Penelitian	4
I.3 Sistematikan Penulisan	4
II. TINJAUAN PUSTAKA	6
II.1 Tinjauan Pustaka	6
II.2 Landasan Teori	7
II.2.1 Jaringan Neural	7
II.2.2 Pemodelan Sistem	18
II.2.3 Pengenalan Kode Pos	20
II.2.3.1 Pengolahan Awal dan Segmentasi	20
II.2.3.2 Pengenalan Angka	24
II.3 Hipotesis	35
II.4 Rencana Penelitian	36
III. CARA PENELITIAN	37
III.1 Bahan Penelitian	37

III.2 Alat yang Digunakan	37
III.3 Jalannya Penelitian	38
III.3.1 Basis Data	38
III.3.2 Pemrosesan Basis Data	42
III.3.3 Perancangan Arsitektur Jaringan Utama	42
III.3.4 Perancangan Sistem Deteksi Kesalahan	43
III.4 Kesulitan-kesulitan dan Cara Pemecahannya	43
IV. HASIL PENELITIAN DAN PEMBAHASAN	45
IV.1 Pengolahan Awal dan Segmentasi	45
IV.2 Pengenalan Angka	47
IV.3 Pengenalan Kode Pos	57
V. PENUTUP	58
V.1 Kesimpulan	58
V.2 Saran	59
VI. RINGKASAN	60
DAFTAR PUSTAKA	
LAMPIRAN	

DAFTAR GAMBAR

Gambar 2.1 Jaringan perambatan-balik dengan satu lapisan tersembunyi	11
Gambar 2.2 Fungsi aktivasi identitas	12
Gambar 2.3 Fungsi aktivasi <i>sigmoid</i> biner	13
Gambar 2.4 Fungsi aktivasi <i>sigmoid</i> bipolar	13
Gambar 2.5 Sistem interpretasi alamat tulisan tangan	19
Gambar 2.6 Diagram blok pengenalan kode pos	20
Gambar 2.7 Proses pencarian baris akhir	21
Gambar 2.8 Analisa jarak kata dengan histogram latar depan	22
Gambar 2.9 Penghapusan garis kotak	24
Gambar 2.10 Diagram blok pengenalan angka	25
Gambar 2.11 Normalisasi citra karakter	25
Gambar 2.12 Citra biner angka '2'	28
Gambar 2.13 Pembuatan ciri batang latar depan arah timur	28
Gambar 2.14 Pembuatan ciri batang latar depan arah timur laut	28
Gambar 2.15 Pembuatan ciri batang latar depan arah utara	29
Gambar 2.16 Pembuatan ciri batang latar depan arah barat laut	29
Gambar 2.17 Contoh perhitungan vektor ciri batang	30
Gambar 2.18 Komputasi ciri transisi suatu baris	32
Gambar 2.19 Komputasi ciri transisi pada citra karakter	33
Gambar 2.20 Penguraian vektor ciri transisi dari kiri ke kanan, atas bawah	34
Gambar 2.21 Diagram blok sistem deteksi kesalahan	35
Gambar 3.1 Contoh formulir isian untuk basis data angka	40
Gambar 3.2 Contoh alamat tulisan tangan	41
Gambar 4.1 Hasil pengujian pencarian baris akhir	46
Gambar 4.2 Hasil dilasi citra	47
Gambar 4.3 Klasifikasi pengenalan angka	57

DAFTAR GRAFIK

Grafik 4.1 Pengujian memorisasi berdasarkan jumlah neuron tersembunyi	50
Grafik 4.2 Pengujian generalisasi berdasarkan jumlah neuron tersembunyi	50
Grafik 4.3 Hasil pengenalan memorisasi berdasarkan fungsi aktivasi	51
Grafik 4.4 Hasil pengenalan generalisasi berdasarkan fungsi aktivasi	51
Grafik 4.5 Pengenalan total berdasarkan fungsi aktivasi	52
Grafik 4.6 Kurva pembelajaran dengan ciri batang	53
Grafik 4.7 Kurva pembelajaran dengan ciri transisi	54
Grafik 4.8 Perbandingan hasil pengenalan angka berdasarkan ciri	54
Grafik 4.9 Perbandingan waktu eksekusi berdasarkan ciri yang digunakan	55
Grafik 4.10 Perbandingan hasil keluaran jaringan acuan berdasarkan ciri	56

DAFTAR TABEL

Tabel 4.1 Rata-rata pengenalan angka berdasarkan fungsi aktivasi	52
Tabel 4.2 Hasil pengenalan kode pos dan angka	57

Intisari

Pembacaan karakter dengan komputer, dikenal sebagai *Optical Character Recognition* (OCR), adalah topik yang telah diteliti selama beberapa tahun. Sistem OCR telah dikomersialkan secara luas untuk pemrosesan dokumen-dokumen hasil cetakan. Namun teknik-teknik untuk membaca tulisan tangan belum sukses diaplikasikan. Permasalahan yang dihadapi untuk pengenalan tulisan tangan sangat kompleks, seperti bervariasinya model tulisan tangan, pena untuk menulis, dan lain lain.

Interpretasi alamat tulisan tangan oleh sistem komputer merupakan hal penting untuk pemrosesan surat secara otomatis. Penelitian ini menerapkan jaringan neural untuk pengenalan angka pada kode pos alamat tulisan tangan sebagai penentu kota tujuan surat. Dengan pengenalan kota tujuan pada alamat tulisan tangan oleh sistem komputer diharapkan surat-surat di Indonesia dapat ditangani secara otomatis.

Yang membedakan penelitian ini dengan penelitian sebelumnya adalah pada penelitian ini masukan berupa seluruh citra alamat tulisan tangan, dan penggunaan jaringan neural berkaskade untuk sistem deteksi kesalahan. Hasil pengenalan angka dengan gabungan vektor ciri batang dan transisi, pengujian memorisasinya mencapai 98,2% dan pengujian generalisasinya mencapai 96,2%. Pengenalan kode pos mencapai 79,56%.

Kata kunci:

pengenalan angka tulisan tangan, jaringan perambatan-balik, ciri batang, ciri transisi, deteksi kesalahan.

Abstract

A character reading by computer, known as *Optical Character Recognition* (OCR), is a topic that has been studied and researched for several years. OCR system has commercial values and widely used for processing machine printed documents. Techniques for handwriting recognition, however, have not successfully applied yet. The problems of handwriting recognition are very complex: due to a variety of handwritten models, sizes, pens used, and so on.

Computer handwritten address interpretation is important for automated letter processing. This research applied back-propagation algorithm for handwritten postcode digit recognition.

The differences between this research and the others are the use of the whole image of handwritten address as system input, and the cascaded neural networks for fault detection system. The recognition results, using bar and transition feature vectors, are the memorization of 98,2%, the generalization of 96,2%, and the postcode recognition success is 79,56%.

Keyword:

Handwritten digit recognition, postcode recognition, back-propagation networks, bar feature, transition feature, and fault detection.

I. Pengantar

I.1 Latar Belakang

Pembacaan karakter dengan komputer dikenal sebagai *Optical Character Recognition* (OCR) adalah topik yang telah diteliti selama beberapa tahun. Sebagian besar solusi untuk permasalahan ini mendasarkan pada proses segmentasi masukan menjadi karakter-karakter. Hasil segmentasi ini yang secara detail akan dikenali. Sistem OCR telah dikomersialkan secara luas untuk pemrosesan dokumen-dokumen hasil cetakan. Namun teknik-teknik untuk membaca tulisan tangan belum sukses diaplikasikan. Permasalahan yang dihadapi untuk pengenalan tulisan tangan sangat kompleks, seperti bervariasinya model tulisan tangan, pena untuk menulis, dan lain lain.

Interpretasi alamat tulisan tangan oleh sistem komputer merupakan hal penting untuk pemrosesan surat secara otomatis. Hampir 80% alamat surat di Indonesia ditulis dengan tulisan tangan. Pengembangan sebuah sistem interpretasi alamat efektif, merupakan tantangan karena besarnya variabel dan ketidakpastian pada alamat tulisan tangan.

Interpretasi alamat tulisan tangan oleh sistem komputer telah dilakukan dan terus dikembangkan di kantor pos Amerika Serikat. Pengenalan kata tulisan tangan dikarakterisasi oleh kecepatan data yang tinggi, jumlah data yang besar, mengurangi kemungkinan masukan yang salah dan kebutuhan untuk tanggapan waktu nyata (*real time response*).

Penelitian ini bermaksud menerapkan jaringan neural untuk pengenalan angka pada kode pos alamat tulisan tangan sebagai penentu

kota tujuan surat. Dengan pengenalan kota tujuan pada alamat tulisan tangan oleh sistem komputer diharapkan bisa mengalokasikan surat-surat di Indonesia secara otomatis, khususnya, dan untuk dikembangkan lagi dalam pengenalan kata tulisan tangan lainnya.

I.1.1 Perumusan Masalah

Pemrosesan dan pengenalan karakter tulisan tangan sangat menantang, disebabkan beragamnya jenis tulisan tangan, peralatan menulis, dan lain-lain. Tiga komponen utama yang berpengaruh pada kompleksitas permasalahan pengenalan tulisan tangan adalah banyaknya karakter, tipe tulisan dan jumlah penulis. Semakin banyak batasan pada ketiga komponen ini, semakin mudah untuk menghasilkan suatu solusi. Kata mudah di sini bukan berarti permasalahan ini suatu yang langsung dapat diselesaikan, namun diartikan menghasilkan pesat pengenalan yang lebih tinggi.

Metode pengenalan karakter tulisan tangan yang telah dilakukan dan terus dikembangkan terdiri atas jaringan neural, logika kabur, dan logika tegas. Perumusan masalah dalam penelitian ini adalah: (1) bagaimana membuat perangkat-lunak untuk mengenali kode pos dalam alamat tulisan tangan, (2) bagaimana proses segmentasi kode pos menjadi karakter angka-angka, serta (3) pemilihan parameter-parameter jaringan perambatan balik untuk pengenalan angka dan deteksi kesalahan.

I.1.2 Batasan Masalah

Batasan masalah pada penelitian ini adalah pengenalan angka kode pos tulisan tangan pada amplop surat berwarna putih polos, penulisan kode pos baku, yaitu kode pos terdapat pada baris terakhir dan blok terakhir serta penulisan angka terpisah satu sama lain.

I.1.3 Keaslian Penelitian

Pengenalan nomor jalan pada alamat tulisan tangan berdasarkan logika kabur dan jaringan neural telah dikembangkan Paul Gader dkk. 1997; 2000, untuk mengalokasikan surat-surat di Amerika Serikat. Pada penelitian yang telah dilakukan mereka menggunakan dua vektor ciri, yaitu batang dan transisi untuk masukan jaringan neural-nya. Metode yang dipakai segmentasi dan tanpa segmentasi. Metode pengenalan karakter hanya pada blok citra tulisan tangan yang akan dikenali. Serta tidak adanya sistem deteksi kesalahan.

Pada penelitian ini citra masukan berupa seluruh blok alamat tulisan tangan, dengan kode pos didalam dan diluar kotak, yang berbeda dengan peneliti sebelumnya. Serta ditambah sistem deteksi kesalahan. Basis data yang dibuat berdasarkan angka tulisan tangan Indonesia.

I.1.4 Manfaat

Dari segi keilmuan, penelitian ini menghasilkan manfaat pada penyediaan basis data bagi peneliti yang berminat pada pengenalan angka tulisan tangan Indonesia untuk aplikasi lainnya, juga menghasilkan suatu

metode yang dapat dipakai untuk pengenalan tulisan tangan maupun untuk perbandingan unjuk kerja dengan metode lainnya. Dari segi sosial memberikan sumbangan ide dan perangkat-lunak yang dapat dikembangkan untuk otomasi pemrosesan surat.

I.2 Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah untuk membuat sistem pengenalan angka kode pos pada blok citra alamat tulisan tangan dengan jaringan neural dan menggunakan jaringan neural kaskade sebagai sistem deteksi kesalahan.

I.3 Sistematika Penulisan

Langkah awal penelitian tentang pemrosesan alamat tulisan tangan ini adalah mengumpulkan literatur dari berbagai sumber, baik perpustakaan *off-line* maupun *on-line*. Sistematika penulisan tesis adalah sebagai berikut:

Bab I Pengantar, memuat latar belakang, dan tujuan penelitian

Bab II Tinjauan Pustaka, memuat uraian sistematis tentang hasil-hasil penelitian yang didapatkan peneliti terdahulu, juga landasan teori yang merupakan tuntunan untuk memecahkan masalah, dan perumusan hipotesis.

Bab III Cara Penelitian, berisi uraian rinci tentang materi penelitian, alat, jalannya penelitian, analisis hasil, dan kesulitan-kesulitan serta cara-cara pemecahannya.

Bab IV Hasil Penelitian dan Pembahasan, berisi hasil penelitian dan pembahasan yang sifatnya terpadu.

Bab V Kesimpulan, memuat kesimpulan atas penelitian yang dilakukan dan saran-saran untuk pengembangan pemecahan lebih lanjut masalah pada penelitian ini.

Bab VI Ringkasan, memuat dengan lengkap dan singkat latar belakang, tinjauan pustaka, landasan teori, cara penelitian, dan hasil.

II. Tinjauan Pustaka

Bagian ini memuat tinjauan pustaka, landasan teori, hipotesis, dan rencana penelitian.

II.1 Tinjauan Pustaka

Pendeteksian dan pengenalan nomor jalan dalam suatu alamat tulisan tangan berdasar sistem logika kabur, telah dikembangkan oleh Paul Gader dkk, 1995. Paul Gader menerapkannya di Kantor Pos Amerika Serikat, karena 15 % alamat ditulis tangan. Pada penelitian ini menerapkan pengolahan citra yaitu segmentasi citra tulisan tangan ke karakter-karakter, pengambilan ciri dan modul pengenalan. Dua tipe vektor ciri dipakai sebagai masukan jaringan neural, vektor ciri transisi dan vektor ciri batang. Jaringan neural dilatih menggunakan perambatan balik dan menggunakan keluaran terkode kelas. Terdapat kelas bernama 'sampah' untuk menerangkan segmen yang tak menyatakan citra karakter, seperti multiple karakter atau potongan karakter. Jaringan umpan maju banyak lapis dapat melakukan klasifikasi pola dengan sangat baik. Sebagai contoh jaringan ini merupakan alat yang baik untuk pengenalan karakter. Namun, kesulitan menggunakan jaringan ini, yaitu adanya beberapa parameter, seperti jumlah unit tersembunyi, fungsi aktivasi dan kecepatan pembelajaran yang harus dipilih.

Paul Gader, dkk, 1995, juga telah melakukan perbandingan pelatihan jaringan neural menggunakan himpunan keanggotaan tegas dan kabur. Dari perbandingan ini diperoleh jaringan neural yang dilatih dengan himpunan keanggotaan tegas hasilnya baik untuk pengenalan karakter, sedangkan

jaringan neural yang dilatih dengan himpunan keanggotaan kabur baik untuk pengenalan kata.

Pengembangan sistem interpretasi alamat efektif adalah tugas yang sangat menantang, karena banyaknya variabel dan ketaktentuan dalam alamat tulisan tangan. Bagian numerik pada suatu alamat, misalnya nomor rumah dan kode pos, memainkan peranan penting dalam menurunkan kekompleksitasan tugas interpretasi alamat. Jika numerik ini dideteksi dan diidentifikasi dengan benar, maka kemungkinan jumlah alamat dapat dikurangi.

Dalam penelitian yang dilakukan Paul Gader dkk. tersebut, untuk mengenali nomor jalan dalam alamat tulisan tangan, digunakan dua vektor ciri transisi dan batang citra karakter sebagai masukan jaringan neuralnya, dan menggunakan logika kabur untuk mengenali daerah numeris suatu alamat tulisan tangan.

II.2 Landasan Teori

II.2.1 Jaringan Neural

Teori, arsitektur, dan algoritma berbagai model jaringan neural telah banyak dibahas dalam literatur. Pada bagian ini hanya dijelaskan gambaran umum penggunaan jaringan neural pada aplikasi pemrosesan kode pos alamat tulisan tangan.

Banyak permasalahan menarik yang termasuk kategori pengenalan pola. Salah satu yang banyak didapatkan pada aplikasi jaringan neural adalah pengenalan karakter/kata tulisan tangan (huruf/angka). Besar variasi

huruf/kata, posisi, dan jenis tulisan membuat permasalahan ini menjadi sulit bagi teknik-teknik tradisional. Pemrosesan kode pos alamat tulisan tangan termasuk di dalamnya, dengan penggunaan teknik tradisional memang telah terbukti dapat menghasilkan pesat pengenalan yang relatif tinggi, namun di sisi lain, yaitu waktu, teknik ini membutuhkan komputasi yang panjang, sehingga waktu eksekusi untuk mencapai tujuan yang diinginkan relatif panjang. Penggunaan jaringan neural disini diharapkan mengurangi kekurangan teknik tradisional dan waktu eksekusi akan tidak menjadi masalah dengan komputer yang cenderung makin cepat dua kali dalam 18 bulan.

Jaringan neural lapis-jamak (*multilayer*) dipakai untuk aplikasi umum, jadi dipilih metode ini untuk aplikasi pengenalan karakter tulisan tangan. Jaringan neural *backpropagation* termasuk jaringan lapis-jamak yang dilatih memakai algoritma *backpropagation*, dan umum dipakai untuk pengenalan tulisan tangan. Walau pun aplikasi jaringan ini berdasar pada algoritma pelatihan standar, namun jaringan ini memiliki kebebasan dalam menentukan arsitektur jaringan yang akan dipakai agar dapat meningkatkan unjuk-kerjanya.

Perambatan-Balik (*Back-Propagation*)

Metode pelatihan perambatan-balik secara sederhana adalah metode *gradient descent* untuk meminimalkan total galat kuadrat keluaran. Aplikasi yang memakai jaringan ini untuk masalah yang melibatkan pemetaan

sekumpulan masukan terhadap sekumpulan target keluaran; jadi masuk kategori jaringan dengan pelatihan terbimbing.

Tujuan pelatihan jaringan ini adalah mendapatkan keseimbangan antara kemampuan tanggapan yang benar terhadap pola masukan yang dipakai untuk pelatihan jaringan (*memorization*) dan kemampuan memberikan tanggapan yang layak untuk masukan yang sejenis namun tidak identis dengan yang dipakai pada pelatihan (*generalization*).

Pelatihan jaringan perambatan balik melibatkan tiga tahap. Umpan-maju pelatihan pola masukan, komputasi dan perambatan-balik galat, serta perubahan bobot. Setelah pelatihan, aplikasi jaringan hanya melibatkan tahap komputasi umpan-maju. Walau pun pelatihan jaringan lambat, jaringan yang telah dilatih dapat menghasilkan keluaran dengan sangat cepat. Beragam variasi perambatan-balik telah banyak dikembangkan untuk meningkatkan kecepatan proses pelatihan.

Aplikasi dengan jaringan lapisan tunggal dapat dilakukan dengan pemetaan yang terbatas, sehingga jaringan lapis-jamak dengan satu atau lebih lapisan tersembunyi dapat diaplikasikan untuk pemetaan kontinyu dengan keakuratan tertentu.

Arsitektur

Jaringan neural lapis-jamak dengan satu lapisan tersembunyi (unit Z) diperlihatkan pada Gambar 2.1. Unit keluaran (unit Y) dan unit-unit tersembunyi serta bias diperlihatkan. Bias pada unit keluaran Y_k dilambangkan dengan w_{0k} , bias pada unit tersembunyi Z_j dilambangkan

dengan v_{0j} . Istilah bias-bias ini berlaku sebagai bobot pada hubungan unit-unit dengan keluaran selalu satu. Hanya aliran informasi umpan-maju yang diperlihatkan pada Gambar 2.1. Selama fase pelatihan perambatan-balik, sinyal dikirim pada arah yang berlawanan.

Algoritma untuk lapisan tersembunyi tunggal, yang akan diaplikasikan pada pengenalan kata untuk penelitian ini disajikan di bawah ini.

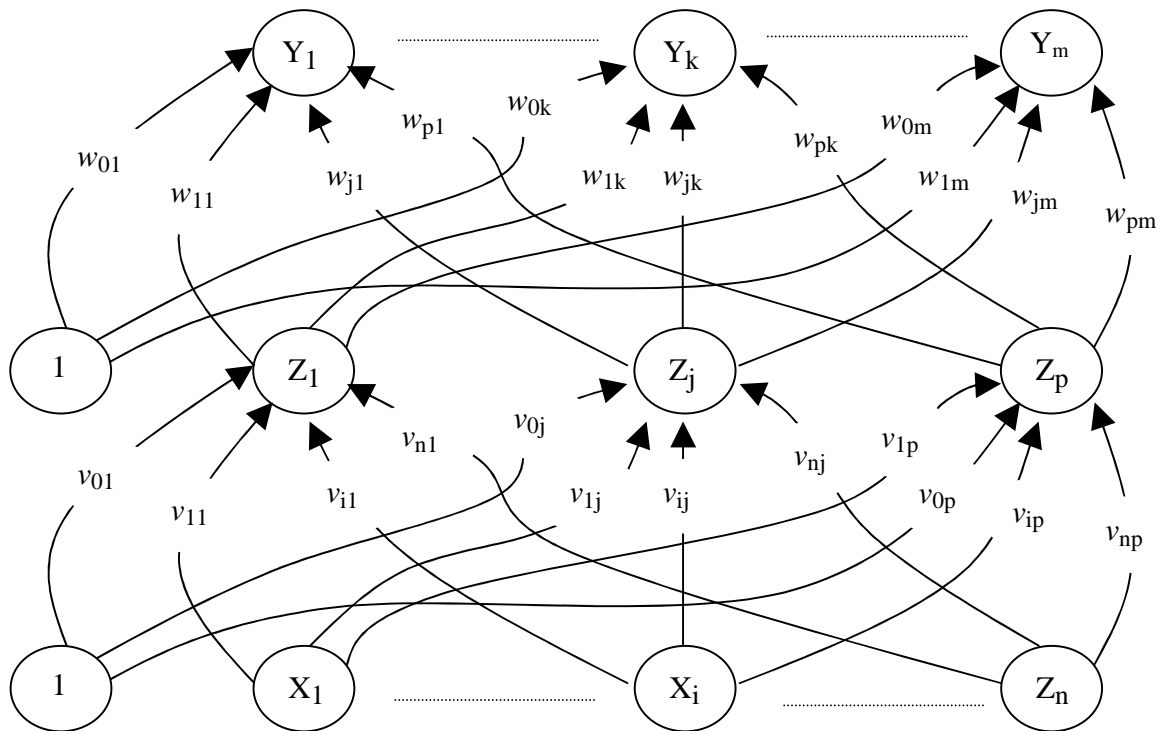
Algoritma

Seperti telah disebutkan sebelumnya, pelatihan jaringan perambatan balik melibatkan tiga tahap: umpan-maju pola-pola masukan untuk pelatihan, perambatan-balik galat, dan perubahan bobot-bobot.

Selama umpan-maju, tiap unit masukan (X_i) menerima sinyal masukan dan mengirimkannya ke tiap unit-unit tersembunyi Z_1, \dots, Z_p . Tiap unit tersembunyi menghitung aktivasinya dan mengirim sinyal tersebut (z_j) ke tiap unit keluaran. Tiap unit keluaran (Y_k) menghitung aktivasinya (y_k) untuk membuat tanggapan jaringan untuk pola masukan yang diberikan.

Pada pelatihan, tiap unit keluaran diperbandingkan nilai aktivasi y_k dengan nilai target t_k untuk menentukan galat. Berdasar pada galat ini, faktor δ_k ($k = 1, \dots, m$) dihitung. Kemudian δ_k dipakai untuk mendistribusikan galat pada unit keluaran Y_k ke semua unit pada lapisan tersembunyi yang berhubungan dengan Y_k . Juga akan dipakai untuk merubah bobot antara unit keluaran dan unit lapisan tersembunyi. Dengan cara yang sama, faktor δ_j ($j = 1, \dots, p$) dihitung untuk tiap unit tersembunyi Z_j . Namun tidak perlu

mengembalikan galat ke lapisan masukan, δ_j dipakai untuk mengubah bobot antara lapisan tersembunyi dan lapisan masukan.



Gambar 2.1 Jaringan neural perambatan-balik dengan satu lapisan tersembunyi

Setelah semua faktor δ didapatkan, bobot-bobot pada semua lapisan diubah secara simultan. Perubahan bobot w_{jk} (dari unit tersembunyi Z_j ke unit keluaran Y_k) berdasar faktor δ_k dan aktivasi z_j pada unit tersembunyi Z_j . Perubahan bobot v_{ij} (dari unit masukan X_i ke unit tersembunyi Z_j) berdasar faktor δ_j dan aktivasi x_i pada unit masukan.

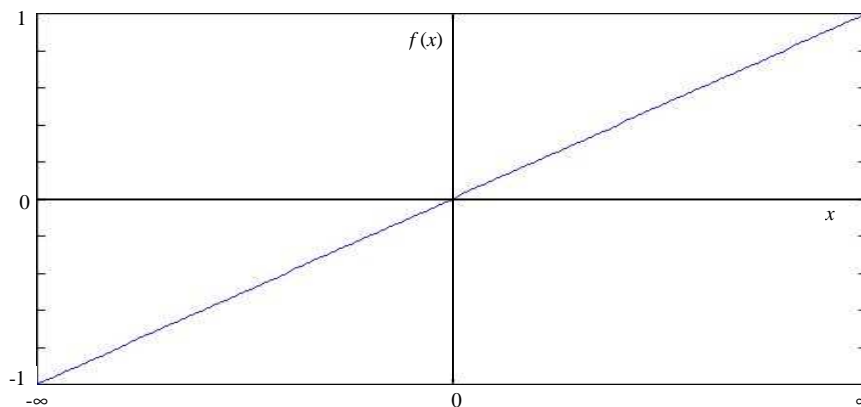
Fungsi Aktivasi

Karakteristik yang harus dimiliki oleh fungsi aktivasi jaringan perambatan-balik di antaranya adalah: harus kontinu, *differentiable*, dan

monotonically non-decreasing. Untuk efisiensi komputasi, turunan fungsi tersebut mudah didapatkan dan nilai turunannya dapat dinyatakan dengan fungsi aktivasi itu sendiri. Beberapa fungsi aktivasi dijelaskan di bagian berikut.

Fungsi aktivasi pertama adalah fungsi identitas atau linear ditunjukkan pada Gambar 2.2, didefinisikan sebagai:

$$f(x) = x, \quad \text{untuk semua } x \quad (1)$$



Gambar 2.2 Fungsi aktivasi identitas

Fungsi aktivasi kedua adalah *sigmoid* biner atau *sigmoid logistic*, ditunjukkan pada Gambar 2.3, didefinisikan sebagai:

$$f(x) = \frac{1}{1 + \exp(-\sigma x)} \quad (2)$$

dengan

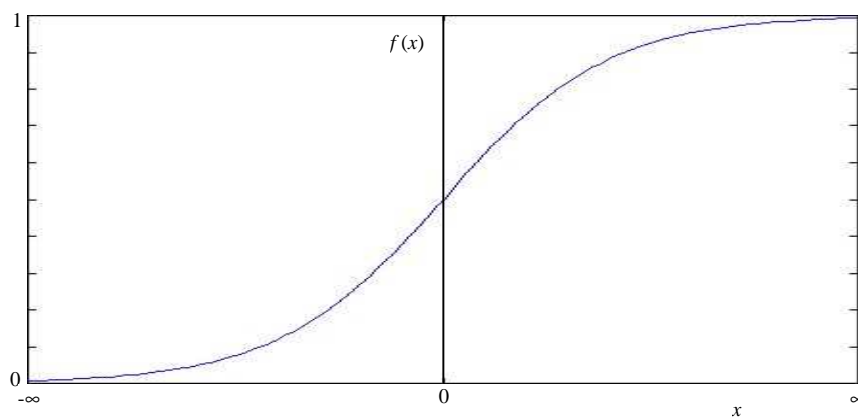
$$f'(x) = \sigma f(x)[1 - f(x)] \quad (3)$$

Fungsi aktivasi ketiga adalah *sigmoid* bipolar (Gambar 2.4), yang memiliki jangkauan nilai $(-1,1)$ dan didefinisikan sebagai:

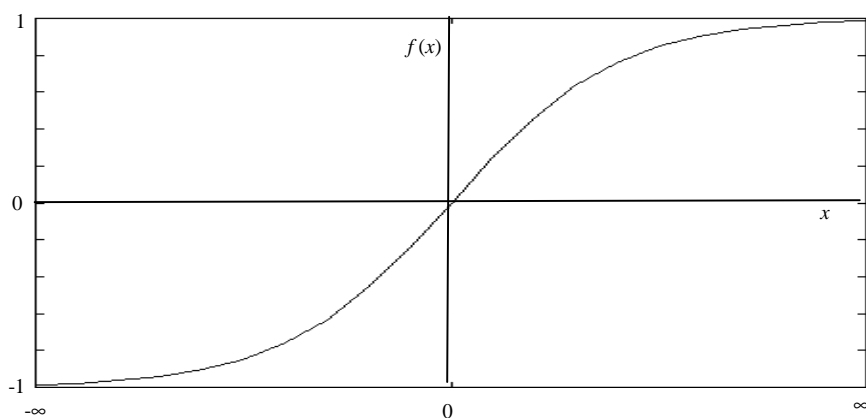
$$f(x) = \frac{2}{1 + \exp(-\sigma x)} - 1 \quad (4)$$

dengan

$$f'(x) = \frac{\sigma}{2} [1 + f(x)][1 - f(x)] \quad (5)$$



Gambar 2.3 Fungsi aktivasi *sigmoid* biner



Gambar 2.4 Fungsi aktivasi *sigmoid* bipolar

Algoritma Pelatihan

Langkah-langkah pelatihan jaringan perambatan balik sebagai berikut:

0. Inisialisasi bobot.

(bobot diberi nilai acak yang kecil)

1. Jika kondisi berhenti salah, kerjakan 2-9.

2. Untuk tiap pasangan pola masukan, kerjakan 3-8.

Umpan-maju:

3. Tiap unit masukan (X_i , $i = 1, \dots, n$) menerima sinyal masukan x_i dan meneruskan sinyal ini ke semua unit pada lapisan di atasnya (unit tersembunyi).

4. Tiap unit tersembunyi (Z_j , $j = 1, \dots, p$) hitung sinyal masukan berbobotnya,

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

Terapkan fungsi aktivasinya untuk menghitung sinyal keluaran,

$$z_j = f(z_in_j)$$

Dan kirim sinyal ini ke semua unit pada layer di atasnya (unit keluaran).

5. Tiap unit keluaran (Y_k , $k = 1, \dots, m$) hitung sinyal masukan berbobotnya,

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

dan terapkan fungsi aktivasinya untuk menghitung sinyal keluaran,

$$y_k = f(y_in_k)$$

Perambatan balik galat:

6. Tiap unit keluaran (Y_k , $k = 1, \dots, m$) menerima pola target yang bersesuaian dengan pola masukan, dihitung parameter informasi galat,

$$\delta_k = (t_k - y_k) f'(y_{in_k})$$

Hitung parameter koreksi bobotnya (dipakai untuk mengubah w_{jk}),

$$\Delta w_{jk} = \alpha \delta_k z_j, \quad \alpha = \text{pesat pembelajaran}$$

Hitung parameter koreksi bias (dipakai untuk mengubah w_{0k})

$$\Delta w_{0k} = \alpha \delta_k$$

dan kirim δ_k ke unit-unit dibawahnya.

7. Tiap unit tersembunyi (Z_j , $j = 1, \dots, p$) hitung masukan deltanya (dari unit pada lapisan dibawahnya),

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

Kalikan dengan turunan fungsi aktivasinya untuk menghitung parameter informasi galatnya,

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

Hitung parameter koreksi bobotnya (dipakai untuk mengubah v_{ij}),

$$\Delta v_{ij} = \alpha \delta_j x_i$$

Hitung parameter koreksi bias (dipakai untuk mengubah v_{0j})

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

Perbaharui bobot dan bias:

8. Tiap unit keluaran (Y_k , $k = 1, \dots, m$) ubah bias dan bobotnya ($j = 0, \dots, p$):

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

- Tiap unit tersembunyi (Z_j , $j = 1, \dots, p$) ubah bias dan bobotnya ($i = 0, \dots, n$):

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

9. Test kondisi berhenti.

Satu *epoch* adalah satu siklus pasangan pola pelatihan lengkap. Biasanya, diperlukan beberapa kali *epoch* untuk pelatihan jaringan perambatan-balik. Algoritma ini memperbaharui bobot setelah tiap pola pelatihan disajikan.

Turunan $f'(y_{in_k})$ dan $f'(z_{in_j})$ dapat dinyatakan dengan parameter y_k dan z_j . Dasar matematis untuk algoritma perambatan-balik adalah teknik optimasi yang dikenal dengan *gradient descent*. *Gradient* suatu fungsi positif (dalam kasus ini fungsi galat, dan variabel adalah bobot jaringan) menggambarkan fungsi meningkat secara cepat. Gradient negatif memberi arah ketika fungsi menurun secara cepat.

Inisialisasi Bobot dan Bias

Inisialisasi acak. Pemilihan inisialisasi bobot akan mempengaruhi apakah jaringan mencapai galat pada minimum global (atau hanya lokal), jika demikian, seberapa cepat konvergensnya. Perubahan bobot antara dua unit

tergantung pada kedua turunan fungsi aktivasi unit di atas dan unit di bawahnya. Dengan alasan ini, sangatlah penting untuk menghindari pemilihan bobot awal yang akan membuat fungsi aktivasi atau pun turunannya menjadi nol. Nilai bobot awal jangan terlalu besar, jika tidak masukan awal ke tiap unit tersembunyi atau keluaran akan berada pada daerah ketika turunan fungsi *sigmoid* memiliki nilai sangat kecil (disebut daerah saturasi). Sebaliknya, jika bobot awal terlalu kecil, sinyal masukan pada unit tersembunyi atau pun unit keluaran akan mendekati nol, juga menyebabkan pelatihan jaringan akan sangat lambat.

Prosedur umum adalah menginisialisasi bobot (dan bias) dengan nilai acak antara $-0,5$ dan $0,5$ (atau antara -1 dan 1) atau pada rentang nilai lainnya yang sesuai. Nilai bisa positif atau pun negatif karena bobot akhir setelah pelatihan juga memiliki kedua tanda tersebut. Modifikasi sederhana inisialisasi acak, dikembangkan oleh Nguyen dan Widrow (1990), dijabarkan pada bagian berikut.

Panjang Waktu Pelatihan Jaringan

Motivasi aplikasi jaringan perambatan-balik adalah mencapai keseimbangan antara tanggapan yang benar untuk pola pelatihan dan tanggapan yang baik untuk pola masukan yang baru (keseimbangan antara memorisasi dan generalisasi), dan tidak perlu melanjutkan pelatihan lebih lama lagi untuk mencapai total galat kuadrat mencapai minimum. Hecht-Nielsen (1990) menyarankan memakai dua kumpulan data selama pelatihan: satu untuk pola pelatihan dan lainnya untuk pola pengujian. Keduanya harus

berbeda. Perubahan bobot didasarkan pada pola pelatihan. Pada interval pelatihan, dilakukan pengujian jaringan memakai pola pengujian, jika galat masih mengecil, pelatihan dilanjutkan. Ketika galat meningkat, jaringan mulai mengingat pola pelatihan terlalu spesifik (dan mulai kehilangan kemampuan untuk generalisasi). Pada titik ini, pelatihan dihentikan.

Representasi Data

Pada banyak permasalahan, vektor masukan dan vektor keluaran memiliki komponen dengan rentang nilai yang sama. Karena faktor satu pada koreksi bobot adalah aktivasi unit terkecil, maka unit-unit dengan aktivasi nol tidak akan dilatih. Hal ini menyarankan pelatihan akan meningkat jika masukan direpresentasikan dalam bentuk bipolar dan fungsi aktivasi yang dipakai adalah *sigmoid* bipolar.

Jumlah Lapisan Tersembunyi

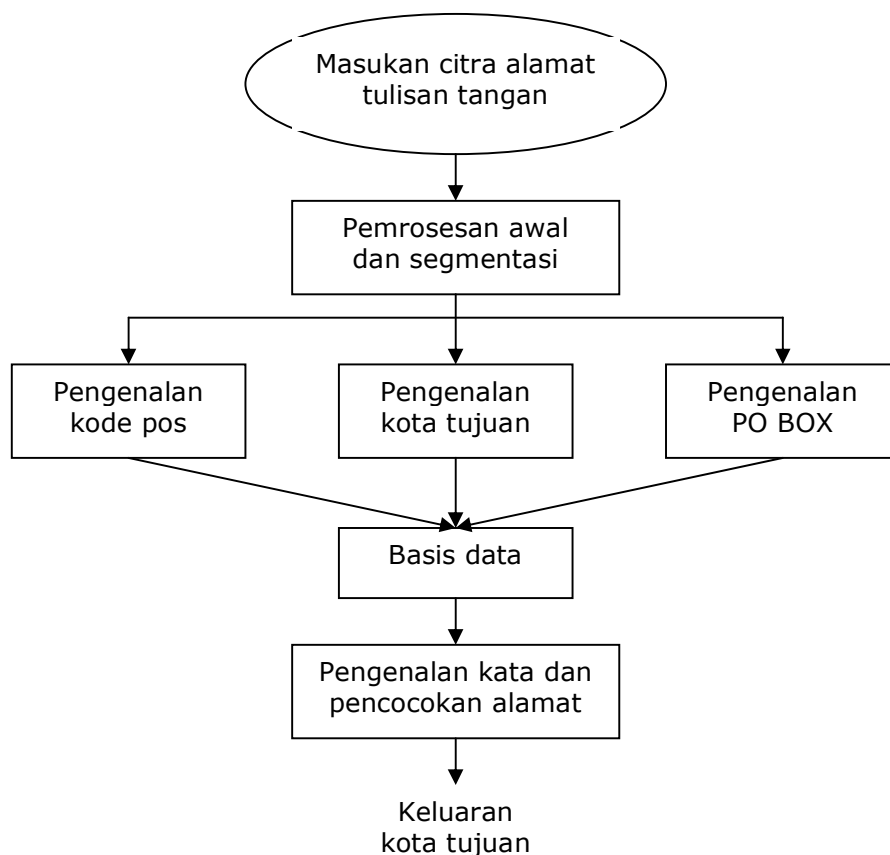
Satu lapisan tersembunyi telah cukup bagi jaringan perambatan-balik untuk memperkirakan pemetaan pola-pola masukan ke dalam pola-pola keluaran dengan tingkat keakuratan yang beragam, sedang dua lapisan tersembunyi mungkin akan membuat pelatihan akan lebih mudah pada situasi ataupun aplikasi tertentu.

II.2.2 Pemodelan Sistem

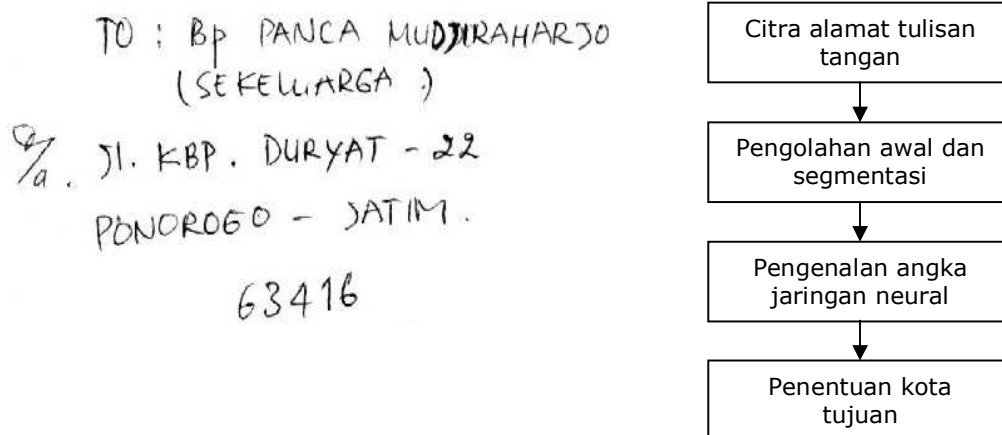
Sistem interpretasi alamat tulisan tangan oleh Paul Gader dkk. (1995) diperlihatkan pada Gambar 2.5. Pada sistem ini dapat dijelaskan secara umum, masukan adalah citra alamat tulisan tangan. Sebelum diumpankan ke

jaringan neural dilakukan terlebih dahulu pengolahan awal, terdiri atas: pemisahan baris, penghilangan derau, pembersihan coretan yang tak diinginkan di atas dan di bawah, koreksi kecenderungan, dan koreksi kemiringan. Setelah dilakukan pengolahan awal dilanjutkan dengan pemisahan kata atau blok kata. Kata atau blok kata ini kemudian disegmentasi menjadi primitif-primitif karakter atau bagian karakter.

Primitif-primitif karakter atau bagian karakter ini, diambil dua ciri transisi dan batangnya sebagai masukan jaringan neural, untuk dikenali karakternya. Dengan pemrograman dinamis akan dikenali daerah numeris, dan PO BOX. Kemudian membandingkan dengan basis data pos yang ada akan dikenali kata dan pencocokan alamat.



Gambar 2.5 Sistem interpretasi alamat tulisan tangan



Gambar 2.6 Diagram blok pengenalan kode pos

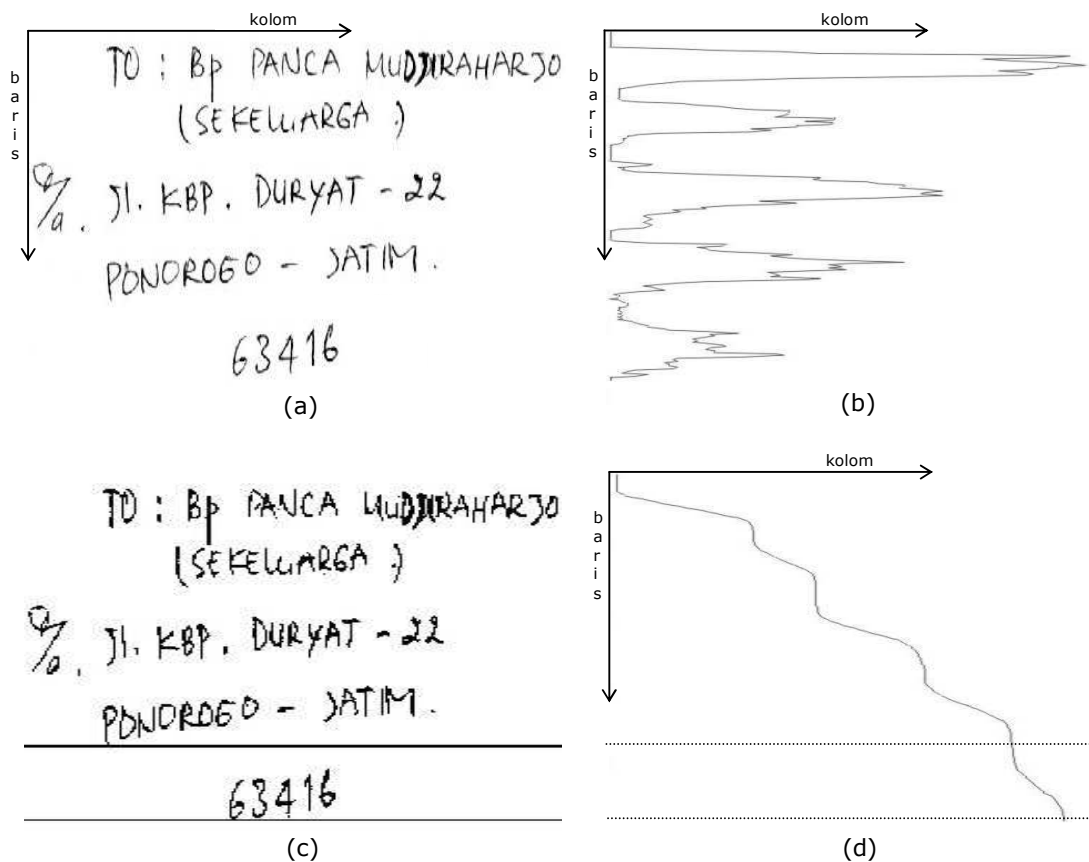
II.2.3 Pengenalan Kode Pos

Penelitian ini hanya bagian dari semua modul yang diperlukan untuk pengenalan keseluruhan sistem interpretasi alamat. Penelitian ini terfokus pada modul pengenalan kode pos alamat tulisan tangan. Diagram sistem pengenalan kode pos dengan contoh citra alamat tulisan tangan tampak pada Gambar 2.6. Keempat submodul yang berbeda pada sistem pengenalan kode pos akan diuraikan pada bagian berikut.

II.2.3.1 Pengolahan Awal dan Segmentasi

Pengolahan awal diperlukan untuk mengisi keterbatasan pada langkah penguraian ciri, seperti adanya derau pada citra masukan. Dilakukan normalisasi sebanyak mungkin citra masukan dalam upayanya untuk mengurangi variasi jenis tulisan tangan. Beberapa proses yang akan dipakai diuraikan pada bagian berikut.

Pencarian Baris Akhir



Gambar 2.7 (a) citra alamat asal (b) histogram latar depan (c) hasil pencarian baris akhir (d) distribusi histogram

Kode pos umumnya terletak di baris terakhir alamat, sehingga pertama kali perlu pencarian baris akhir. Pencarian baris akhir dibentuk dengan menganalisis histogram latardepan. Pemisahan baris terjadi pada global minimum diantara dua global maksimum histogram. Ini karena pada global minimum tersebut terjadi transisi dari satu baris ke baris berikutnya. Sedang bukit histogram menandakan adanya coretan atau tulisan di baris tersebut. Untuk memudahkan pencarian global-global minimum ini, pemisahan baris dilakukan dengan menganalisis distribusi histogram

latar depan. Suatu citra alamat, histogram latar depan, dan distribusi histogram diperlihatkan pada Gambar 2.7.

Untuk citra dengan m baris dan n kolom, histogram latar depan diperoleh:

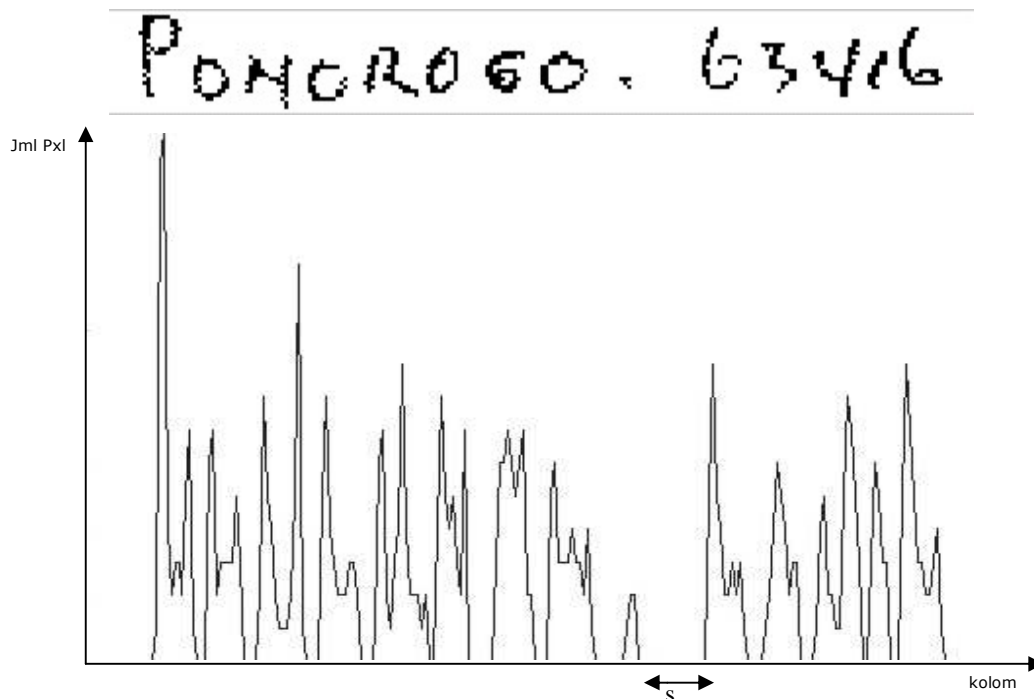
$$JmlPxl(k) = \sum_{b=1}^m Pxl(k, b) \quad k = 1, \dots, n \quad (6)$$

dengan distribusi histogram:

$$DistJmlPxl(k) = \sum_{k=1}^n JmlPxl(k) \quad (7)$$

Baris terbentuk pada perubahan kemiringan datar ke menanjak, karena pada perubahan tersebut terletak global minimum di antara global maksimum.

Segmentasi Baris ke Blok Kode Pos



Gambar 2.8 Analisa jarak kata menggunakan histogram latar depan

Kode pos alamat umumnya terletak di baris terakhir atau di bagian dari baris terakhir. Maka, untuk mengurangi komputasi saat pengenalan karakter perlu pembatasan kata. Segmentasi baris ke blok kode pos dilakukan dengan menganalisis histogram latar depan, seperti tampak pada Gambar 2.8. Spasi blok kode pos dalam suatu baris umumnya ditandai dengan histogram bernilai nol pada rentang jarak tertentu s .

Penghapusan Garis Kotak Kode Pos

Untuk kode pos dalam kotak, sebelum proses segmentasi blok kode pos ke karakter angka, garis-garis kotak kode pos perlu dihapus terlebih dahulu untuk menghindari kekeliruan pengenalan citra karakter. Penghapusan garis ini menggunakan operasi *opening* morfologi citra biner dengan batang horisontal dan vertikal, yang diuraikan dalam algoritma berikut:

I citra biner

H elemen struktur batang horisontal yang lebarnya sama dengan lebar citra,

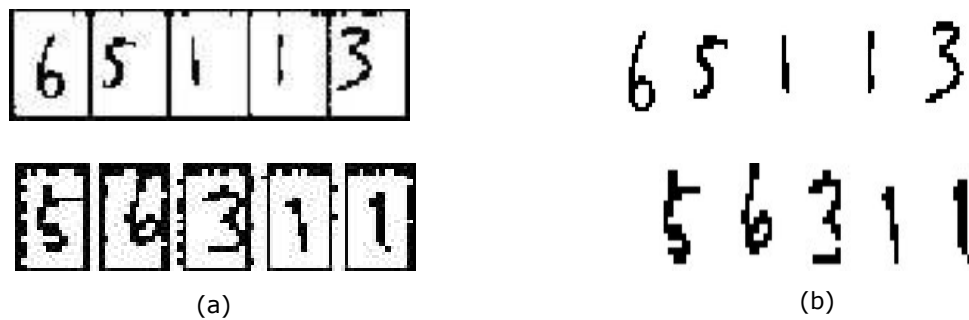
V elemen struktur batang vertikal yang tingginya sama dengan tinggi citra,

$M_H = I \circ H$ hasil opening I dengan H

$M_V = I \circ V$ hasil opening I dengan V

$R = I - M_H - M_V$ hasil citra biner

Penghapusan garis diperlihatkan dalam Gambar 2.9.



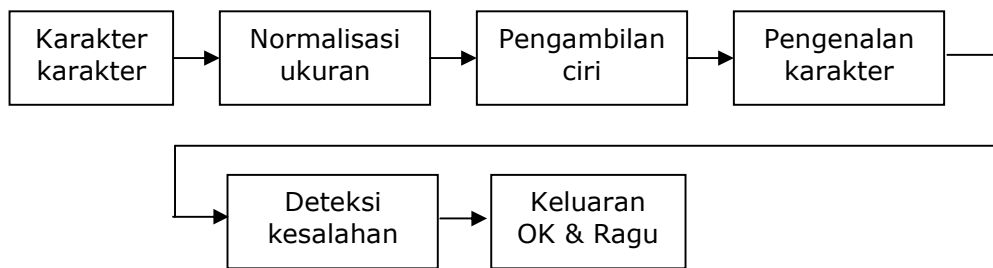
Gambar 2.9 Penghapusan garis kotak pos (a) citra asli (b) citra bersih

Segmentasi Blok Kode Pos ke Karakter

Penulisan angka umumnya terpisah antara angka yang satu dengan lainnya, sehingga proses segmentasi blok kode pos ke karakter menggunakan dua pendekatan. Pendekatan pertama dengan menghitung jumlah spasi antar karakter dalam suatu blok kode pos, untuk lima angka akan terdapat empat spasi. Pendekatan kedua dengan menghitung jumlah coretan dalam blok tersebut, idealnya untuk lima angka akan terdapat lima coretan.

II.2.3.2 Pengenalan Angka

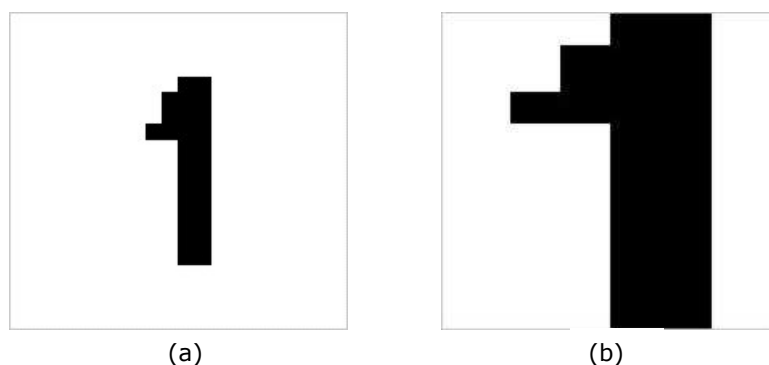
Pengenalan angka disini melibatkan segmentasi kata ke karakter individual yang diumpankan ke pengenalan karakter (*character recognition*). Pengenal karakter yang digunakan adalah jaringan perambatan-balik (*backpropagation*). Diagram pengenalan karakter diperlihatkan pada Gambar 2.10. Karakter hasil segmentasi sebelum diumpankan ke pengenalan karakter dinormalisasi ukurannya, kemudian diambil ciri batangnya untuk membentuk vektor masukan.



Gambar 2.10 Diagram blok pengenalan angka

Normalisasi Ukuran Karakter

Normalisasi ukuran karakter dilakukan karena tidak samanya ukuran karakter penulis yang satu dengan lainnya. Ukuran karakter memegang peranan penting dalam pengambilan ciri batang. Contoh karakter asli dan hasil normalisasinya ditunjukkan dalam Gambar 2.11.



Gambar 2.11 Normalisasi citra karakter (a) Citra karakter asli
(b) citra karakter ternormalisasi

Ciri Batang

Ciri batang dihitung pada citra biner karakter terpisah. Awalnya, delapan citra ciri dibentuk. Masing-masing citra ciri berhubungan dengan salah satu arah: timur, timur laut, utara dan barat laut, baik itu latar depan atau latar belakang. Masing-masing citra ciri memiliki nilai bilangan bulat

pada masing-masing lokasi yang menyatakan panjang batang terpanjang yang menempati titik itu dalam arah tersebut.

Suatu kalang tertutup, algoritma dua pelolosan digunakan untuk menghitung citra ciri. Dalam pelolosan arah maju citra dipayar (*scanned*) dari kiri ke kanan dan dari atas ke bawah. Pada masing-masing titik, baik itu citra ciri latardepan atau latarbelakang di perbaharui seperti dibawah ini. Pada pelolosan arah balik, maksimum dibalikkan dari bawah ke atas, dan dari kanan ke kiri. Kode buatan untuk perhitungan ciri batang pada latardepan adalah sebagai berikut.

```
% Pelolosan Arah Maju
For i = 1,2,.....,nbaris Do
    For j = 1,2,.....,nkolom Do
         $t(i,j) = t(i,j-1) + 1$ 
         $tl(i,j) = tl(i-1,j+1) + 1$ 
         $u(i,j) = u(i-1,j) + 1$ 
         $bl(i,j) = bl(i-1,j-1) + 1$ 
    End For
End For

% Pelolosan Arah Balik
For i = nbaris, nbaris - 1, ..... , 1 Do
    For j = nkolom, nkolom - 1, ..... , 1 Do
         $t(i,j) = \max(t(i,j), t(i,j+1))$ 
         $tl(i,j) = \max(tl(i,j), tl(i+1,j-1))$ 
         $u(i,j) = \max(u(i,j), u(i+1,j))$ 
         $bl(i,j) = \max(bl(i,j), bl(i+1,j+1))$ 
    End For
End For
```

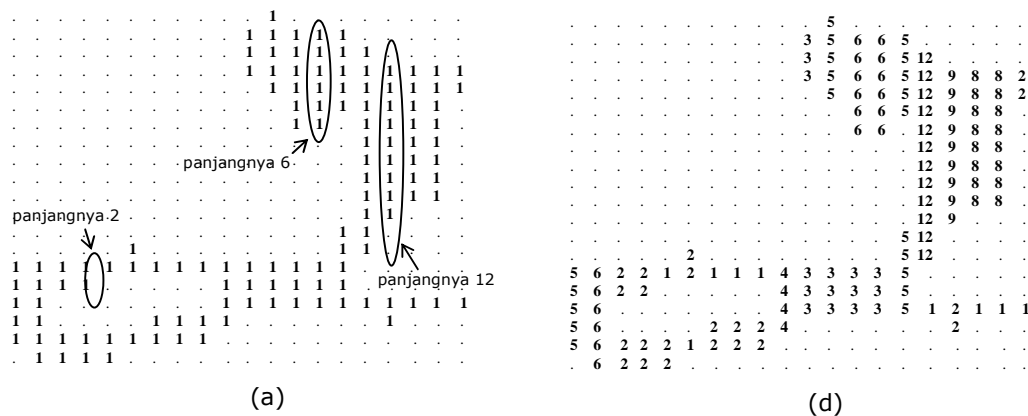
Vektor ciri dihitung dari citra ciri menggunakan daerah tumpang-tindih. Dua puluh satu daerah persegi diatur dalam tujuh baris dengan tiga daerah masing-masing digunakan; masing-masing daerah diperkirakan berukuran $t/4 \times l/2$ dengan t adalah tinggi citra dan l adalah lebar citra. Sudut kiri atas daerah tersebut diperkirakan pada posisi $\{(r,c)|r = 0, t/8, 2t/8, 3t/8, 4t/8, 5t/8, 6t/8 \text{ dan } c = 0, l/4, 2l/4\}$. Nilai dalam masing-masing daerah dalam masing-masing citra ciri dijumlahkan. Jumlah dinormalisasi antara nol dan satu dengan membagi jumlah tersebut dengan kemungkinan jumlah maksimum dalam daerah. Dimensi vektor ciri adalah $21 \times 8 = 168$. Lebih tepatnya, diberikan f_1, f_2, \dots, f_8 sebagai citra ciri bersesuaian dengan pola masukan dan z_i menyatakan daerah persegi berukuran $t/4 \times l/2$ dengan sudut kiri atas pada (r_0, c_0) . Nilai ciri ke- k yang bersesuaian dengan daerah z_i adalah:

$$z_{ik} = \left(\frac{1}{N} \right) \sum_{r=r_0}^{r_0+(l/2)} \sum_{c=c_0}^{c_0+(t/4)} f_k(r,c) \quad (8)$$

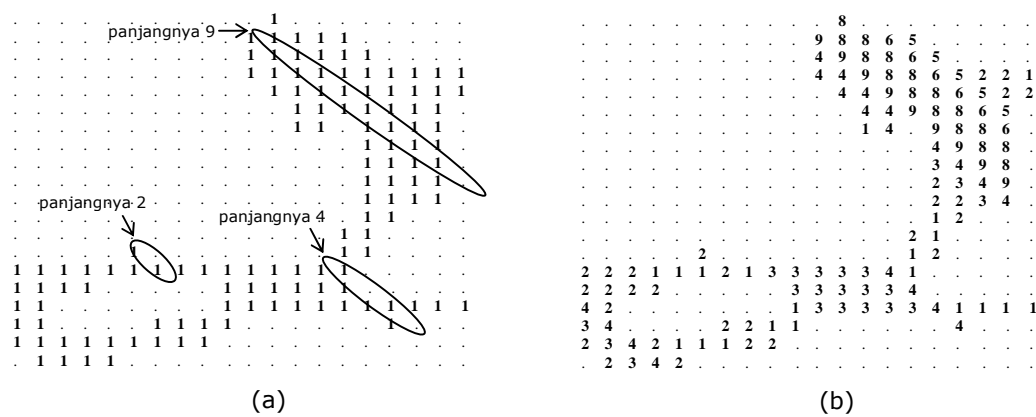
dengan:

$$\begin{aligned} N &= l, & \text{bila } f_k &= t, t/2 \text{ atau } bl \\ &= t, & \text{bila } f_k &= n \end{aligned}$$

Vektor dengan 168 nilai (z_{ik} : $i = 1, 2, \dots, 21$ dan $k = 1, 2, \dots, 8$) adalah ciri batang yang digunakan sebagai masukan jaringan neural pengenalan karakter. Pembuatan ciri batang latar depan diperlihatkan dalam Gambar 2.13 hingga 2.16 .

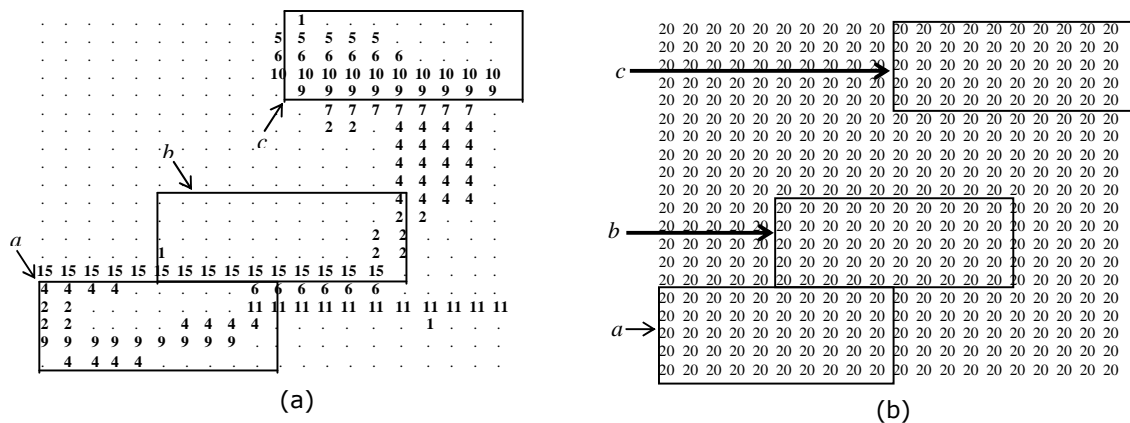


Gambar 2.15 Pembuatan ciri batang latar depan arah utara
(a) citra biner asli (b) ciri batang arah utara



Gambar 2.16 Pembuatan ciri batang latar depan arah barat laut
(a) citra biner asli (b) ciri batang arah barat laut

Untuk ciri batang latar belakang, pertama meng-*invers*-kan citra biner, kemudian pembuatan ciri batangnya dengan menghitung panjang '1' pada masing-masing arah mata angin mengikuti Gambar 2.13 hingga 2.16. Contoh perhitungan vektor ciri batang latar depan diperlihatkan dalam Gambar 2.17.



	0/1000	= 0	}	vektor latar depan arah timur
	0/1000	= 0		
	0/1000	= 0		
	0/1000	= 0		
	0/1000	= 0		
	155/1000	= 0.155		
$z_{ik}(a)$	→ 154/1000	= 0.154		
	142/1000	= 0.142		
	111/1000	= 0.111		
	25/1000	= 0.025		
	2/1000	= 0.002		
$z_{ik}(b)$	→ 155/1000	= 0.155	}	vektor latar depan arah timur laut
	271/1000	= 0.271		
	154/1000	= 0.154		
$z_{ik}(c)$	→ 243/1000	= 0.243		
	266/1000	= 0.266		
	117/1000	= 0.117		
	56/1000	= 0.056		
	103/1000	= 0.103		
	220/1000	= 0.220		
	141/1000	= 0.141		
	...		}	vektor latar depan arah utara
	...			
	...			
	...			
	...			
	...		}	vektor latar depan arah barat laut
	...			
	...			
	...			
	...			

Untuk latar depan terdapat 21×4 elemen dan untuk latar belakang terdapat 21×4 elemen, sehingga total terdapat 168 elemen vektor masukan untuk setiap karakter.

Gambar 2.17 Contoh perhitungan vektor ciri batang
 (a) ciri batang latar depan arah timur
 (b) kemungkinan panjang maksimum dalam area
 (c) vektor ciri batang

Kelebihan ciri batang adalah vektornya cocok digunakan sebagai masukan jaringan neural untuk pengenalan karakter, karena ciri ini menyatakan bentuk karakter. Sedang kelemahannya adalah berhubungan dengan penjendelaan, karena besarnya jendela tetap, maka citra karakter sebelum diambil cirinya harus memiliki ukuran yang tetap dan sama. Untuk mengatasi kelemahan ini dengan menormalisasi ukuran citra sebelum diambil cirinya.

Ciri Transisi

Ciri transisi ini menghitung lokasi dan jumlah transisi dari latar belakang ke latar depan sepanjang baris horizontal dan kolom vertikal suatu citra kata. Perhitungan transisi dikerjakan dari kiri ke kanan, kanan ke kiri, atas ke bawah, dan bawah ke atas. Karena dimensi yang konstan suatu ciri diperlukan untuk masukan ke jaringan neural.

Transisi citra karakter pada tiap arah dihitung. Tiap transisi melambangkan jarak-jarak pada arah tersebut. Jarak-jarak ini dihitung dengan urutan menurun. Sebagai contoh menghitung lokasi transisi arah kiri ke kanan, transisi dekat sisi kiri akan bernilai tinggi dan transisi pada sisi kanan bernilai rendah.

Jumlah transisi maksimum, M , dihitung pada tiap baris dan kolom. Jika transisi lebih besar dari M , maka hanya M transisi pertama yang dipakai, lainnya diabaikan (Gambar 2.18 dengan $M=3$). Jika transisi lebih kecil dari M , maka nilai selebihnya adalah nol. Jika diasumsikan terdapat n transisi pada

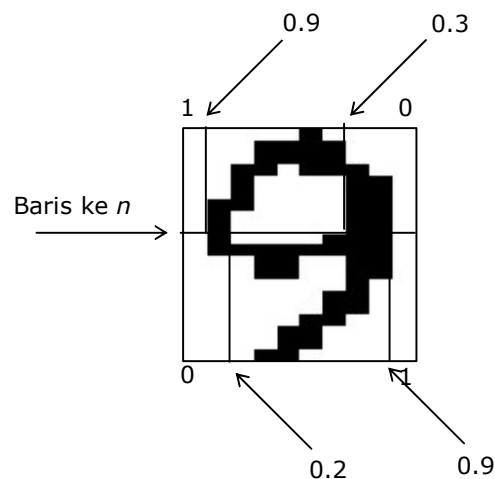
tiap baris/kolom pada lokasi (x_i, y_i) untuk $i = 1, 2, \dots, n$, algoritma perhitungan lokasi transisi disajikan sebagai berikut:

```

For i=1:min(n,M),
  If line_is_a_row then
    pi=xi;
    d=w;
  else
    pi=yi;
    d=h
  end;
  ti=1-(pi/d)
end;
if n<M then
  for I=n-1 : M
    ti=0;
  end;
end;

```

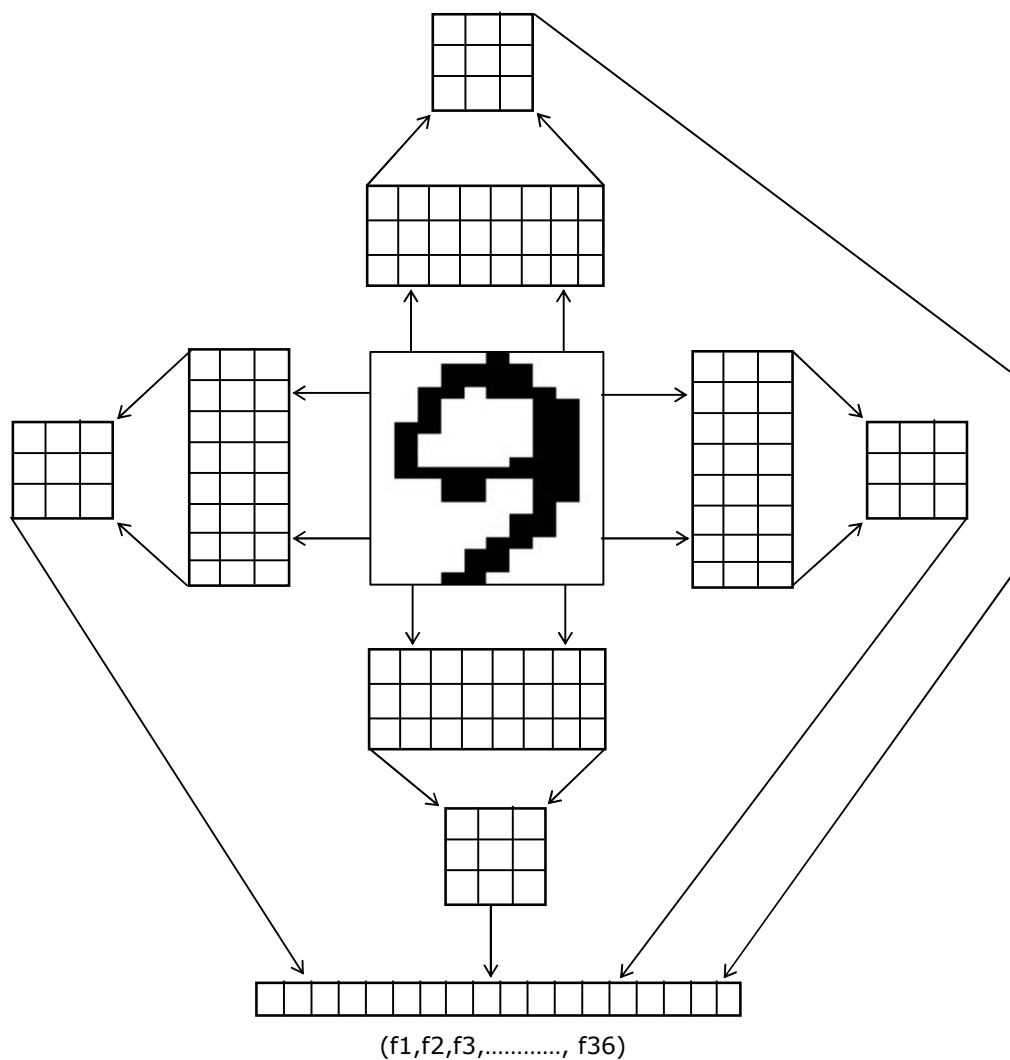
Setelah transisi dihitung, kemudian tiap kolom atau baris dari matriks transisi tersebut direrata untuk mendapatkan matriks rerata ciri transisi berdimensi $M \times M$ (Gambar 2.19).



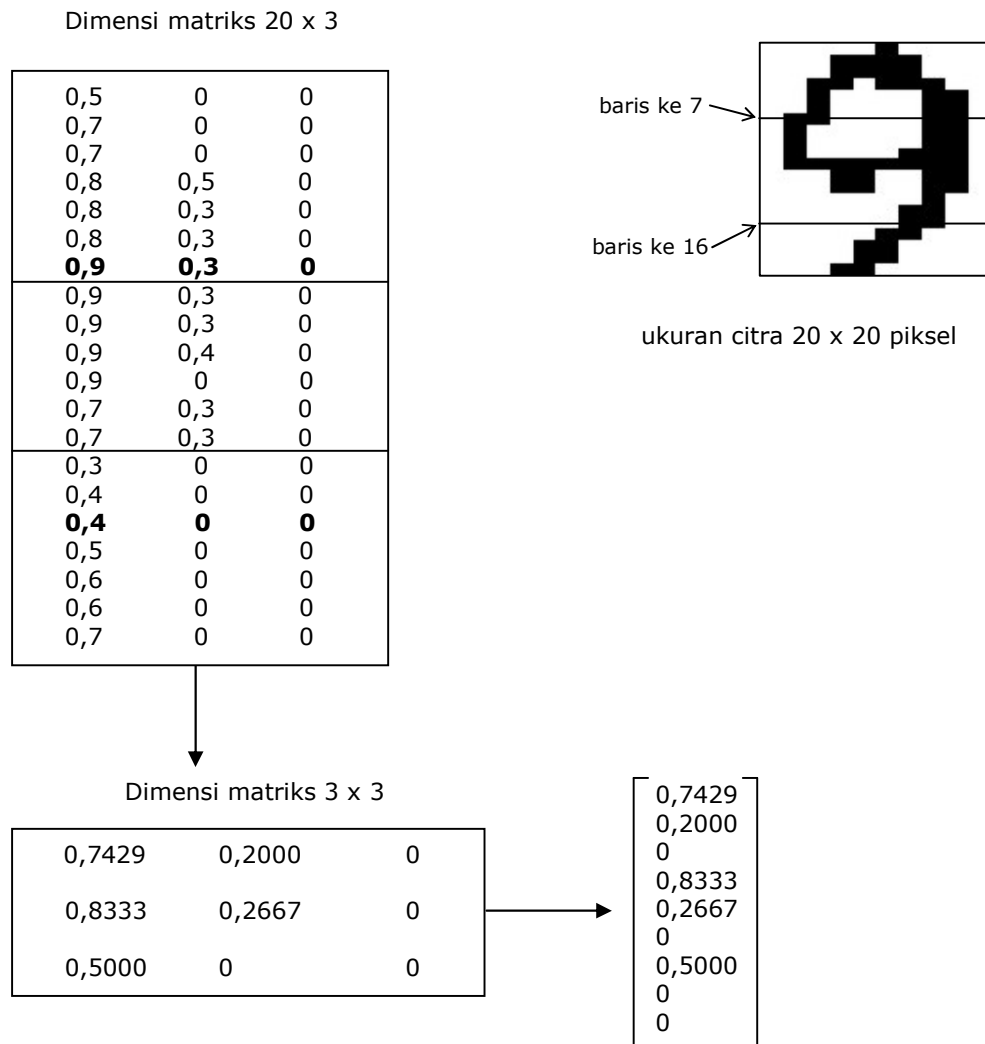
Gambar 2.18 Komputasi ciri transisi suatu baris

Kelebihan ciri transisi adalah vektornya cocok digunakan untuk masukan jaringan neural pada pengenalan sekumpulan karakter atau kata

yang menyatakan banyaknya karakter pembentuk kata tersebut. Sedang untuk pengenalan karakter, ciri ini menyatakan banyaknya coretan pembentuk karakter. Kelemahan ciri ini adalah tidak menyatakan bentuk karakter, sehingga dalam penerapan pengenalan karakter dan kata yang kompleks ciri ini selalu digunakan bersama-sama dengan ciri batang.



Gambar 2.19 Komputasi ciri transisi pada citra karakter

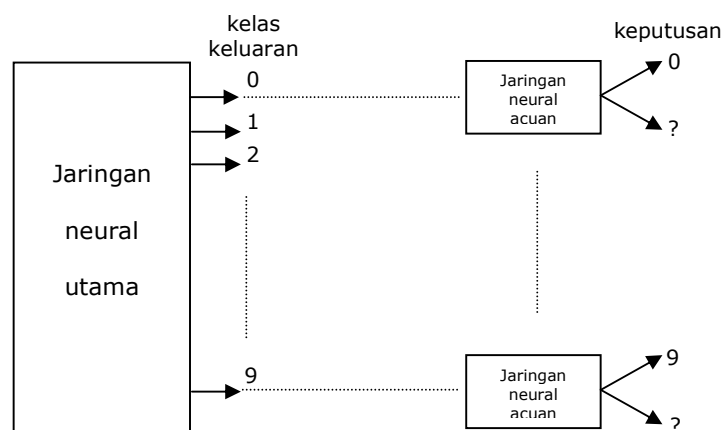


Gambar 2.20 Penguraian vektor ciri transisi dari kiri ke kanan-atas bawah

Sistem Deteksi Kesalahan

Keluaran jaringan neural umumnya berupa vektor kelas tujuan yang menyatakan besarnya nilai *confidence* untuk menyatakan karakter kelas tujuan. Untuk memutuskan apakah kelas tujuannya benar, maka disini perlu adanya konfirmasi, yang menyatakan bahwa karakter benar atau salah. Sistem deteksi kesalahan, sekali lagi menggunakan jaringan neural dengan

algoritma perambatan-balik. Jaringan dilatih dengan masukan acuan dan masukan selain acuan untuk mendapatkan keluaran benar dan ragu-ragu. Bila keluaran menyatakan ragu-ragu, maka sistem akan melaporkan sebuah kesalahan. Diagram blok sistem deteksi kesalahan ditunjukkan dalam Gambar 2.21.



Gambar 2.21 Diagram blok sistem deteksi kesalahan

II.3 Hipotesis

Dengan memperhatikan kelebihan dan kekurangan ciri batang dan ciri transisi citra karakter dalam proses pengenalan karakter, maka pemakaian ciri batang citra karakter tersegmentasi memberikan hasil yang dominan dalam hasil pengenalan karakter.

Selanjutnya, dengan menggunakan jaringan neural acuan yang dilatih dengan ciri batang karakter acuan dan karakter selain acuan dapat diterapkan sebagai sistem deteksi kesalahan, sehingga untuk citra yang cenderung tampak seperti dua angka akan dinyatakan ragu-ragu dan sistem akan melaporkan sebuah kesalahan.

II.4 Rencana Penelitian

Langkah awal penelitian ini adalah merancang proses pengumpulan sampel angka tulisan tangan, mengumpulkan sampel-sampel angka tulisan tangan, memproses sampel-sampel tersebut menjadi ciri batang yang siap diumpankan ke jaringan neural. Variasi sampel tulisan tangan yang ingin didapatkan sebisa mungkin sangat beragam. Untuk mewujudkannya sampel dilakukan di tiga tempat yang berbeda, di Jombang, Ponorogo dan Yogyakarta. Pemrosesan sampel-sampel tersebut dikerjakan dengan bantuan program Matlab, sehingga pengerjaannya dapat secara semi otomatis yang menghasilkan ciri batang.

Langkah selanjutnya mengumpulkan alamat-alamat tulisan tangan untuk pengujian, diantaranya dari surat yang pernah diterima, dari kantor pos terdekat, tulisan alamat rekan-rekan, dan tulisan alamat sanak famili.

Mempertimbangkan spesifikasi komputer yang dipakai pada penelitian ini yang sangat terbatas pada kemampuan melatih jaringan neural (Pentium 233 MMX, RAM 32 MHz), maka banyaknya jumlah neuron tersembunyi menyesuaikan kemampuan komputer tersebut.

Halaman ini tak perlu di print

Kata-kata tak terpakai sementara

Hipotesis

Adanya sebuah sistem deteksi kesalahan menyatakan bahwa sistem tersebut sempurna, karena dengan sendirinya sistem akan mengenali kesalahan yang dilakukannya. Untuk mencapai itu sangatlah sulit dilakukan, yang bisa dilakukan adalah memperluas klasifikasi BENAR dan SALAH, dengan BENAR, SALAH dan RAGU-RAGU. Klasifikasi BENAR menyatakan hasil pengenalan sesuai dengan karakter yang dituliskannya, klasifikasi SALAH menyatakan pengenalan tidak sesuai dengan karakter yang dituliskan, namun benar untuk

III. Cara Penelitian

III.1 Bahan Penelitian

Data utama pada penelitian ini adalah sampel angka tulisan tangan yang dipakai untuk pelatihan dan pengujian jaringan perambatan balik. Perancangan sampel mempertimbangkan jumlah yang seimbang atau sama untuk pelatihan maupun pengujian jaringan. Data kedua adalah sampel alamat tulisan tangan yang diperoleh dari surat yang pernah diterima, dari kantor pos terdekat, alamat tulisan tangan rekan mahasiswa dan alamat tulisan tangan sanak famili. Pengumpulan sampel tidak memberi kekangan pada para pengisi lembar isian, mereka diberi kebebasan pemakaian pena dan tipe tulisan. Data terakhir adalah basis data kode pos.

III.2 Alat yang Digunakan

Peralatan bantu yang dipakai pada penelitian ini:

- Komputer dengan spesifikasi: Pentium 233 MMX, RAM 32 MHz, *harddisk* satu giga byte.
- Pemayar.
- Program bantu MATLAB 5.3 beserta *toolbox neural network*, dan *image processing*, untuk pemrosesan sampel angka tulisan tangan dan pelatihan serta pengujian jaringan neural perambatan balik.
- Program *Corel Photo-paint 8*, untuk memotong-motong basis data angka tulisan tangan pada formulir isian.

III.3 Jalannya Penelitian

Metode penelitian yang dipakai adalah melakukan eksperimen untuk membangun suatu sistem pengenalan kode pos. Untuk mengimplementasikan sistem yang dimaksud dibuat suatu perangkat-lunak yang memakai kombinasi algoritma-algoritma yang didapat dari literatur.

Langkah awal penelitian tentang pengenalan kode pos ini adalah mengumpulkan literatur dari berbagai sumber, baik perpustakaan *off-line* maupun *on-line*. Dari rangkuman literatur yang didapatkan, Paul Gader dkk. telah meneliti pengenalan daerah angka dan nomor jalan pada alamat tulisan tangan di Amerika Serikat. Langkah selanjutnya adalah perancangan, pengumpulan, dan pemrosesan basis data serta pembuatan sistem untuk pengenalan kode pos. Sistem yang telah dicoba adalah sistem dengan pemakaian jaringan neural. Kemudian merancang dan mengamati jaringan acuan untuk menentukan sistem deteksi kesalahan. Untuk pengolahan basis data maupun sistem pengenalan kode pos akan dikerjakan menggunakan bahasa pemrograman MATLAB versi 5.3.

III.3.1 Basis Data

Pada bagian ini diuraikan basis data citra untuk pengenalan kode pos. Diawali dengan merancang basis data, kemudian mengumpulkan sampel angka tulisan tangan dan memrosesnya menjadi basis data citra tulisan tangan.

Basis Data Sebelumnya

Beberapa basis data karakter tulisan tangan dan huruf digital telah tersedia di berbagai institusi seperti NIST, CEDAR (*Center of Excellence for Document Analysis and Recognition*) dan CENPARMI (*Centre for Pattern Recognition and Machine Intelligence*). Institusi ini memungkinkan tim-tim peneliti mana pun memakai basis data yang telah ada untuk pelatihan kelayakan algoritma pengenalan mereka, juga sebagai basis perbandingan hasil-hasil yang telah mereka peroleh.

Namun untuk mendapatkan basis data dari institusi tersebut yang telah berbentuk piringan kecil (CD) tidaklah murah. Karena harga untuk sekeping CD dari CEDAR adalah 950 US dollar, sedang untuk mendapatkan kopi CD dari NIST secara berkelompok harus mengeluarkan biaya sebesar 150 US dollar. Alasan-alasan inilah yang menyebabkan pembuatan basis data mutlak dilakukan.

Pengumpulan Sampel Angka

Penelitian ini pada dasarnya adalah pengenalan angka tulisan tangan yang diaplikasikan pada pengenalan kode pos, sehingga berbagai data yang dikumpulkan untuk melatih dan menguji jaringan neural adalah data angka tulisan tangan. Secara sederhana pengumpulan data dengan meminta beberapa orang menuliskan angka pada form yang telah tersedia. Setiap orang menuliskan angka 0 hingga 9 sebanyak dua kali. Contoh formulir pengisian angka terlihat pada Gambar 3.1. Pengambilan sampel dilakukan di tiga tempat yang berbeda yaitu di SMPN 5 Jombang, SDN Banyudono

Ponorogo dan rekan-rekan mahasiswa di Yogyakarta serta sanak famili.

Didapatkan sejumlah 15 lembar isian dengan 600 penulis yang berbeda.

1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6

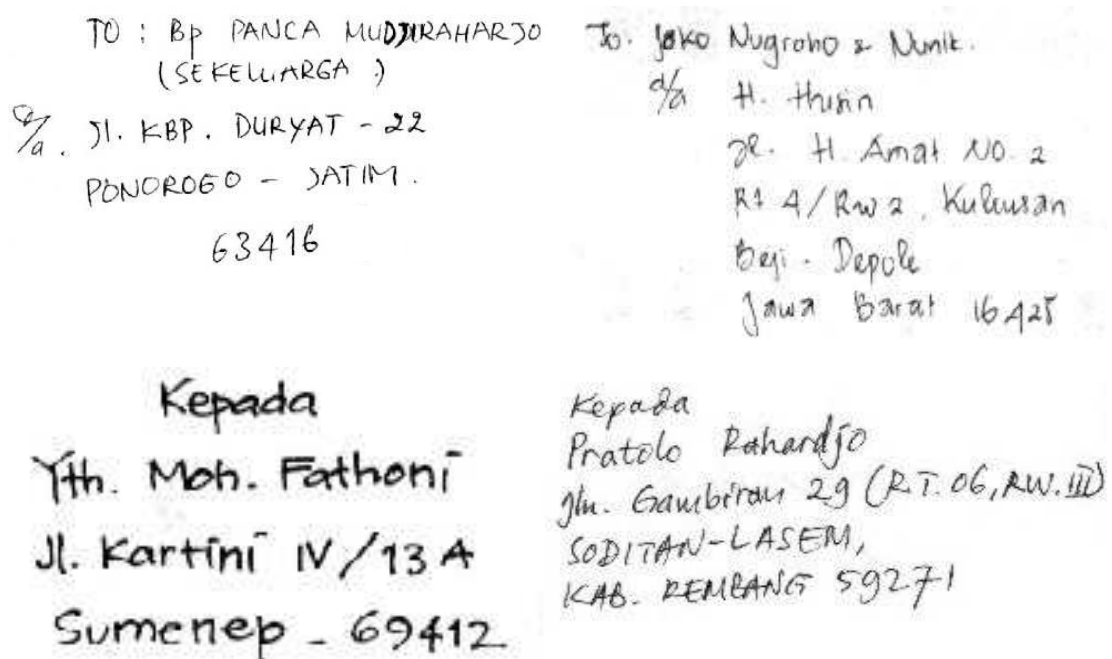
Tolong tuliskan angka seperti contoh di atas, untuk data, trim's !!!

1	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
2	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
3	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
4	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
5	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
6	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
7	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
8	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
9	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
10	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
11	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
12	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
13	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
14	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
15	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
16	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
17	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
18	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
19	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
20	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
21	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
22	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
23	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
24	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
25	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
26	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
27	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
28	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
29	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
30	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
31	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
32	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
33	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6

Gambar 3.1 Contoh formulir isian untuk basis data angka tulisan tangan

Pengumpulan Sampel Alamat Tulisan Tangan

Untuk pengujian sistem pengenalan kode pos diperlukan sampel alamat tulisan tangan yang berkode pos. Sampel ini dikumpulkan dari surat yang pernah diterima, meminjam, dan memfotokopi surat-surat dari kantor pos terdekat, meminta rekan mahasiswa dan sanak famili menuliskan alamat suratnya. Karena fokus penelitian ini hanya pada pengenalan kode pos dengan mengindahkan proses pengambilan citra alamat dari latar-belakangnya, maka untuk membatasi permasalahan sampel alamat diambil dari amplop dengan latarbelakang putih polos. Beberapa contoh sampel alamat tulisan tangan diperlihatkan pada Gambar 3.2.



Gambar 3.2 Contoh alamat tulisan tangan

III.3.2 Pemrosesan Basis Data

Sampel-sampel tulisan tangan yang telah didapatkan dipayar pada 300 dpi (*dot per inch*) dengan tiap piksel disimpan dalam satu bit (citra hitam-putih).

III.3.3 Perancangan Arsitektur Jaringan Utama

Masukan jaringan perambatan-balik adalah vektor ciri batang dan transisi citra karakter dan keluarannya adalah angka yang dimaksud, sehingga arsitektur jaringan pertama yang akan dibangun memiliki 168 neuron masukan dan 10 neuron keluaran, dan komponen yang melibatkan neuron pada lapisan tersembunyi yang jumlahnya akan dicari secara optimal, sedang arsitektur jaringan kedua memiliki 36 neuron masukan dan 10 neuron keluaran.

Umumnya untuk pengenalan karakter tulisan tangan memakai satu lapisan tersembunyi, sehingga disini juga menggunakan satu lapisan tersembunyi.

Seperti disinggung dalam tinjauan pustaka, yaitu kesulitan dalam menerapkan jaringan neural adalah tidak adanya rumusan baku tentang jumlah lapisan tersembunyi, fungsi aktivasi dan kecepatan pembelajaran untuk mendapatkan jaringan yang optimal, maka sebelum menetapkan jaringan utama diperlukan uji coba parameter-parameter tersebut sehingga diperoleh jaringan yang optimal.

III.3.4 Perancangan Sistem Deteksi Kesalahan

Sistem deteksi kesalahan disini menggunakan jaringan neural yang dilatih dengan karakter acuan dan karakter selain acuan untuk mendapatkan dua kelas keluaran yaitu keluaran BENAR dan RAGU-RAGU. Jaringan neural disini juga dilatih dengan algoritma perambatan-balik. Untuk jaringan acuan NOL, karakter acuannya adalah 0 dan karakter selain acuannya adalah 1,2, ... 9. Jaringan acuan SATU, karakter acuannya adalah 1 dan karakter selain acuannya adalah 0,2,3,... 9, dan seterusnya.

III.4 Kesulitan-kesulitan dan Cara Pemecahannya

Banyaknya data angka tulisan tangan yang dibutuhkan untuk pelatihan dan pengujian jaringan neural membutuhkan banyak waktu serta ketelatenan dan kesabaran dalam memotong basis data angka pada formulir isian.

Tidak adanya rumusan baku dalam menentukan jumlah lapisan tersembunyi, dan fungsi aktivasi untuk mendapatkan jaringan yang optimal. Hal ini diatasi dengan percobaan perbandingan parameter-parameter diatas sehingga diperoleh jaringan yang optimal. Percobaan ini memerlukan waktu yang cukup lama.

Keterbatasan spesifikasi komputer membuat pengerjaan pelatihan jaringan dengan mencoba-coba berbagai variasi pada neuron tersembunyi berjalan sangat lambat. Karena keterbatasan waktu dan biaya untuk melakukan penelitian ini, lambatnya proses tersebut mengakibatkan

penelitian tidak dapat mengkreasikan arsitektur jaringan neural lebih jauh lagi.

IV. Hasil Penelitian dan Pembahasan

Pada bab ini akan diuraikan hasil penelitian tiap-tiap modul pengenalan kode pos berdasarkan landasan teori di Bab II, diikuti dengan pembahasan untuk tiap-tiap modul pengenalan kode pos.

IV.1 Pengolahan Awal dan Segmentasi

Pengolahan awal diperlukan untuk mengisi keterbatasan pada langkah penguraian ciri, seperti adanya derau pada citra masukan. Dilakukan normalisasi sebanyak mungkin citra masukan dalam upayanya untuk mengurangi variasi jenis tulisan tangan. Hasil yang telah dilakukan diuraikan pada bagian berikut.

Pencarian Baris Akhir

Seperti dijelaskan pada landasan teori, pemisahan baris dilakukan dengan menganalisis grafik distribusi histogram seperti tampak pada Gambar 2.7 (d). Dari hasil penelitian ditentukan duakali pencarian lembah histogram, pertama lembah sementara dengan mencari perubahan kemiringan grafik distribusi hitogram dengan perhitungan:

$$\text{distribusi}_{i+10} - \text{distribusi}_i < \text{distribusi}_i - \text{distribusi}_{i-11}$$

dengan distribusi_i adalah distribusi histogram baris ke i . Kemudian lembah yang sebenarnya adalah global minimum diantara dua lembah sementara. Program pemisahan baris ini dapat dilihat pada **baris.m**. Beberapa hasil pemisahan baris terlihat pada Gambar 4.1.

<p>Kepada</p> <p>Yth. Albertus Andrejante</p> <p>Jl. Batu Jajar 1/32</p> <hr/> <p>Malang - 65113</p> <hr/>	<p>Kepada</p> <p>Budi Setiyono</p> <p>Gembong Rt 03 kw 01 No. 8</p> <p>Bojongsari</p> <hr/> <p>Purbalingga - 53362</p> <hr/>
--	--

kepada :

Yth. M. R. Ferry Sulisno

Jl. Raya Condut Gg. 4 L218 No 57

Batu Ampar

Garut Time 13520

Gambar 4.1 Hasil pengujian pencarian baris akhir

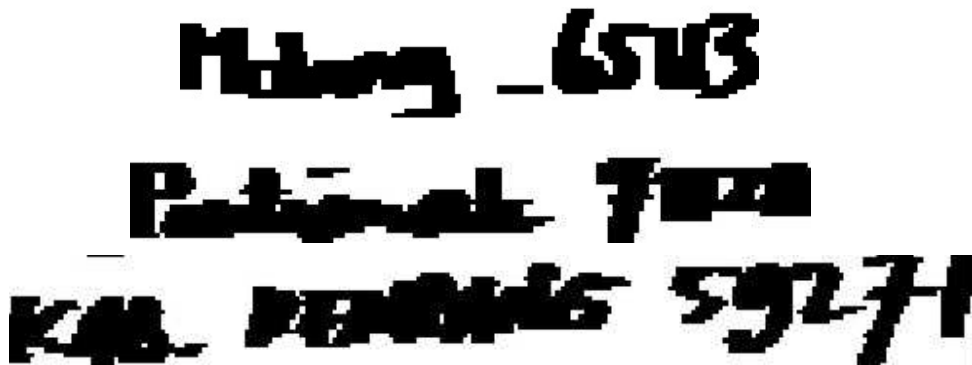
Segmentasi Baris ke Blok Kode Pos

Pada penulisan alamat surat baku, kode pos terletak pada baris terakhir dan blok kata terakhir. Seperti disinggung dalam landasan teori, untuk pencarian blok kata terakhir dilakukan dengan menganalisis histogram latar depan dan mencari spasi antar kata. Untuk menutupi spasi antar karakter dalam blok kode pos tersebut, dilakukan proses dilasi pada citra baris terakhir:

```
structuringElemen = ones(1,5);
```

```
DataDilasi = dilate (BarisAkhir,StructuringElemen);
```

Beberapa hasil dilasi diperlihatkan dalam Gambar 4.2. Listing program dalam berkas **Kata.m**.



Gambar 4.2 Hasil dilasi citra

Segmentasi Blok Kode Pos ke Karakter

Segmentasi blok kode pos ke karakter dilakukan dengan menghitung jumlah karakter dan menghitung jumlah spasi antar karakter. Untuk menghitung jumlah karakter menggunakan sintaks MATLAB:

```
LabelCitra = bwlabel(Citra);  
JumlahKarakter = max(max(LabelCitra));
```

Listing program dalam berkas **Segmentasi.m**.

IV.2 Pengenalan Angka

Mengacu pada diagram blok pengenalan angka di Gambar 2.10 di depan, berikut akan diuraikan hasil normalisasi ukuran karakter, pengambilan ciri batang, ciri transisi dan pengenalan angka dengan jaringan perambatan balik.

Normalisasi Ukuran Karakter

Normalisasi ukuran karakter dilakukan karena tidak samanya ukuran karakter penulis yang satu dengan lainnya. Ukuran karakter memegang peranan penting dalam pengambilan ciri batang. Pada penelitian ini semua karakter dinormalisasi 20x20 piksel. Normalisasi dilakukan dengan fungsi Matlab:

```
Normalisasi = imresize(CitraAsli,[20 20], 'nearest')
```

Program lengkap normalisasi ukuran karakter di berkas **Normalkan20x20.m**.

Ciri Batang

Mengacu pada landasan teori ciri batang, algoritma pembentukan ciri batang di halaman 26, dan persamaan (8), pada penelitian ini dicoba membuat vektor ciri batang 168 elemen. Evaluasi kedua vektor ciri batang tersebut diuraikan di bagian pengenalan angka setelah ini. Program selengkapnya di berkas **BuatCB21x8x1.m**

Ciri Transisi

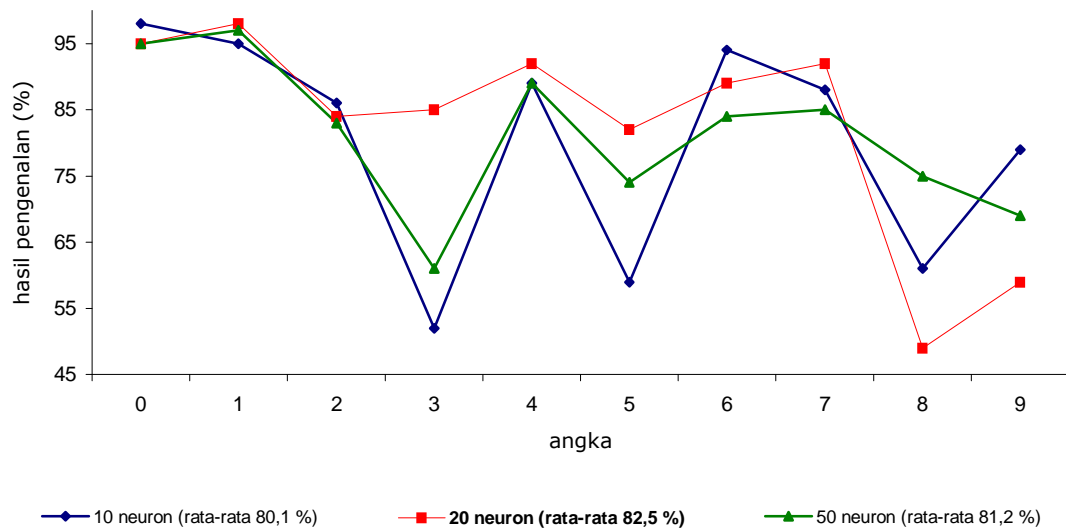
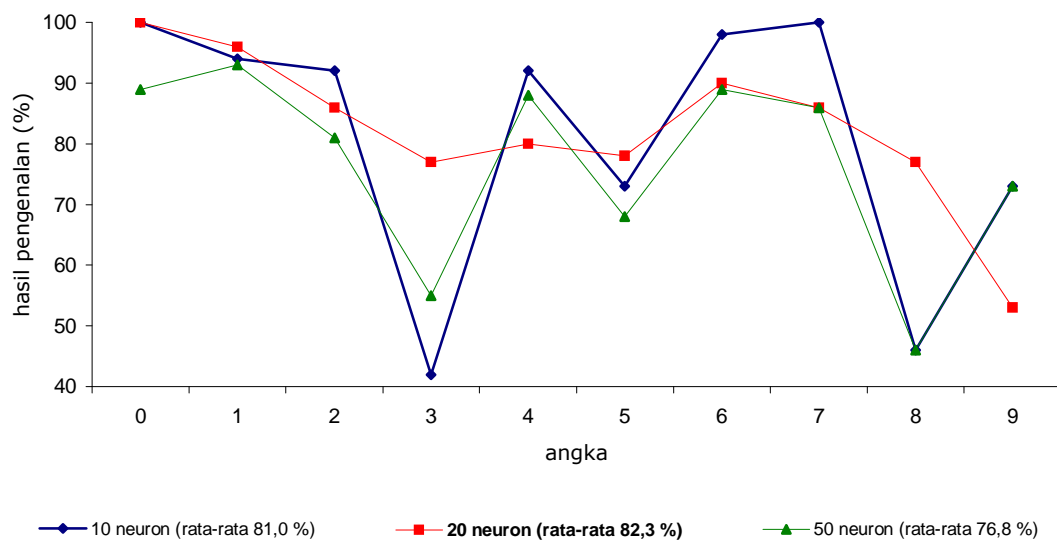
Ciri transisi seperti dijelaskan dalam landasan teori di depan, menghasilkan vektor masukan sebesar 36 elemen. Evaluasi vektor ciri bersama-sama dengan vektor ciri batang diuraikan setelah berikut ini. Program membuat vektor ciri transisi di berkas **BuatCT3x3.m**.

Ciri Batang dan Ciri Transisi

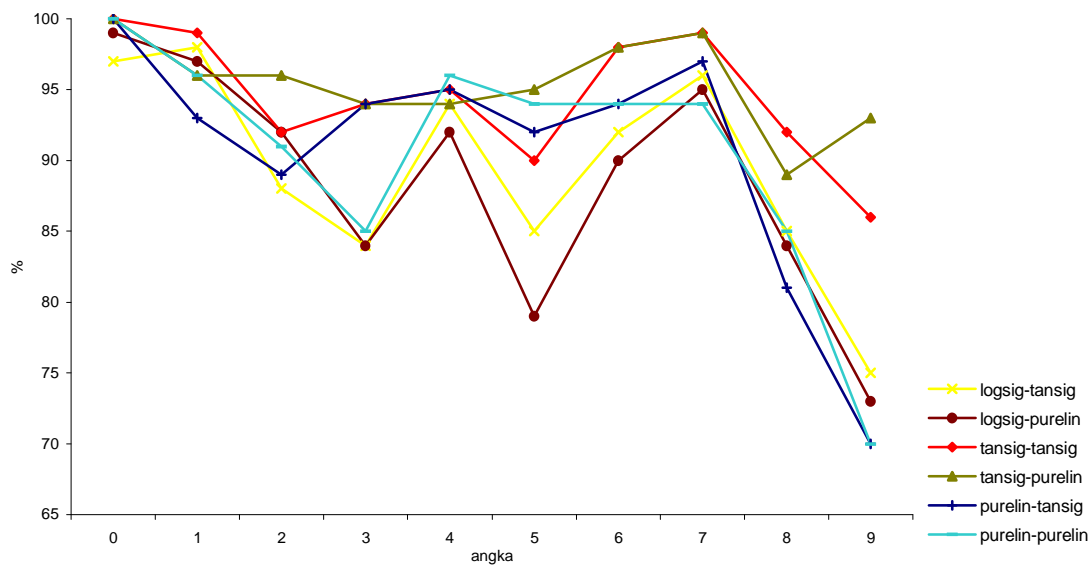
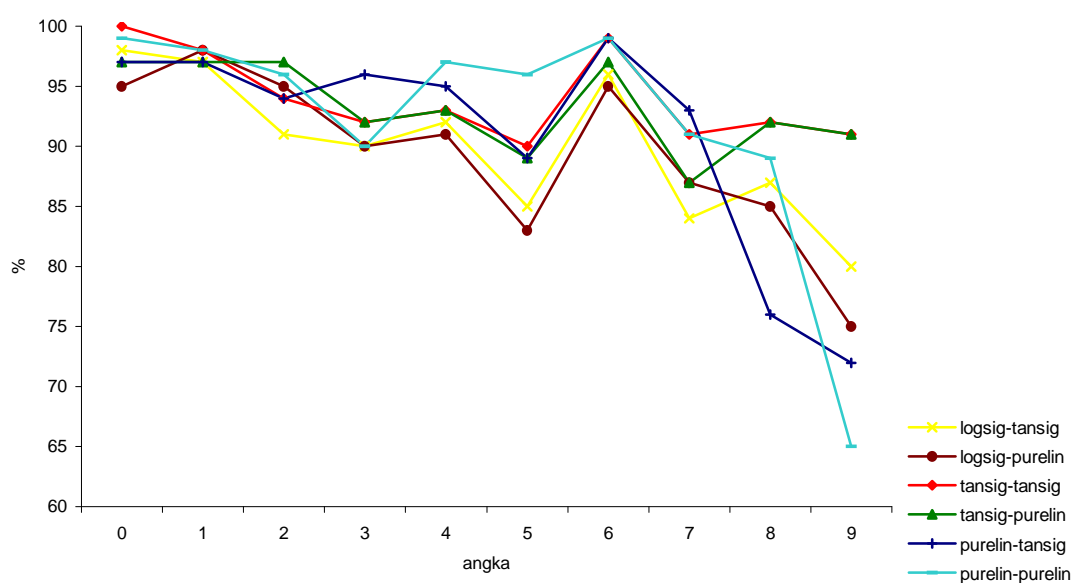
Seperti diuraikan dalam kelebihan dan kelemahan kedua ciri di depan, penggunaan ciri transisi dalam pengenalan karakter, harus digunakan bersama-sama dengan ciri batang. Sehingga disini perlu diuji hasil pengenalan gabungan kedua ciri tersebut. Gabungan kedua ciri diperoleh dengan menjumlahkan vektor ciri batang dengan vektor ciri transisi.

Pengenalan Angka

Pada bagian ini akan diuraikan hasil pengenalan angka menggunakan jaringan perambatan-balik dengan masukan berupa vektor ciri batang seperti diuraikan di atas dan satu lapisan tersembunyi. Keluaran jaringan adalah sepuluh klas yaitu angka 0 hingga 9. Untuk mengoptimalkan jaringan dicoba beberapa variasi terhadap jumlah vektor masukan, jumlah neuron tersembunyi, dan fungsi aktivasi. Pengujian terdiri atas pengujian memorisasi yaitu kemampuan tanggapan yang benar terhadap pola masukan yang dipakai untuk pelatihan jaringan dan pengujian generalisasi yaitu kemampuan memberikan tanggapan yang layak untuk masukan yang sejenis namun tidak identis dengan yang dipakai pada pelatihan. Jumlah data pelatihan dan pengujian masing-masing sebanyak 100 data angka. Hasilnya dapat dilihat pada Grafik 4.1, 4.2, 4.3, 4.4 dan 4.5 berikut.

Grafik 4.1 Pengujian memorisasi berdasarkan jumlah neuron tersembunyi**Grafik 4.2** Pengujian generalisasi berdasarkan jumlah neuron tersembunyi

Berdasarkan Grafik 4.1 dan 4.2 jumlah 20 neuron tersembunyi lebih baik.

Grafik 4.3 Hasil pengenalan memorisasi berdasarkan fungsi aktivasi**Grafik 4.4** Hasil pengenalan generalisasi berdasarkan fungsi aktivasi

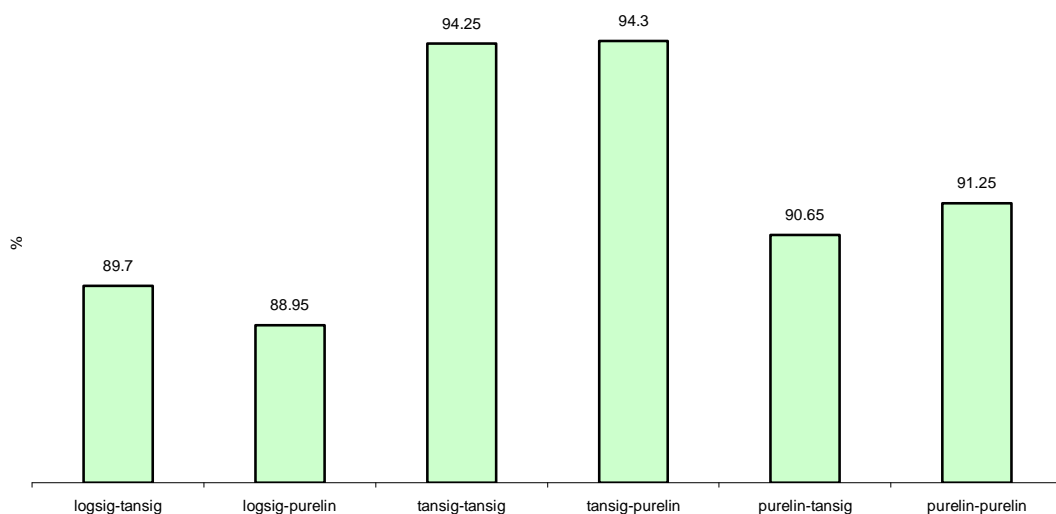
Dari keempat grafik terlihat bahwa pengenalan angka 3, 5, 8 dan 9 lebih rendah dibandingkan dengan angka lainnya, ini karena keragaman penulisan angka-angka tersebut mirip dengan angka-angka yang lain, angka 3 mirip dengan angka 0, 5, 8 dan 9; angka 5 mirip dengan angka 3, 6 dan 8;

angka 8 mirip dengan angka 0, 3, 5, 6 dan 9; serta angka 9 mirip dengan angka 0, 3, 4, 5, 7 dan 8. Sehingga pola masukan angka-angka tersebut sering dikenali angka miripnya.

Tabel 4.1 Rata-rata pengenalan angka berdasarkan fungsi aktivasi

	Rata-rata pengenalan angka (%)					
	logsig-tansig	logsig-purelin	tansig-tansig	tansig-purelin	purelin-tansig	purelin-purelin
memorisasi	89,4	88,5	94,5	95,4	90,5	90,5
generalisai	90	89,4	94	93,2	90,8	92
total	89,7	88,95	94,25	94,3	90,65	91,25

Grafik 4.5 Pengenalan total berdasarkan fungsi aktivasi



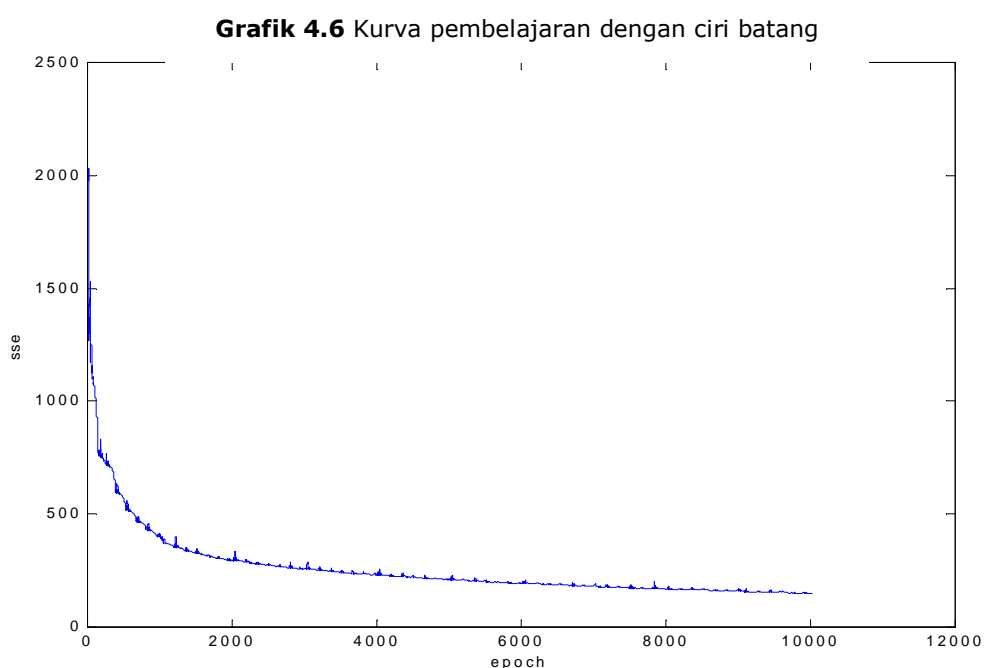
Dari Grafik 4.5 terlihat dengan fungsi aktivasi *logsig*, hasil pengenalan angka rendah, ini menyatakan pengenalan angka dengan vektor ciri batang dan ciri transisi sebagai masukan jaringan neural memberikan hasil yang kurang baik dengan fungsi aktivasi dengan rentang nilai 0 hingga +1, dan

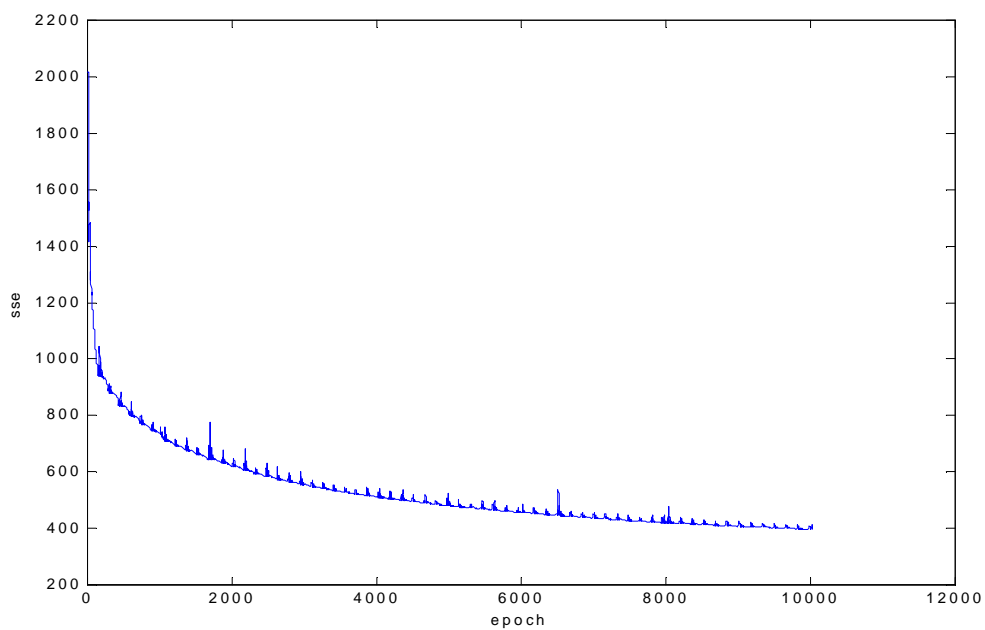
sebaliknya memberikan hasil yang baik dengan menggunakan fungsi aktivasi dengan rentang nilai -1 hingga $+1$.

Dari kelima grafik diperoleh jaringan perambatan-balik untuk pengenalan angka yang optimal dengan 20 neuron tersembunyi dan fungsi aktivasi untuk lapisan pertama menggunakan *tansig* dan lapisan kedua menggunakan *purelin*.

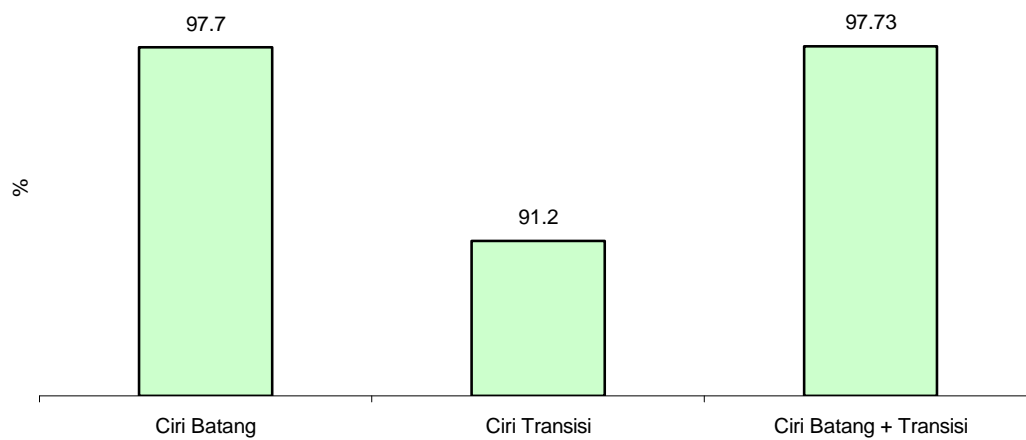
Untuk mencapai hasil pengenalan yang optimal jaringan dilatih hingga mencapai konvergensi, pada penelitian ini dengan data pelatihan 200 data jaringan dilatih hingga 10000 epoch, dilanjutkan dengan 500 data dilatih 250 epoch, kurva pembelajarannya diperlihatkan dalam Grafik 4.6 dan 4.7.

Terlihat pada Grafik 4.6 dan 4.7 bahwa proses pelatihan mulai epoch 9000 dan seterusnya, perubahan total galat kuadrat jaringan sangat kecil atau tidak berarti, ini berarti dengan data pelatihan 200 perubahan bobot jaringan telah optimal pada epoch tersebut. Untuk ciri batang sse-nya 149,59 atau mse-nya 0,0075 dan untuk ciri transisi sse-nya 396,19 atau mse-nya 0,0198.

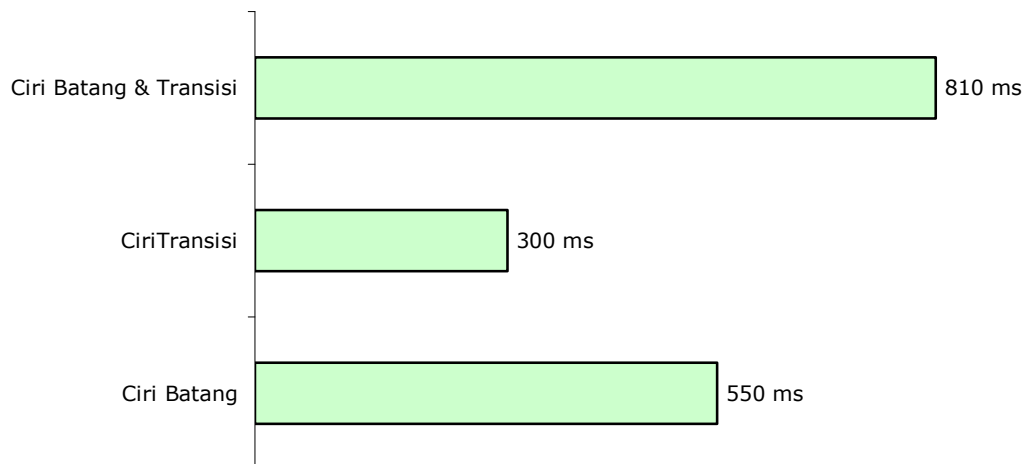


Grafik 4.7 Kurva pembelajaran dengan ciri transisi

Perbandingan pengenalan angka dan waktu komputasi dengan menggunakan ciri batang, ciri transisi dan gabungan keduanya berturut-turut diperlihatkan dalam Gambar 4.8 dan 4.9.

Grafik4.8 Perbandingan hasil pengenalan angka berdasarkan ciri yang digunakan

Grafik 4.9 Perbandingan waktu eksekusi berdasarkan ciri yang digunakan (menggunakan prosesor Pentium MMX 233 MHz)

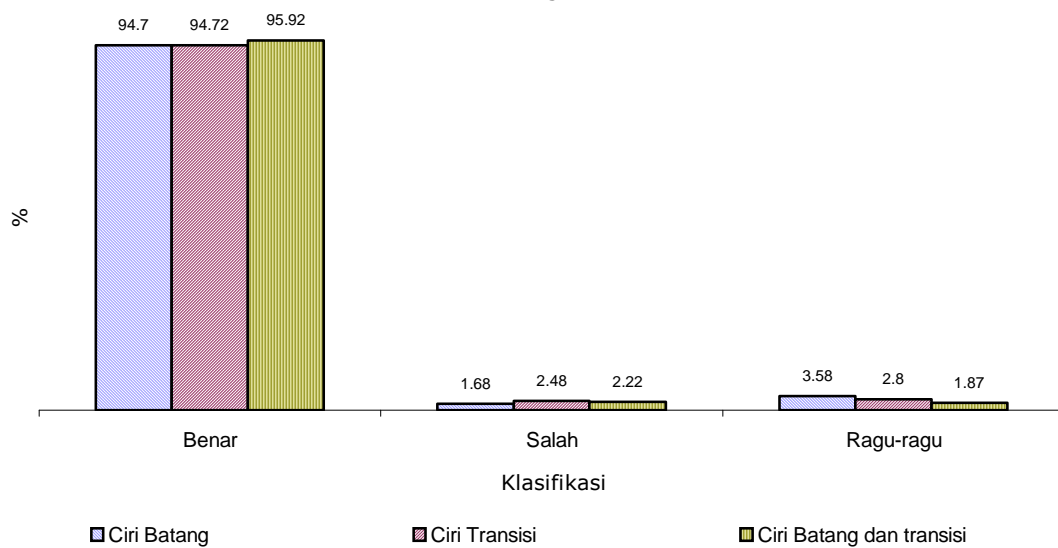


Sistem Deteksi Kesalahan

Seperti dibahas dalam landasan teori, sistem deteksi kesalahan dilakukan dengan membangun jaringan acuan setelah jaringan utama. Hasil pengenalan oleh jaringan utama, citra karakternya akan diuji lagi dengan jaringan acuan untuk diputuskan apakah karakter tersebut benar atau ragu-ragu. Bila hasil pengenalan oleh jaringan acuan ragu-ragu, maka sistem akan melaporkan sebuah kesalahan.

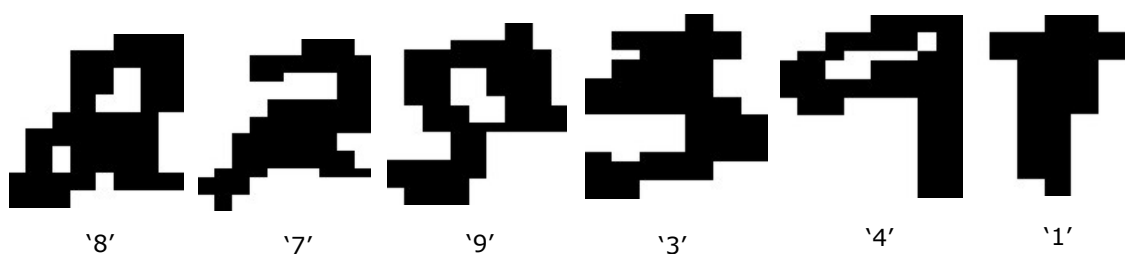
Untuk mengetahui keakuratan jaringan acuan, sekali lagi dilakukan pengamatan terhadap hasil pengenalan berdasarkan ciri citra yang digunakan untuk masukan jaringan acuan. Hasil pengamatan diperlihatkan pada Grafik 4.10.

Grafik 4.10 Perbandingan hasil keluaran jaringan acuan berdasarkan ciri yang digunakan



Dari Grafik 4.10 dapat dilihat bahwa dengan menggunakan ciri batang sebagai vektor masukan, jumlah kesalahannya relatif lebih kecil daripada kedua ciri lainnya, sehingga sistem akan melakukan sedikit kesalahan. Selanjutnya dalam pengenalan kode pos penelitian ini, masukan jaringan acuan adalah vektor ciri batang.

Beberapa contoh pengenalan angka yang salah dan ragu-ragu diperlihatkan dalam Gambar 4.3. Tampak bahwa pada pengenalan ragu-ragu, karakter tidak dapat diidentifikasi dengan jelas untuk menyatakan sebuah angka karena mirip dengan angka lain. Sedang pada pengenalan angka salah karakter lebih cocok menyatakan angka lain daripada yang kita tuliskan.



IV.3 Pengenalan Kode Pos

Di bagian ini akan dibahas hasil penerapan kesemua modul di atas terhadap pengenalan kode pos. Jumlah data pengujian 137 citra alamat berkode pos, yang terdiri atas 49 kode pos dalam kotak dan 88 di luar kotak. Hasilnya dapat dilihat pada Tabel 4.2.

Tabel 4.2 Hasil pengenalan kode pos dan angka

	Hasil pengenalan			
	Kode pos yang sukses	Angka benar	Angka salah	Angka meragukan
Dalam kotak	75,51%	93,88%	2,45%	3,67%
Diluar kotak	81,82%	96,36%	2,73%	0,91%
T o t a l	79,56%	95,47%	2,63%	1,90%

Terlihat pada Tabel 4.2 bahwa masih terdapat kesalahan yang belum terdeteksi oleh sistem, artinya angka-angka tersebut masih dibenarkan oleh jaringan acuan.

V. Penutup

V.1 Kesimpulan

Bagian ini merupakan kesimpulan atas hasil penelitian yang telah dilakukan.

1. Hasil pengenalan angka dengan menggunakan gabungan vektor ciri batang dan ciri transisi memberikan hasil yang lebih baik daripada hanya menggunakan salah satu vektor ciri. Hasil pengenalan angka ini didominasi dari penggunaan vektor ciri batang.
2. Jaringan perambatan balik mencapai hasil pengenalan angka tulisan tangan optimal dengan 20 neuron tersembunyi, serta fungsi aktivasi untuk lapisan pertama menggunakan *tansig* dan lapisan kedua menggunakan *purelin*.
3. Sistem deteksi kesalahan dapat didekati dengan menggunakan jaringan acuan kaskade, yaitu dengan membangun sebuah jaringan acuan setelah jaringan utama untuk memutuskan apakah hasil pengenalan jaringan utama benar atau meragukan. Bila hasilnya meragukan maka sistem akan melaporkan sebuah kesalahan.
4. Hasil pengujian keseluruhan, hasil pengenalan benar 95,47%, salah 2,63 % dan ragu-ragu 1,9%. Pengenalan kode pos yang sukses adalah 79,56%.

V.2 Saran

Bagian ini merupakan saran untuk perbaikan penelitian yang telah dilakukan.

1. Untuk mendapatkan hasil pengenalan yang lebih optimal lagi, disarankan untuk menambah jumlah data pelatihan serta dilatih beberapa epoch lagi dan dilatih secara *on-line*, artinya setiap ada data baru jaringan dilatih dengan data tersebut.
2. Disarankan untuk pengamatan beberapa variasi jumlah lapisan tersembunyi dan fungsi aktivasi sehingga benar-benar diperoleh jaringan yang lebih optimal.
3. Diperlukan penelitian dengan metode lain untuk menerapkan sistem deteksi kesalahan, sehingga sistem hanya memiliki hasil pengenalan BENAR dan RAGU-RAGU.
4. Sistem dapat dikembangkan lagi dengan tidak hanya mengenali angka kode pos tetapi dapat pula mengenali karakter lain, dengan melatih beberapa karakter lain tersebut dan menambah neuron keluaran jaringan.
5. Penelitian hendaknya ditingkatkan lagi dengan berbagai variasi warna latar belakang amplop surat dan menggunakan kamera sebagai pengganti pemayar.
6. Program dapat dikembangkan lagi dengan bahasa pemrograman yang lain.

VI. Ringkasan

VI.1 Latar Belakang

Pembacaan karakter dengan komputer dikenal sebagai *Optical Character Recognition* (OCR) adalah topik yang telah diteliti selama beberapa tahun. Sebagian besar solusi untuk permasalahan ini mendasarkan pada proses segmentasi masukan menjadi karakter-karakter. Hasil segmentasi ini yang secara detail akan dikenali. Sistem OCR telah dikomersialkan secara luas untuk pemrosesan dokumen-dokumen hasil cetakan. Namun teknik-teknik untuk membaca tulisan tangan belum sukses diaplikasikan. Permasalahan yang dihadapi untuk pengenalan tulisan tangan sangat kompleks, seperti bervariasinya model tulisan tangan, pena untuk menulis, dan lain lain.

Interpretasi alamat tulisan tangan oleh sistem komputer telah dilakukan dan terus dikembangkan di kantor pos Amerika Serikat. Pengenalan kata tulisan tangan dikarakterisasi oleh kecepatan data yang tinggi, jumlah data yang besar, mengurangi kemungkinan masukan yang salah dan kebutuhan untuk tanggapan waktu nyata (*real time response*).

Penelitian ini bermaksud menerapkan jaringan neural untuk pengenalan angka pada kode pos alamat tulisan tangan sebagai penentu kota tujuan surat. Dengan pengenalan kota tujuan pada alamat tulisan tangan oleh sistem komputer diharapkan bisa mengalokasikan surat-surat di Indonesia secara otomatis, khususnya, dan untuk dikembangkan lagi dalam pengenalan kata tulisan tangan lainnya.

Pada penelitian ini citra masukan berupa seluruh blok alamat tulisan tangan, dengan kode pos didalam dan diluar kotak, yang berbeda dengan peneliti sebelumnya. Serta ditambah sistem deteksi kesalahan dengan menggunakan jaringan neural kaskade. Basis data yang dibuat berdasarkan angka tulisan tangan Indonesia.

VI.2 Landasan Teori

Pendeteksian dan pengenalan nomor jalan dalam suatu alamat tulisan tangan berdasar sistem logika kabur, telah dikembangkan oleh Paul Gader dkk, 1995. Paul Gader menerapkannya di Kantor Pos Amerika Serikat, karena 15 % alamat ditulis tangan. Pada penelitian ini menerapkan pengolahan citra yaitu segmentasi citra tulisan tangan ke karakter-karakter, pengambilan ciri dan modul pengenalan. Dua tipe vektor ciri dipakai sebagai masukan jaringan neural, vektor ciri transisi dan vektor ciri batang. Jaringan neural dilatih menggunakan perambatan balik dan menggunakan keluaran terkode kelas. Terdapat kelas bernama 'sampah' untuk menerangkan segmen yang tak menyatakan citra karakter, seperti multiple karakter atau potongan karakter. Jaringan umpan maju banyak lapis dapat melakukan klasifikasi pola dengan sangat baik. Sebagai contoh jaringan ini merupakan alat yang baik untuk pengenalan karakter. Namun, kesulitan menggunakan jaringan ini, yaitu adanya beberapa parameter, seperti jumlah unit tersembunyi, fungsi aktivasi dan kecepatan pembelajaran yang harus dipilih.

Paul Gader, dkk, 1995, juga telah melakukan perbandingan pelatihan jaringan neural menggunakan himpunan keanggotaan tegas dan kabur. Dari

perbandingan ini diperoleh jaringan neural yang dilatih dengan himpunan keanggotaan tegas hasilnya baik untuk pengenalan karakter, sedangkan jaringan neural yang dilatih dengan himpunan keanggotaan kabur baik untuk pengenalan kata.

Dalam penelitian yang dilakukan Paul Gader dkk. tersebut, untuk mengenali nomor jalan dalam alamat tulisan tangan, digunakan dua vektor ciri transisi dan batang citra karakter sebagai masukan jaringan neuralnya, dan menggunakan logika kabur untuk mengenali daerah numeris suatu alamat tulisan tangan.

Perambatan-Balik (*Back-Propagation*)

Metode pelatihan perambatan-balik secara sederhana adalah metode *gradient descent* untuk meminimalkan total galat kuadrat keluaran. Aplikasi yang memakai jaringan ini untuk masalah yang melibatkan pemetaan sekumpulan masukan terhadap sekumpulan target keluaran; jadi masuk kategori jaringan dengan pelatihan terbimbing.

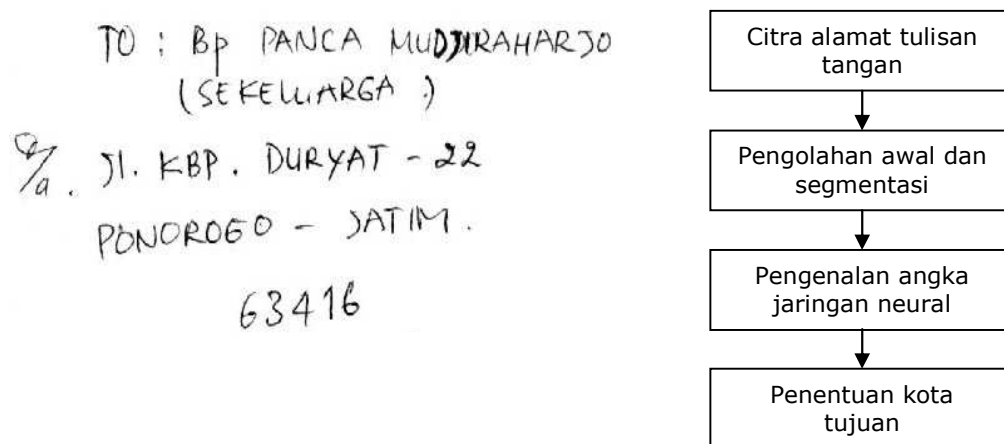
Tujuan pelatihan jaringan ini adalah mendapatkan keseimbangan antara kemampuan tanggapan yang benar terhadap pola masukan yang dipakai untuk pelatihan jaringan (*memorization*) dan kemampuan memberikan tanggapan yang layak untuk masukan yang sejenis namun tidak identis dengan yang dipakai pada pelatihan (*generalization*).

Pelatihan jaringan perambatan balik melibatkan tiga tahap. Umpan-maju pelatihan pola masukan, komputasi dan perambatan-balik galat, serta perubahan bobot. Setelah pelatihan, aplikasi jaringan hanya melibatkan

tahap komputasi umpan-maju. Walau pun pelatihan jaringan lambat, jaringan yang telah dilatih dapat menghasilkan keluaran dengan sangat cepat. Beragam variasi perambatan-balik telah banyak dikembangkan untuk meningkatkan kecepatan proses pelatihan.

Aplikasi dengan jaringan lapisan tunggal dapat dilakukan dengan pemetaan yang terbatas, sehingga jaringan lapis-jamak dengan satu atau lebih lapisan tersembunyi dapat diaplikasikan untuk pemetaan kontinyu dengan keakuratan tertentu. Arsitektur, algoritma, beberapa fungsi aktivasi serta parameter lainnya secara detail dijelaskan oleh Fausett, 1994, halaman 289 - 330.

Pengenalan Kode Pos



Gambar 6.1 Diagram blok pengenalan kode pos

Penelitian ini hanya bagian dari semua modul yang diperlukan untuk pengenalan keseluruhan sistem interpretasi alamat. Penelitian ini terfokus pada modul pengenalan kode pos alamat tulisan tangan. Diagram sistem pengenalan kode pos dengan contoh citra alamat tulisan tangan tampak

pada Gambar 6.1. Keempat submodul yang berbeda pada sistem pengenalan kode pos akan diuraikan pada bagian berikut.

Pencarian Baris Akhir

Kode pos umumnya terletak di baris terakhir alamat, sehingga pertama kali perlu pencarian baris akhir. Pencarian baris akhir dibentuk dengan menganalisis histogram latardepan. Pemisahan baris terjadi pada global minimum diantara dua global maksimum histogram. Ini karena pada global minimum tersebut terjadi transisi dari satu baris ke baris berikutnya. Sedang bukit histogram menandakan adanya coretan atau tulisan di baris tersebut. Untuk memudahkan pencarian global-global minimum ini, pemisahan baris dilakukan dengan menganalisis distribusi histogram latardepan. Suatu citra alamat, histogram latardepan, dan distribusi histogram diperlihatkan pada Gambar 6.2.

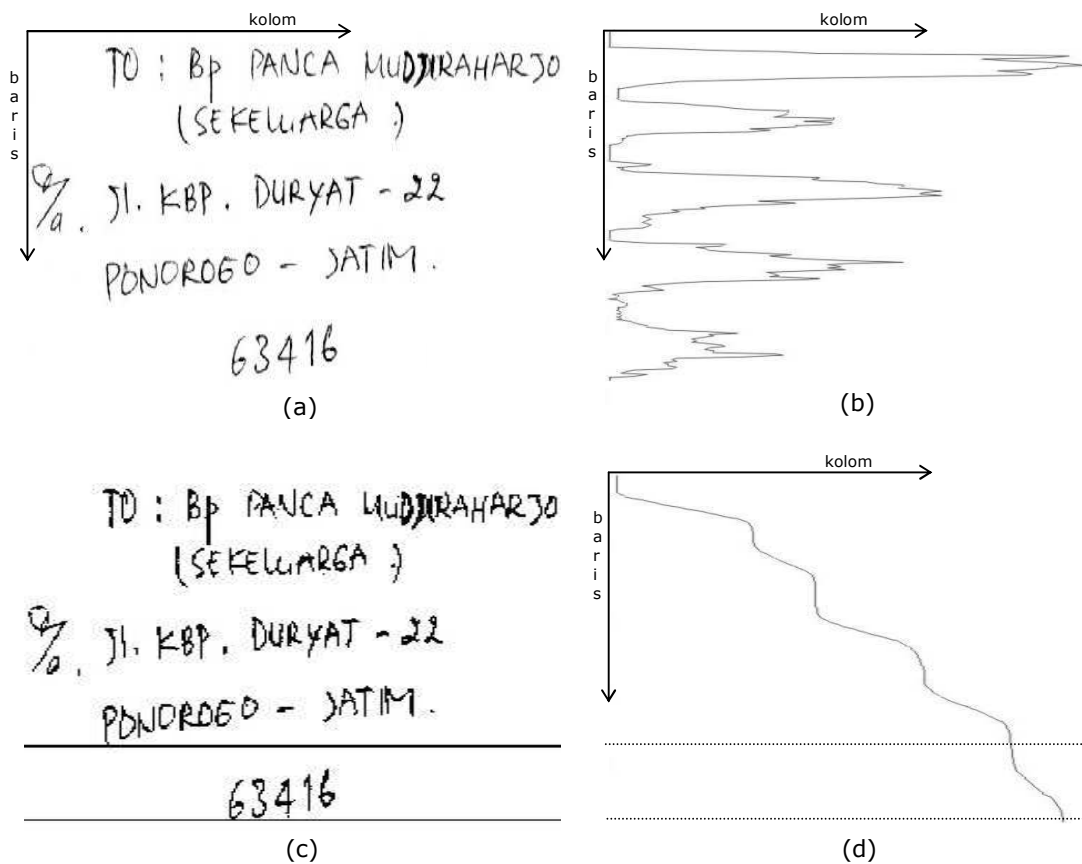
Untuk citra dengan m baris dan n kolom, histogram latar depan diperoleh:

$$JmlPxl(k) = \sum_{b=1}^m Pxl(k,b) \quad k = 1, \dots, n \quad (1)$$

dengan distribusi histogram:

$$DistJmlPxl(k) = \sum_{k=1}^n JmlPxl(k) \quad (2)$$

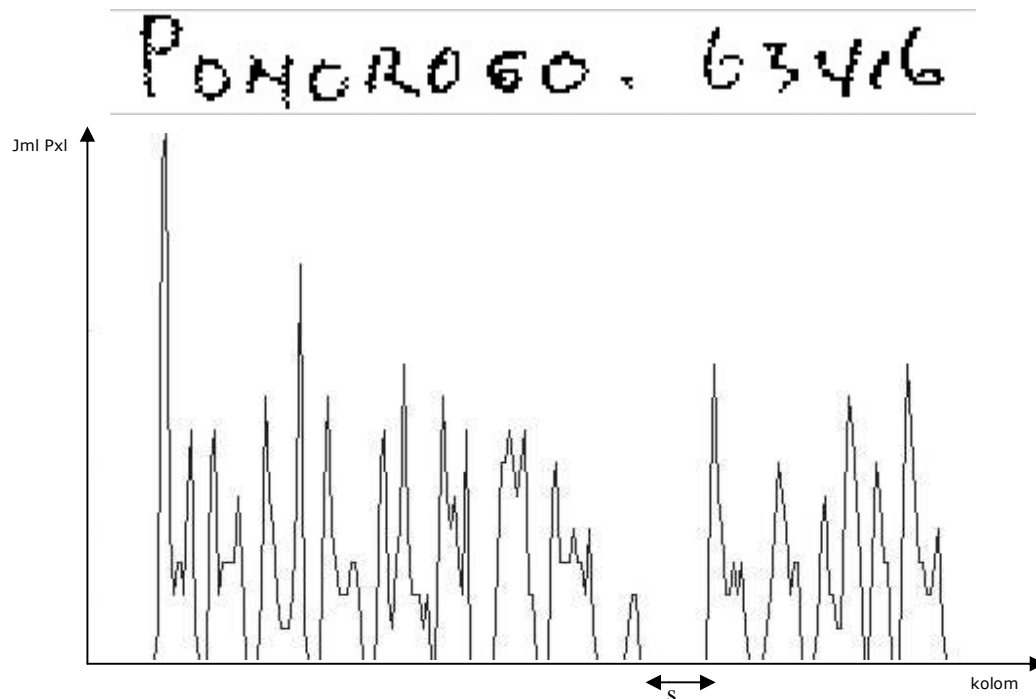
Baris terbentuk pada perubahan kemiringan datar ke menanjak, karena pada perubahan tersebut terletak global minimum di antara global maksimum.



Gambar 6.2 (a) citra alamat asal (b) histogram latar depan
(c) hasil pencarian baris akhir (d) distribusi histogram

Segmentasi Baris ke Blok Kode Pos

Kode pos alamat umumnya terletak di baris terakhir atau di bagian dari baris terakhir. Maka, untuk mengurangi komputasi saat pengenalan karakter perlu pembatasan kata. Segmentasi baris ke blok kode pos dilakukan dengan menganalisis histogram latar depan, seperti tampak pada Gambar 6.3. Spasi blok kode pos dalam suatu baris umumnya ditandai dengan histogram bernilai nol pada rentang jarak tertentu s .



Gambar 6.3 Analisa jarak kata menggunakan histogram latar depan

Penghapusan Garis Kotak Kode Pos

Untuk kode pos dalam kotak, sebelum proses segmentasi blok kode pos ke karakter angka, garis-garis kotak kode pos perlu dihapus terlebih dahulu untuk menghindari kekeliruan pengenalan citra karakter. Penghapusan garis ini menggunakan operasi *opening* morfologi citra biner dengan batang horisontal dan vertikal, yang diuraikan dalam algoritma berikut:

I citra biner

H elemen struktur batang horisontal yang lebarnya sama dengan lebar citra,

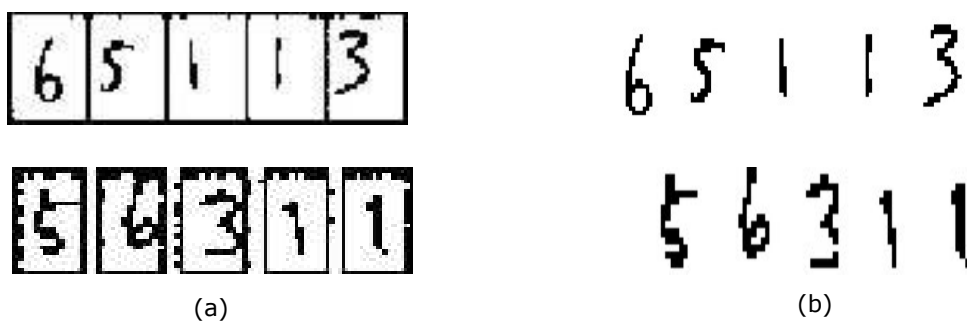
V elemen struktur batang vertikal yang tingginya sama dengan tinggi citra,

$M_H = I \circ H$ hasil opening I dengan H

$M_V = I \circ V$ hasil opening I dengan V

$R = I - M_H - M_V$ hasil citra biner

Penghapusan garis diperlihatkan dalam Gambar 6.4.



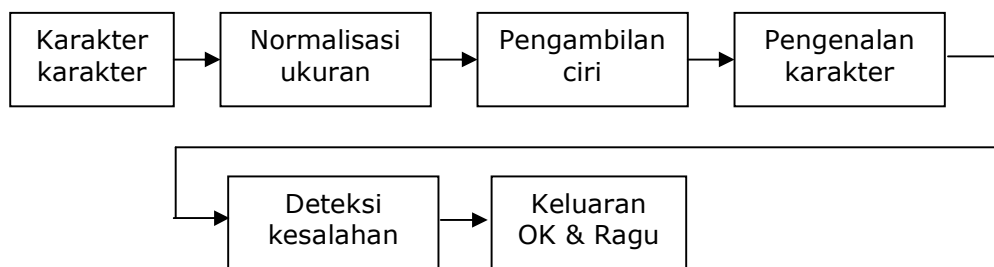
Gambar 6.4 Penghapusan garis kotak pos (a) citra asli (b) citra bersih

Segmentasi Blok Kode Pos ke Karakter

Penulisan angka umumnya terpisah antara angka yang satu dengan lainnya, sehingga proses segmentasi blok kode pos ke karakter menggunakan dua pendekatan. Pendekatan pertama dengan menghitung jumlah spasi antar karakter dalam suatu blok kode pos, untuk lima angka akan terdapat empat spasi. Pendekatan kedua dengan menghitung jumlah coretan dalam blok tersebut, idealnya untuk lima angka akan terdapat lima coretan.

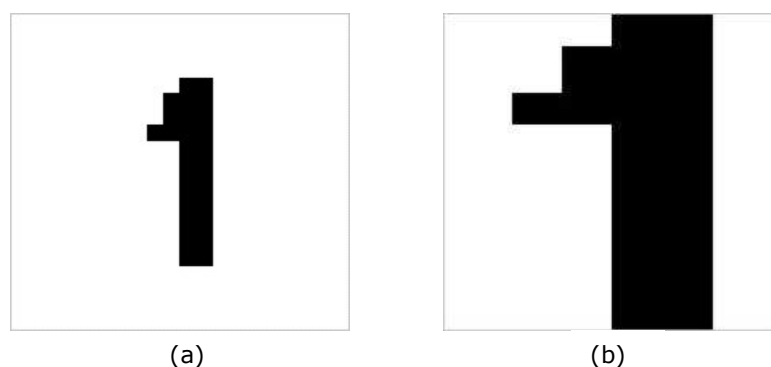
Pengenalan Angka

Pengenalan angka disini melibatkan segmentasi kata ke karakter individual yang diumpankan ke pengenal karakter (*character recognition*). Pengenal karakter yang digunakan adalah jaringan perambatan-balik (*backpropagation*). Diagram pengenalan karakter diperlihatkan pada Gambar 6.5. Karakter hasil segmentasi sebelum diumpankan ke pengenal karakter dinormalisasi ukurannya, kemudian diambil ciri batangnya untuk membentuk vektor masukan.



Gambar 6.5 Diagram blok pengenalan angka

Normalisasi Ukuran Karakter



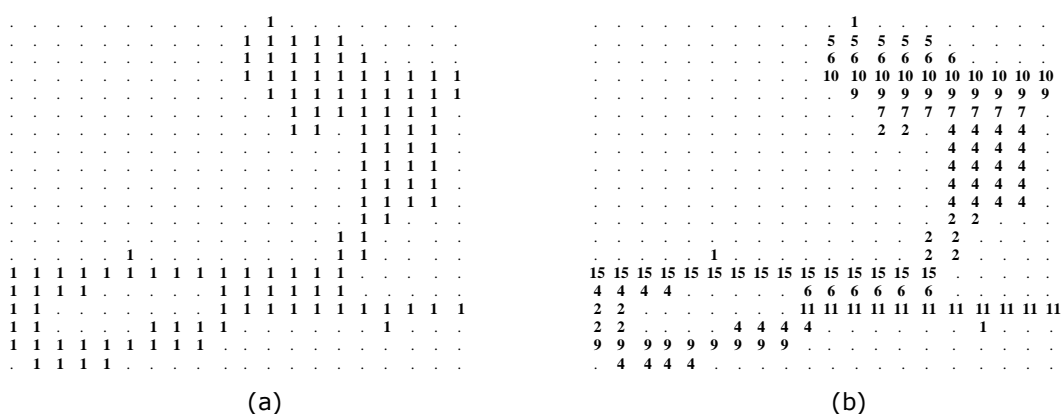
Gambar 6.6 Normalisasi citra karakter (a) Citra karakter asli
(b) citra karakter ternormalisasi

Normalisasi ukuran karakter dilakukan karena tidak samanya ukuran karakter penulis yang satu dengan lainnya. Ukuran karakter memegang peranan penting dalam pengambilan ciri batang. Contoh karakter asli dan hasil normalisasinya ditunjukkan dalam Gambar 6.6.

Ciri Batang

Ciri batang dihitung pada citra biner karakter terpisah. Awalnya, delapan citra ciri dibentuk. Masing-masing citra ciri berhubungan dengan salah satu arah: timur, timur laut, utara dan barat laut, baik itu latar depan atau latar belakang. Masing-masing citra ciri memiliki nilai bilangan bulat pada masing-masing lokasi yang menyatakan panjang batang terpanjang yang menempati titik itu dalam arah tersebut. Algoritma pengambilan ciri batang dijelaskan dalam Gader, dkk, 1995, hal. 358.

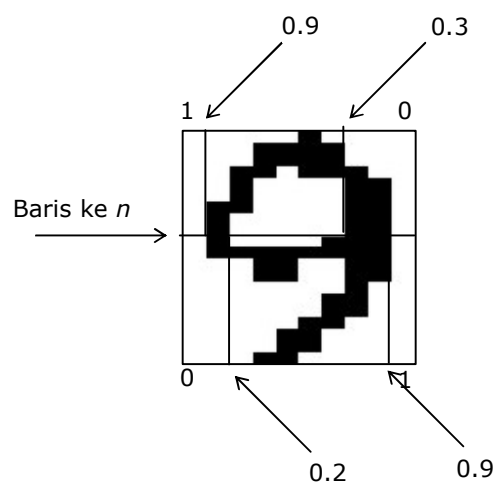
Pada penelitian ini dengan citra berukuran 20 x 20 piksel diperoleh vektor ciri batang 168 elemen. Contoh ciri batang latar depan pada Gambar 6.7 .



Gambar 6.7 Citra karakter dan ciri batang latar depan (a) citra biner asli
(b) ciri batang arah timur

Ciri Transisi

Ciri transisi ini menghitung lokasi dan jumlah transisi dari latar belakang ke latar depan sepanjang baris horizontal dan kolom vertikal suatu citra kata. Perhitungan transisi dikerjakan dari kiri ke kanan, kanan ke kiri, atas ke bawah, dan bawah ke atas. Karena dimensi yang konstan suatu ciri diperlukan untuk masukan ke jaringan neural. Algoritma pengambilan ciri batang dijelaskan dalam Gader, dkk, 1997, hal. 159-160.

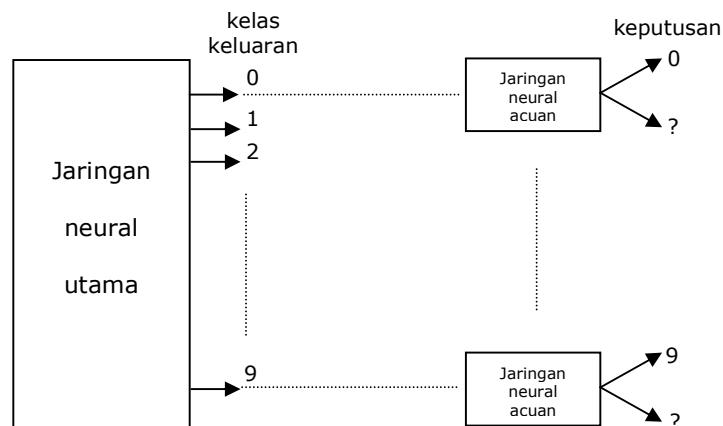


Gambar 6.8 Komputasi ciri transisi suatu baris

Sistem Deteksi Kesalahan

Keluaran jaringan neural umumnya berupa vektor kelas tujuan yang menyatakan besarnya nilai *confidence* untuk menyatakan karakter kelas tujuan. Untuk memutuskan apakah kelas tujuannya benar, maka disini perlu adanya konfirmasi, yang menyatakan bahwa karakter benar atau salah. Sistem deteksi kesalahan, sekali lagi menggunakan jaringan neural dengan algoritma perambatan-balik. Jaringan dilatih dengan masukan acuan dan

masukan selain acuan untuk mendapatkan keluaran benar dan ragu-ragu. Bila keluaran menyatakan ragu-ragu, maka sistem akan melaporkan sebuah kesalahan. Diagram blok sistem deteksi kesalahan ditunjukkan dalam Gambar 6.9.



Gambar 6.9 Diagram blok sistem deteksi kesalahan

VI.3 Cara Penelitian

Bahan Penelitian

Data utama pada penelitian ini adalah sampel angka tulisan tangan yang dipakai untuk pelatihan dan pengujian jaringan perambatan balik. Perancangan sampel mempertimbangkan jumlah yang seimbang atau sama untuk pelatihan maupun pengujian jaringan. Data kedua adalah sampel alamat tulisan tangan yang diperoleh dari surat yang pernah diterima, dari kantor pos terdekat, alamat tulisan tangan rekan mahasiswa dan alamat tulisan tangan sanak famili. Pengumpulan sampel tidak memberi kekangan pada para pengisi lembar isian, mereka diberi kebebasan pemakaian pena dan tipe tulisan. Data terakhir adalah basis data kode pos.

Alat yang Digunakan

Peralatan bantu yang dipakai pada penelitian ini:

- Komputer dengan spesifikasi: Pentium 233 MMX, RAM 32 MHz, *harddisk* satu giga byte.
- Pemayar.
- Program bantu MATLAB 5.3 beserta *toolbox neural network*, dan *image processing*, untuk pemrosesan sampel angka tulisan tangan dan pelatihan serta pengujian jaringan neural perambatan balik.
- Program *Corel Photo-paint 8*, untuk memotong-motong basis data angka tulisan tangan pada formulir isian.

Perancangan Arsitektur Jaringan Utama

Masukan jaringan perambatan-balik adalah vektor ciri batang dan transisi citra karakter dan keluarannya adalah angka yang dimaksud, sehingga arsitektur jaringan pertama yang akan dibangun memiliki 168 neuron masukan dan 10 neuron keluaran, dan komponen yang melibatkan neuron pada lapisan tersembunyi yang jumlahnya akan dicari secara optimal, sedang arsitektur jaringan kedua memiliki 36 neuron masukan dan 10 neuron keluaran.

Umumnya untuk pengenalan karakter tulisan tangan memakai satu lapisan tersembunyi, sehingga disini juga menggunakan satu lapisan tersembunyi.

Seperti disinggung dalam tinjauan pustaka, yaitu kesulitan dalam menerapkan jaringan neural adalah tidak adanya rumusan baku tentang

jumlah lapisan tersembunyi, fungsi aktivasi dan kecepatan pembelajaran untuk mendapatkan jaringan yang optimal, maka sebelum menetapkan jaringan utama diperlukan uji coba parameter-parameter tersebut sehingga diperoleh jaringan yang optimal.

Perancangan Sistem Deteksi Kesalahan

Sistem deteksi kesalahan disini menggunakan jaringan neural yang dilatih dengan karakter acuan dan karakter selain acuan untuk mendapatkan dua kelas keluaran yaitu keluaran BENAR dan RAGU-RAGU. Jaringan neural disini juga dilatih dengan algoritma perambatan-balik. Untuk jaringan acuan NOL, karakter acuannya adalah 0 dan karakter selain acuannya adalah 1,2, ...
 9. Jaringan acuan SATU, karakter acuannya adalah 1 dan karakter selain acuannya adalah 0,2,3,... 9, dan seterusnya.

VI.4 Hasil Penelitian dan Pembahasan

Pencarian Baris Akhir

Kepada	Kepada
Yth. Albertus Andrejonte	Budi Setiyono
Jl. Batu jajar 1/32	Gembong Rt 03 kw 01 No. 8
_____	Bojongsari
Malang - 65113	Purbalingga - 53362
_____	_____

Gambar 6.10 Hasil pengujian pencarian baris akhir

Dari hasil penelitian ditentukan duakali pencarian lembah histogram, pertama lembah sementara dengan mencari perubahan kemiringan grafik distribusi hitogram dengan perhitungan:

$$\text{distribusi}_{i+10} - \text{distribusi}_i < \text{distribusi}_i - \text{distribusi}_{i-11}$$

dengan distribusi_i adalah distribusi histogram baris ke i . Kemudian lembah yang sebenarnya adalah global minimum diantara dua lembah sementara.

Segmentasi Baris ke Blok Kode Pos

Pada penulisan alamat surat baku, kode pos terletak pada baris terakhir dan blok kata terakhir. Seperti disinggung dalam landasan teori, untuk pencarian blok kata terakhir dilakukan dengan menganalisis histogram latar depan dan mencari spasi antar kata. Untuk menutupi spasi antar karakter dalam blok kode pos tersebut, dilakukan proses dilasi pada citra baris terakhir:

```
StructuringElemen = ones(1,5);
```

```
DataDilasi = dilate (BarisAkhir,StructuringElemen);
```



Gambar 6.11 Hasil dilasi citra

Segmentasi Blok Kode Pos ke Karakter

Segmentasi blok kode pos ke karakter dilakukan dengan menghitung jumlah karakter dan menghitung jumlah spasi antar karakter. Untuk menghitung jumlah karakter menggunakan sintaks MATLAB:

```
LabelCitra = bwlabel(Citra);  
JumlahKarakter = max(max(LabelCitra));
```

Normalisasi Ukuran Karakter

Normalisasi ukuran karakter dilakukan karena tidak samanya ukuran karakter penulis yang satu dengan lainnya. Ukuran karakter memegang peranan penting dalam pengambilan ciri batang. Pada penelitian ini semua karakter dinormalisasi 20x20 piksel. Normalisasi dilakukan dengan fungsi Matlab:

```
Normalisasi = imresize(CitraAsli,[20 20],'nearest')
```

Pengenalan Angka

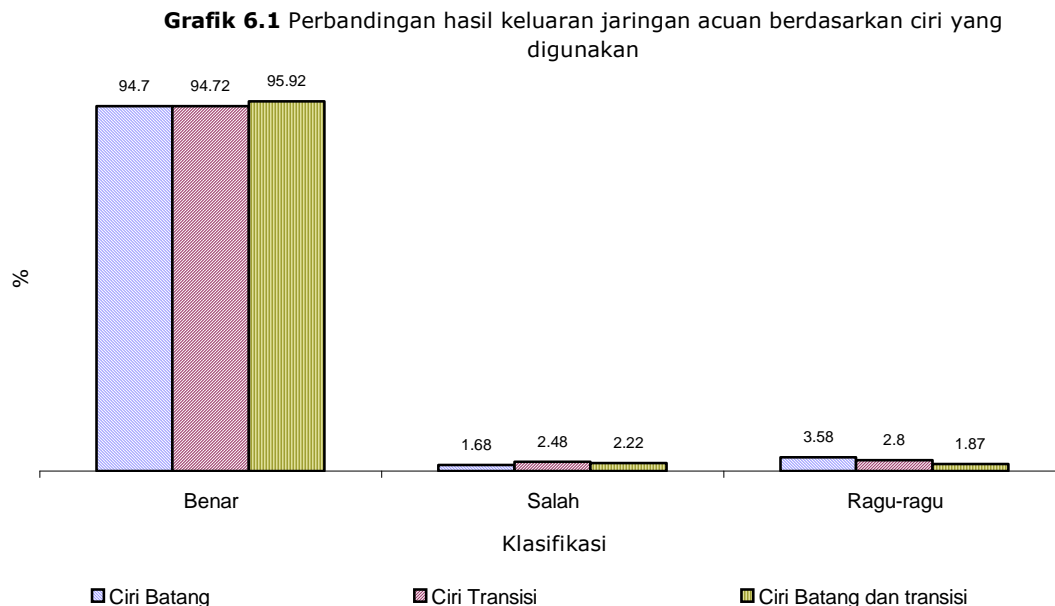
Pada penelitian ini untuk pengenalan angka yang optimal dengan 20 neuron tersembunyi dan fungsi aktivasi untuk lapisan pertama menggunakan *tansig* dan lapisan kedua menggunakan *purelin*.

Untuk mencapai hasil pengenalan yang optimal jaringan dilatih hingga mencapai konvergensi, pada penelitian ini dengan data pelatihan 200 data jaringan dilatih hingga 10000 epoch, dilanjutkan dengan 500 data dilatih 250 epoch.

Sistem Deteksi Kesalahan

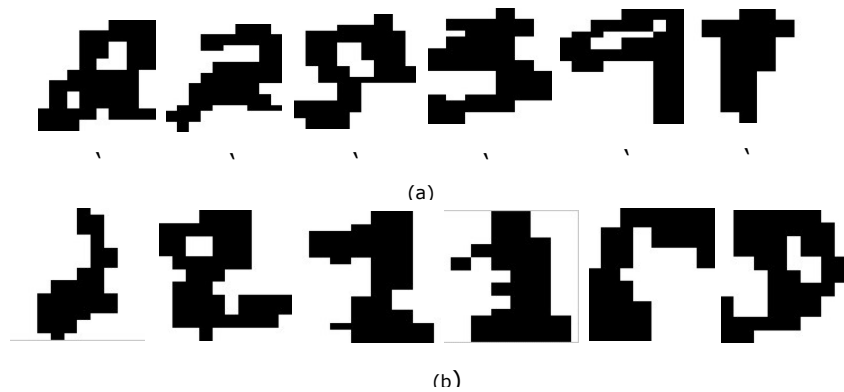
Seperti dibahas dalam landasan teori, sistem deteksi kesalahan dilakukan dengan membangun jaringan acuan setelah jaringan utama. Hasil pengenalan oleh jaringan utama, citra karakternya akan diuji lagi dengan jaringan acuan untuk diputuskan apakah karakter tersebut benar atau ragu-ragu. Bila hasil pengenalan oleh jaringan acuan ragu-ragu, maka sistem akan melaporkan sebuah kesalahan.

Dari Grafik 6.1 dapat dilihat bahwa dengan menggunakan ciri batang sebagai vektor masukan, jumlah kesalahannya relatif lebih kecil daripada kedua ciri lainnya, sehingga sistem akan melakukan sedikit kesalahan. Selanjutnya dalam pengenalan kode pos penelitian ini, masukan jaringan acuan adalah vektor ciri batang.



Beberapa contoh pengenalan angka yang salah dan ragu-ragu diperlihatkan dalam Gambar 6.12. Tampak bahwa pada pengenalan ragu-ragu, karakter tidak dapat diidentifikasi dengan jelas untuk menyatakan

sebuah angka karena mirip dengan angka lain. Sedang pada pengenalan angka salah karakter lebih cocok menyatakan angka lain daripada yang kita tuliskan.



Gambar 6.12 Klasifikasi pengenalan angka (a) salah (b) ragu-ragu

Pengenalan Kode Pos

Di bagian ini akan dibahas hasil penerapan kesemua modul di atas terhadap pengenalan kode pos. Jumlah data pengujian 137 citra alamat berkode pos, yang terdiri atas 49 kode pos dalam kotak dan 88 di luar kotak.

Tabel 6.2 Hasil pengenalan kode pos dan angka

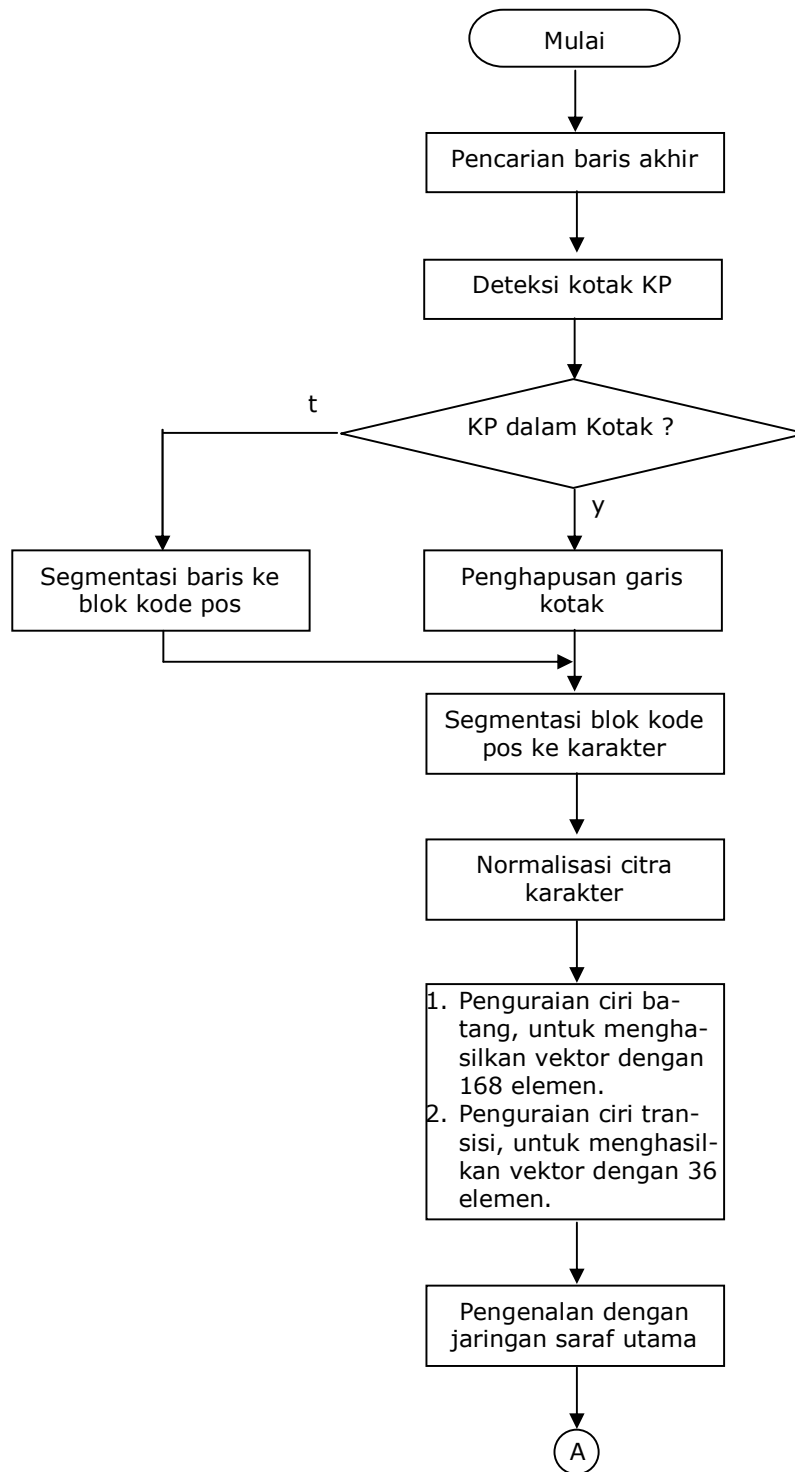
	Hasil pengenalan			
	Kode pos yang sukses	Angka benar	Angka salah	Angka meragukan
Dalam kotak	75,51%	93,88%	2,45%	3,67%
Diluar kotak	81,82%	96,36%	2,73%	0,91%
T o t a l	79,56%	95,47%	2,63%	1,90%

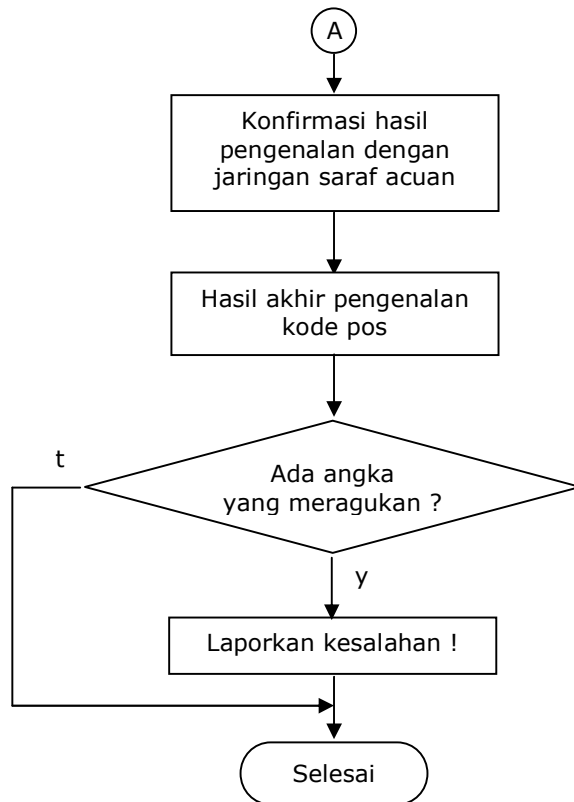
DAFTAR PUSTAKA

- Bozinovic R.M., Srihari S.N., 1989, "*Off-Line Cursive Script Word Recognition*", IEEE on Pattern Analysis and Machine Intelligence, Vol. 11. NO. 1, pp. 68-83.
- Chen, M-Y., Kundu, A., Srihari, S.N., 1995, "*Variable Duration Hidden Markov Model and Morphological Segmentation for Handwritten Word Recognition*", IEEE Transactions on Image Processing, Vol. 4. NO. 12, pp. 1675-1687.
- Chiang, J.H., Gader, P.D., 1997, "*Hybrid Fuzzy-Neural Systems in Handwritten Word Recognition*", IEEE Transactions on Fuzzy Systems, Vol. 5. NO.4, pp. 497-509.
- Demuth, H., Beale, M., 1998, "*Neural Network Toolbox*", The Math Work. Inc.
- Fausett, L., 1994, "*Fundamentals of Neural Networks*", Prentice Hall, New Jersey.
- Fontaine T., Shastri L., 1993, "*A Hybrid System for Handprinted Word Recognition*", Computer and Information Science Department, University of Pennsylvania, Philadelphia.
- Gader, P.D., Keller, J.M., Cai, J., 1995, "*A Fuzzy Logic System for the Detection and Recognition of Handwritten Street Number*", IEEE Transactions on Fuzzy Systems, Vol. 3. NO. 1, pp. 83-94.
- Gader, P.D., Mohamed, M.A., Chiang, J.H., 1995, "*Comparison of Crisp and Fuzzy Character Neural Networks in Handwritten Word recognition*", IEEE Transactions on Fuzzy Systems, Vol. 3. NO. 3, pp. 357-363.
- Gader, P.D., Mohamed, M.A., Chiang, J.H., 1997. "*Handwritten Word Recognition with Character and Inter-Character Neural Networks*", IEEE Transactions on Systems, Man, and Cybernetics, Vol. 27, No. 1, pp. 158-164.
- Guillevic, D., 1995, "*Unconstrained Handwriting Recognition Applied to the Processing of Bank Cheques*", Thesis, Concordia University, Montreal, Quebec, Canada.
- Hanselman, D., Littlefield, B., 1998, "*Mastering MATLAB 5*", Prentice Hall, New Jarsey.
- Jain, A.K., 1995, "*Fundamentals of Digital Image Processing*", Prentice Hall, New Delhi.
- Mohamed, M.A., Gader, P.D., 2000, "*Generalized Hidden Markov Models-Part II: Application to Handwritten Word Recognition*", IEEE Transactions on Fuzzy Systems, Vol. 8. NO.1, pp. 82-94.
- The Math Work. Inc, 1998, "*Image Processing Toolbox*".

LAMPIRAN A

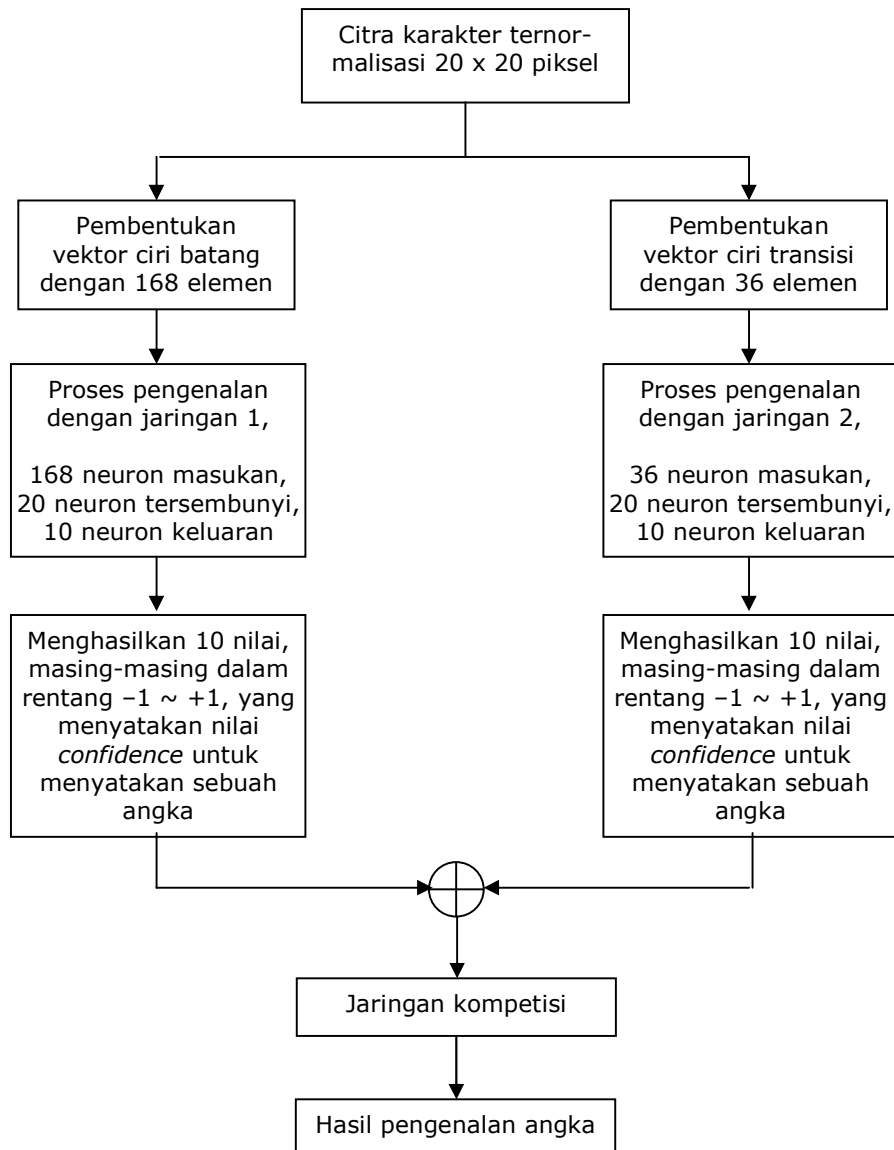
DIAGRAM ALIR PROSES PENGENALAN KODE POS





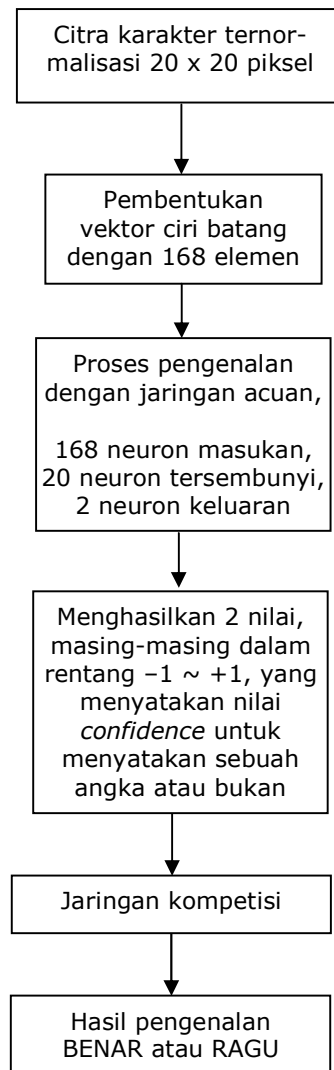
LAMPIRAN B

PROSES PENGENALAN KARAKTER DENGAN JARINGAN UTAMA MENGUNAKAN CIRI BATANG DAN TRANSISI



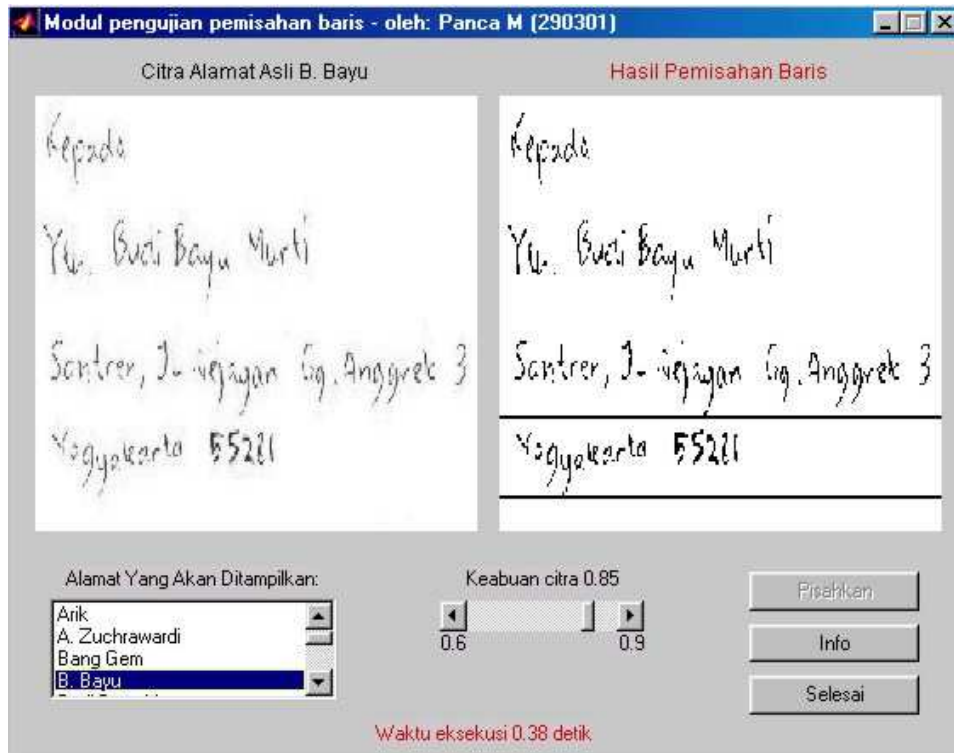
LAMPIRAN C

PROSES PENGENALAN KARAKTER DENGAN JARINGAN ACUAN MENGUNAKAN CIRI BATANG

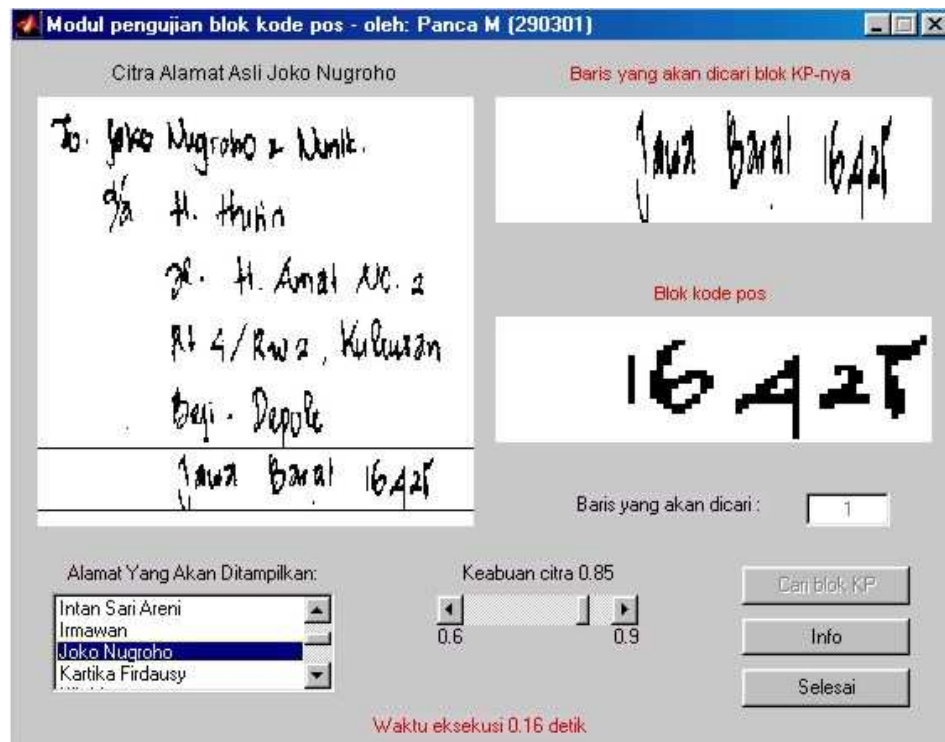


LAMPIRAN D

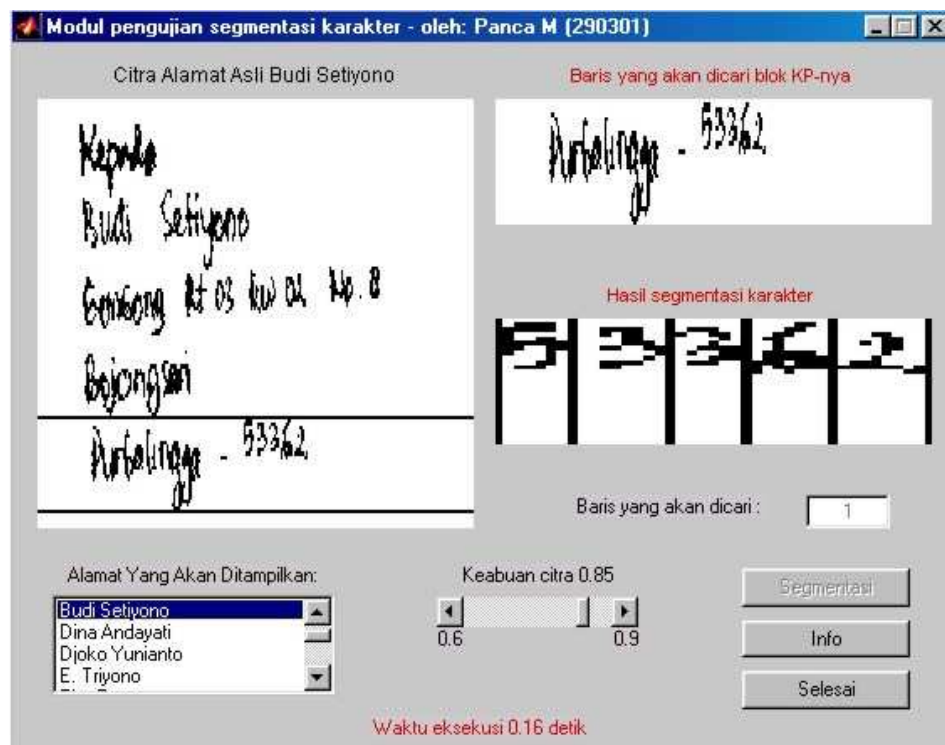
MODUL PENGUJIAN



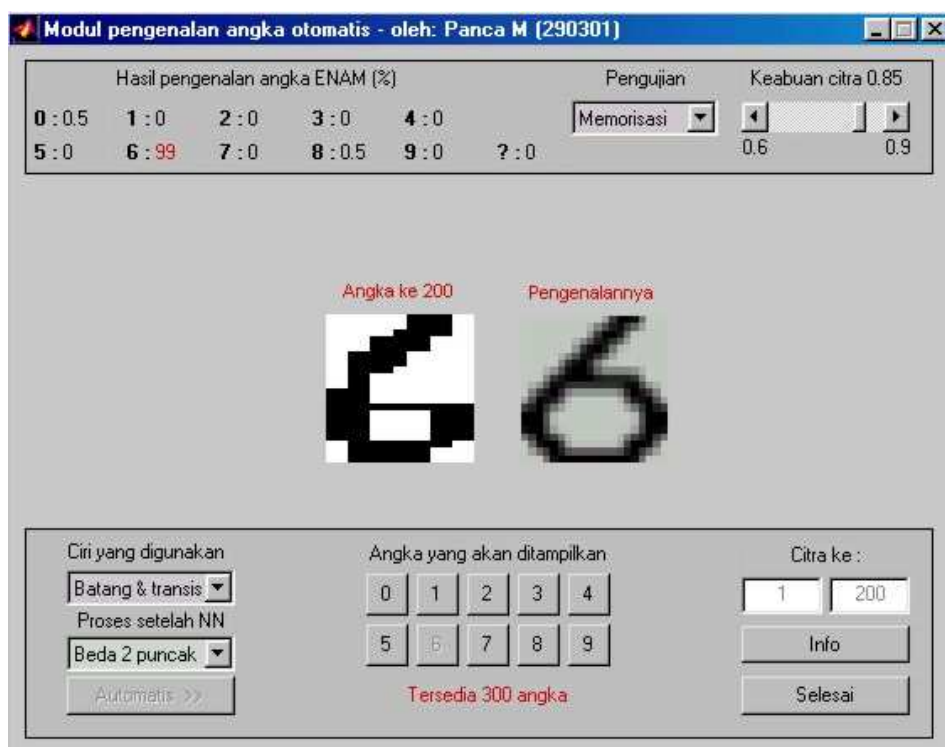
D.1 Modul pengujian pencarian baris akhir



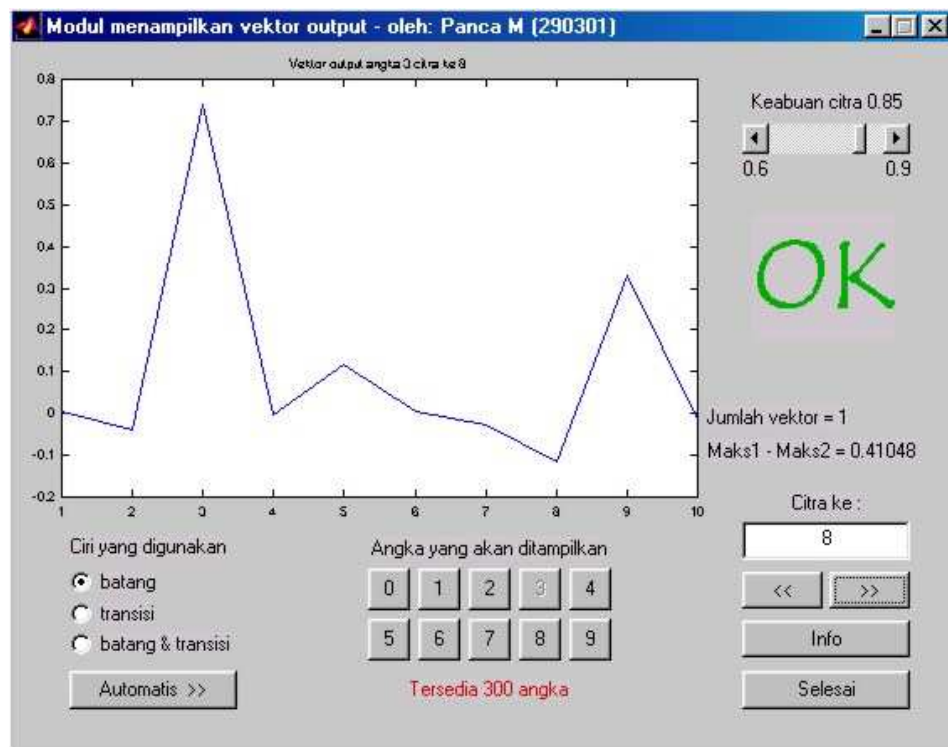
D.2 Modul pengujian pencarian blok kode pos



D.3 Modul pengujian segmentasi karakter



D.4 Modul pengenalan angka



A.5 Modul menampilkan vektor keluaran jaringan

LAMPIRAN E

SIMULASI PENGENALAN KODE POS

Simulasi pengenalan kode pos - oleh Panca M. (290301)

Citra Alamat Asli Suhandoyo

Kode Pos Yang Akan Dikenali

Kepada
gth. Bpk. Suhandoyo
H. Sumber Sari I/40.
Malang - 65145

Keabuan citra 0.85

0.6 0.9

Hasil Pengenalan Kode Pos

65145

Alamat Yang Akan Ditampilkan:

Suhandoyo
Suhartanto
Suharti
Suhartono

Ciri yang digunakan

Batang & Transisi

Proses setelah NN

NN Acuan

Kenali

Info

Selesai

Waktu eksekusi 3.79 detik

Simulasi pengenalan kode pos - oleh Panca M. (290301)

Citra Alamat Asli Yermia I.

Kode Pos Yang Akan Dikenali

Kepada :
yeli : Yermia Intania
waras, Sariharjo, Maglik RT 04/32
Suman, Yogyakarta.

Keabuan citra 0.85

0.6 0.9

Hasil Pengenalan Kode Pos

55581

Alamat Yang Akan Ditampilkan:

Y. Intania
Yanti
Yermia I.
Zainal Abidin

Ciri yang digunakan

Batang & Transisi

Proses setelah NN

NN Acuan

Kenali

Info

Selesai

Waktu eksekusi 3.9 detik