

Aplikasi MATLAB untuk Mengenali Karakter Tulisan Tangan

Ali Mahmudi, Rofila El Maghfiroh, Agung Panji Sasmito

Abstract— Handwriting recognition is one of the very interesting research object in the field of image processing, artificial intelligence and computer vision. This is due to the handwritten characters is varied in every individual. The style, size and orientation of handwriting characters has made every body's is different, hence handwriting recognition is a very interesting research object. Handwriting recognition application has been used in quite many applications, such as reading the bank deposits, reading the postal code in letters, and helping people in managing documents.

This paper presents a handwriting recognition application using Matlab. Matlab toolbox that is used in this research are Image Processing and Neural Network Toolbox.

Keywords: Image Processing, Edge Detection, Neural Networks, Back Propagation Network (BPN).

Abstrak— Pengenalan tulisan tangan adalah salah satu objek penelitian yang sangat menarik di bidang pengolahan citra, kecerdasan buatan dan komputer vision. Hal ini dikarenakan karakter tulisan tangan cukup bervariasi pada tiap-tiap individu. Gaya tulisan, ukuran dan orientasi tulisan tangan menjadikan pengenalan tulisan tangan menjadi objek penelitian yang sangat menarik. Aplikasi pengenalan tulisan tangan dapat dipergunakan pada banyak hal, antara lain untuk membaca giro bank, membaca kode pos surat, dan membantu manusia dalam menata dokumen. Makalah ini menyajikan pengenalan tulisan tangan dengan menggunakan aplikasi Matlab. *Toolbox* Matlab yang dipergunakan antara lain *Image Processing* dan *Neural Network Toolbox*.

Kata kunci: Pengolahan citra, *Image Processing*, Deteksi Tepi, Jaringan Syaraf Tiruan, *Back Propagation Network* (BPN).

Manuscript received January 12, 2007. This work was supported in part by Teknik Informatika, Institut Teknologi Nasional Malang.

Ali Mahmudi is with Teknik Informatika, Institut Teknologi Nasional Malang, Indonesia; email amahmudi@hotmail.com

Rofila El Maghfiroh, was Teknik Informatika, Institut Teknologi Nasional Malang, Indonesia; e-mail: rofila.elma@gmail.com.

Agung Panji Sasmito is is the Teknik Informatika, Institut Teknologi Nasional Malang, Indonesia; e-mail: agungpanji@yahoo.com.

I. PENDAHULUAN

Bila seseorang membaca sebuah huruf, sebut saja X, maka manusia akan dengan sangat mudah mengenali huruf X tersebut. Jika pada tulisan huruf X tersebut, ada beberapa bagian yang tidak lurus, maka manusia akan mencoba mengenali dan memperkirakan huruf tersebut, dan kemungkinan bisa mengenalinya dengan benar. Hal ini dikarenakan akal manusia memiliki teknologi pengenalan pola yang sangat baik.

Aplikasi komputer untuk pengenalan tulisan tangan (*Handwriting recognition*) adalah kemampuan komputer untuk menerima dan menafsirkan *input* tulisan tangan yang dapat dimengerti dari sumber seperti teks tertulis, foto, layar sentuh dan perangkat lainnya. Gambar dari teks tertulis dapat digunakan secara luring dari selembar kertas oleh pemindai optik ([*optical character recognition*](#) atau OCR). Selain itu, gerakan ujung pena dapat dimengerti secara daring, misalnya dengan menggunakan permukaan layar komputer berbasis pena. Pengenalan tulis tangan memerlukan rekognisi karakter optik. Namun, sistem pengenalan tulisan tangan yang lengkap juga menangani format, melakukan segmentasi yang tepat kedalam karakter dan menemukan kata yang paling masuk akal.

Pengenalan tulisan tangan biasa dikenal dengan [*optical character recognition*](#) (OCR). OCR memproses gambar dengan menggunakan beberapa langkah, yakni segmentasi, ekstraksi fitur dan klasifikasi. OCR membandingkan antara gambar dengan pola tulisan yang disimpan di dalam system. Hal ini tentu menimbulkan ketidakakuratan dikarenakan keterbatasan pola tulisan yang disimpan. Bagaimanapun juga, pengenalan tulisan tangan adalah hal yang mungkin dilakukan oleh komputer, sedangkan hasil pemrosesan sangat tergantung dari kualitas gambar masukan[1][2][3].

Untuk mengenali karakter tulisan tangan, langkah pertama adalah memperoleh gambar untuk diproses. Selanjutnya, *Image Processing toolbox* dipergunakan untuk mengekstrak karakter dan kemudian menggunakan *Neural Network toolbox* untuk melatih dataset yang cocok. Setelah pelatihan jaringan, pengujian dilakukan dan kurva kinerja yang dihasilkan bersama dengan karakter individu yang diperlukan[4][5][6][7][8].

II. GRAYSCALE DAN THRESHOLDING

Setelah memperoleh gambar untuk diproses, langkah berikutnya adalah pengolahan gambar dengan menggunakan *Image Processing Toolbox*. Gambar masukan dikonversi menjadi grayscale, dengan tujuan menghilangkan warna dan informasi yang berlebih, tetapi tetap mempertahankan pencahayaan.

$$\text{Grayscale} = 0,299 \text{ Red} + 0,587 \text{ Green} + 0,114 \text{ Blue} \quad (1)$$

Gambar grayscale diproses lebih lanjut menjadi citra biner. Pada proses ini, semua piksel dalam gambar masukan dengan pencahayaan lebih besar dari *threshold* diberi nilai 1 (putih) dan menggantikan semua piksel lainnya dengan nilai 0 (hitam). Ini mengikuti penggunaan deteksi tepi untuk menemukan tepi citra.

$$f_b(i, j) = \begin{cases} 1 & \text{jika } f(i, j) \geq T \\ 0 & \text{jika } f(i, j) < T \end{cases} \quad (2)$$

Setelah mendeteksi tepi, perlu untuk menggunakan morfologi untuk melebarkan dan mengisi gambar yang mendefinisikan segmentasi gambar untuk mengisi dengan memilih titik-titik yang diperlukan[7].

III. SEGMENTASI DAN CROPPING

Gambar kemudian dibagi menjadi beberapa bagian yang mana masing-masing bagian memiliki kemiripan atribut. Proses ini disebut dengan segmentasi. Segmentasi bertujuan untuk memecah sebuah citra menjadi beberapa segmen dengan suatu kriteria tertentu[7].

Langkah berikutnya setelah segmentasi adalah *Cropping*. *Cropping* pada pengolahan citra bermakna memotong satu bagian dari citra sehingga diperoleh citra yang diharapkan. Yang diperlukan dalam proses *cropping* suatu citra adalah menentukan titik koordinat dari citra yang akan dicrop dan kemudian menentukan batas dari citra.

$$W' = (X_r - X_l) + 1 \quad (3)$$

$$H' = (Y_b - Y_t) + 1 \quad (4)$$

IV. NORMALISASI

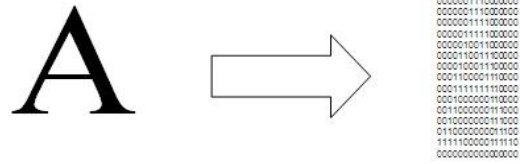
Langkah berikutnya adalah normalisasi[7]. Proses normalisasi adalah mentransformasikan citra ke bentuk citra normal yang sesuai dengan kebutuhan. Setelah normalisasi adalah proses penskalaan atau *scaling*. Proses *scaling* bertujuan agar nilai citra atau fitur berada pada suatu interval tertentu. Penskalaan citra dirumuskan sebagai berikut :

$$x' = S_x \cdot x \quad (5)$$

$$y' = S_y \cdot y \quad (6)$$

V. EKSTRAKSI CIRI

Ekstraksi ciri pada pengolahan citra berarti mengubah nilai-nilai intensitas koordinat piksel yang terdapat dalam citra menjadi susunan kode-kode nilai pada setiap piksel[7][9].



VI. JARINGAN SARAF TIRUAN

Jaringan saraf tiruan bisa dibayangkan seperti otak buatan di dalam cerita-cerita fiksi ilmiah. Otak buatan ini dapat berfikir seperti manusia, dan juga sepandai manusia dalam menyimpulkan sesuatu dari potongan-potongan informasi yang diterima. Khayalan manusia tersebut mendorong para peneliti untuk mewujudkannya. Komputer diusahakan agar bisa berfikir sama seperti cara berfikir manusia. Caranya adalah dengan melakukan peniruan terhadap aktivitas-aktivitas yang terjadi di dalam sebuah jaringan saraf biologis[5][11].

Ketika manusia berfikir, aktivitas-aktivitas yang terjadi adalah aktivitas mengingat, memahami, menyimpan, dan memanggil kembali apa yang pernah dipelajari oleh otak. Sesungguhnya apa yang terjadi di dalam otak manusia jauh lebih rumit dari apa yang telah disebutkan di atas. Para ahli bedah otak sering membicarakan mengenai adanya pengaktifan *neuron*, pembuatan koneksi baru, atau pelatihan kembali pola-pola tingkah laku pada otak manusia. Sayangnya hingga saat ini bagaimana sesungguhnya aktivitas-aktivitas tersebut berlangsung belum ada yang mengetahui dengan pasti. Itulah sebabnya mengapa jaringan saraf tiruan dikatakan hanya mengambil ide dari cara kerja jaringan saraf biologis[5][11].

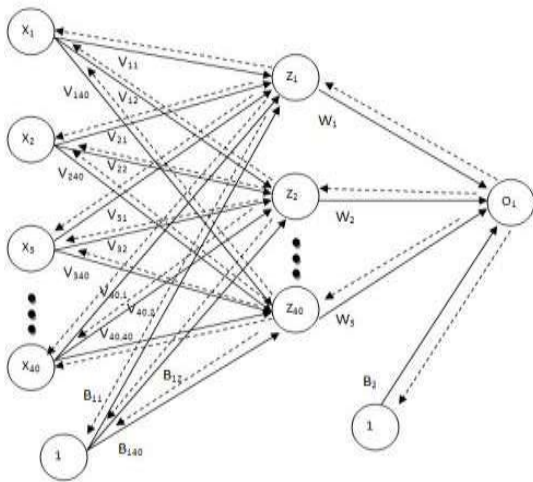
VII. BACKPROPAGATION

Backpropagation (propagasi balik) pertama dikembangkan pada tahun 1986 oleh Rumelhart, Hinton dan Williams untuk menentukan bobot dan digunakan untuk pelatihan *perceptron* multi lapis. Metode *Backpropagation* (propagasibalik) merupakan metode yang sangat baik dalam menangani masalah pengenalan pola-pola kompleks[7][10][11].

Metode ini merupakan metode jaringan saraf tiruan yang populer. Beberapa contoh aplikasi melibatkan metode ini adalah pengompresian data, pendeteksian virus komputer, pengidentifikasian objek, sintesis suara dari teks, dan lain-lain[11].

Istilah *Backpropagation* (propagasi balik) diambil dari cara kerja jaringan ini, yaitu bahwa *gradient error* unit-unit tersembunyi diturunkan dari penyiaran kembali error-error yang diasosiasikan dengan unit-unit *output*. Hal ini karena nilai target untuk unit-unit tersembunyi tidak diberikan. Back Propagation (propagasi balik) adalah metode pembelajaran terawasi (

supervised learning). Metode ini membutuhkan nilai yang sudah ditentukan sebelumnya untuk mendapatkan output yang diinginkan pada proses pembelajaran[11].



Gambar 1. Arsitektur Backpropagation.

Jaringan saraf tiruan *backpropagation*, mempunyai 3 buah lapisan, yaitu :

- Lapisan Masukan (*Input Layer*)
- Lapisan Tersembunyi (*Hidden Layer*)
- Lapisan Keluaran (*Output Layer*)

Arsitektur dari jaringan saraf tiruan ini dapat dilihat pada gambar 1.

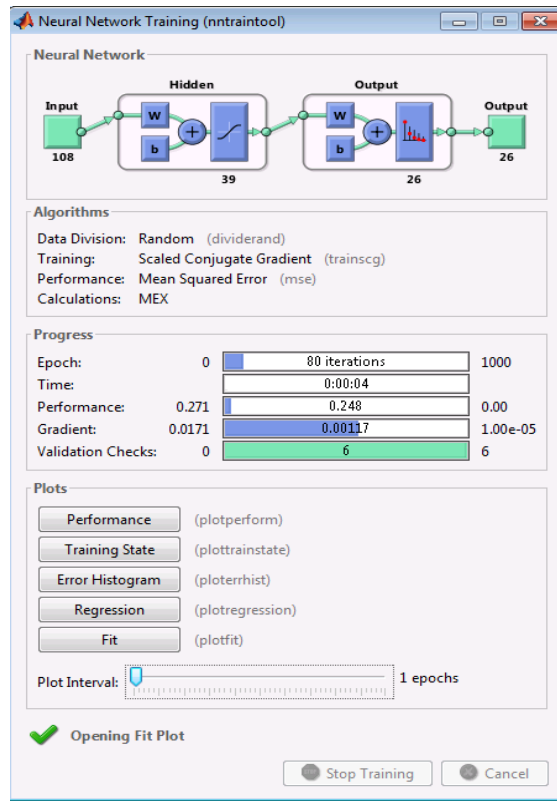
VIII. PELATIHAN

Pembelajaran / *training* pada jaringan saraf tiruan bertujuan untuk mendapatkan keseimbangan nilai antar sistem jaringan pada saat diberikan pola inputan. Keseimbangan antar sistem jaringan tersebut berupa nilai numerik yang disebut dengan bobot (*weight*). Dengan nilai bobot yang sudah tersimpan dalam sistem jaringan, maka jaringan saraf tiruan bisa mengenali/menghafal jenis pola yang sudah diajarkan sebelumnya. Sehingga pada saat tahap pengujian jaringan saraf tiruan dapat mengenali pola inputan yang diujikan kepadanya.

Pada proses pembelajaran *backpropagation* terhadap 3 tahapan, yaitu :

1. pemberian pola masukan (*input*) saat proses pembelajaran.
2. Perhitungan dan proses pelatihan dari nilai *error*.
3. Pengaturan nilai penimbang.

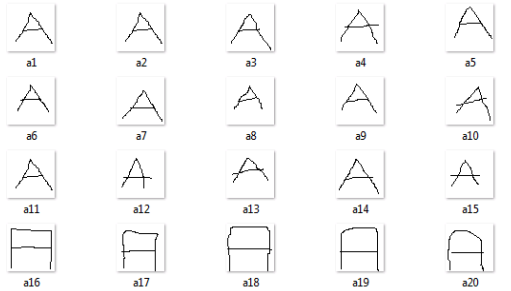
Hasil dari proses pembelajaran adalah berupa nilai bobot interkoneksi yang nilainya disimpan untuk digunakan dalam proses pengenalan pola. Tampilan hasil pelatihan dari Jaringan Syaraf Tiruan (JST) ditunjukkan pada gambar 2.



Gambar 2. Tampilan Hasil Pelatihan JST.

IX. HASIL DAN PENGUJIAN

Pada pengujian aplikasi ini digunakan data training (data referensi) sebanyak 520 buah data, yang terdiri dari 20 jenis variasi untuk setiap karakternya. Masing-masing data berukuran 50x50 piksel. Gambar 3 menunjukkan 20 jenis variasi untuk huruf A.



Gambar 3. Data referensi untuk huruf A.

Adapun tampilan dari hasil implementasi perangkat lunak pengenalan tulisan tangan adalah sebagai berikut



Gambar 4. Citra masukan

Gambar 5. Citra Hasil Grayscale



Gambar 6. Citra Hasil Deteksi Tepi

DONT OPEN YOUR BOOK

Gambar 7. Citra Hasil Pelebaran / Dilation

DONT OPEN YOUR BOOK

Gambar 8. Citra Hasil Pengisian / Filling

DONT OPEN YOUR BOOK

Gambar 9. Citra Hasil Pengenalan Tulisan Tangan

Gambar 9 menunjukkan citra hasil ekstraksi karakter pada gambar masukan. Tabel berikut ini adalah hasil yang didapat dari percobaan aplikasi yang telah dilakukan:

	Input	Hasil	% Keberhasilan
1	DONT OPEN YOUR BOO	OONT OPEN YOUA AOOK	81,25%
2	DONT OPEN YOUR BOOK OPEN YOUR MIND	OONT OPEN YOUA AOOK OPEN YOUR PUNO	78,57%
3	LIFE IS ONE LONG PROCESS OF GETTING TIRED	EUEE US ONE EONG PROCESS OE GETTUNG TUAEO	70,59%
4	NEW MOON	NEW WQON	71,42%
5	PROGRAM ST	PROCRA P STUOU	66,67%
6	CITRA	ETTRA	60%
7	ABCDEFGHIJ KLMNOPQRS TUVWXYZ	AECOEAC HEJKKWNOPQQ STUVWXVZ	61,53%

	RATA-RATA	70%
--	-----------	-----

Berdasarkan hasil pengujian di atas, maka persentase pengenalan dari aplikasi yang dibuat adalah 70%.

X. KESIMPULAN DAN SARAN

10.1. Kesimpulan

Dari penelitian ini dapat ditarik kesimpulan bahwa aplikasi pengenalan tulisan tangan berbasis Matlab telah berhasil dibuat. Aplikasi ini mampu menganalisa tulisan tangan sehingga aplikasi ini diharapkan dapat mempermudah dalam mengkonversi tulisan tangan menjadi text digital. Pengujian pada 7 buah citra menunjukkan bahwa aplikasi ini memiliki kemampuan pengenalan karakter sebesar 70%.

Aplikasi pengenalan tulisan tangan ini dibuat berbasis Matlab dengan menggunakan metode Jaringan Syaraf Tiruan *Back Propagation*.

10.2. Saran

Saran yang dapat diberikan adalah:

1. Citra referensi perlu diperbanyak agar proses pengenalan karakternya menjadi lebih baik.

REFERENSI

- [1]. Aprijani, Dwi Astuti. 2011. "Aplikasi Jaringan Syaraf Tiruan Untuk Mengenali Tulisan Tangan Huruf A,B,C,D Pada Jawaban Soal Pilihan Ganda". Jurnal Matematika, Sains dan Teknologi Volume 12 nomor 1.
- [2]. Arief, Achmad Fauzi. 2010. "Perangkat Lunak Konversi Tulisan Tangan Menjadi Teks Digital"
- [3]. Emanuel, Andi Wahyu Rahardjo dan Arie Hartono. 2008. "Pengembangan Aplikasi Pengenalan Karakter Alfanumerik Dengan Menggunakan Algoritma Neural Network Three-Layer Backpropagation". Jurnal Informatika (4), 49 – 58
- [4]. Hermawan, Arief. 2006. Jaringan Syaraf Tiruan teori dan Aplikasi. Yogyakarta : ANDI
- [5]. Fausett, L. 1994. Fundamentals of Neural Network. Prentice-Hall International, Inc., Florida Institut of Technology.
- [6]. Fauziah, M Iwan Wahyuddin. 2009. "Metode Jaringan Saraf Tiruan Penjejukan Balik Untuk Pengenalan Huruf Cetak Pada Citra Digital". Jurnal Artificial, ICT Research Center UNAS (3)
- [7]. Fifin Hietania, Wayan Santiyasa, Ida Bagus Gede Dwidasmara, "Implementasi Backpropgation Dalam Pengolahan Citra Teks Tulisan Tangan Menjadi Teks Digital", Teknik Informatika, Universitas Udayana.

- [8]. Murni,A.,1992.Pengantar Pengolahan Citra. Jakarta : PT.Elex Media Komputindo.
- [9]. Dinesh Dileep, “A Feature Extraction Technique Based on Character Geometry fot Character Recognition”.
- [10]. Nugraha, A.P. dan A.B. Mutiara . 2002. “Metode ekstraksi data untuk pengenalan huruf dan angka tulisan tangan dengan menggunakan 10 jaringan syaraf buatan propagasi balik”
- [11]. Puspitaningrum,Diyah.2006. Pengantar Jaringan Syaraf Tiruan. Yogyakarta : ANDI