

ANALISIS DAN PERANCANGAN APLIKASI PENERJEMAH AKSARA JAWA MENGGUNAKAN METODE BACKPROPAGATION

Ardyandrea Ersty Surya¹, Ratri Dwi Atmaja², Suci Aulia³

^{1,2}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

³Prodi D3 Teknik Telekomunikasi, Fakultas Ilmu Terapan, Universitas Telkom ardyandrea1804@gmail.com¹,

ratriwiatmaja@tess.telkomuniversity.a.id², sucia@tass.telkomuniversity.ac.id³

Abstrak— Teknologi yang berkembang pesat berpengaruh terhadap kesadaran masyarakat pada warisan budaya. Banyak masyarakat yang telah melupakan budaya yang ada, salah satunya adalah aksara nusantara. Pada tugas akhir sebelumnya membahas tentang pengenalan aksara sunda menggunakan metode *Learning Vector Quantization (LVQ)* pada android dengan tingkat akurasi 60.90%. Pada tugas akhir lainnya tentang aksara jawa menggunakan metode *Self-Organizing Map (SOM)* pada matlab dengan tingkat akurasi 96.25% dengan waktu komputasi 7.9 detik.

Dari tugas akhir diatas, maka pada tugas akhir ini dirancang sebuah aplikasi pengenalan aksara jawa berbasis android. Pada tugas akhir ini *input* aksara jawa diambil menggunakan kamera android, lalu diproses pada *software* MATLAB. Parameter-parameter yang digunakan antara lain jenis deteksi tepi antara lain Canny, Sobel, Prewitt, dan Robert, serta jenis ekstraksi ciri antara lain *Sum*, DCT, DWT, dan DFT. Sedangkan metode klasifikasi yang digunakan adalah *JST backpropagation*. Keluaran dari aplikasi ini berupa tulisan bahasa Indonesia. Untuk menilai akurasi aplikasi ini dilakukan pengujian pada parameter-parameter pada *JST backpropagation* yang digunakan.

Dari pengujian yang telah dilakukan dapat disimpulkan bahwa akurasi terbaik adalah 80 % dengan parameter jumlah data latih 40 data, deteksi tepi Sobel, ekstraksi ciri DCT, jumlah *hidden layers* sebanyak 120 layer, *learning rate* 0.01, fungsi aktivasi yang digunakan *purelin*, dan algoritma *training* menggunakan *trainrp*.

Katakunci : Aksara Jawa, pengolahan citra, android, MATLAB, ekstraksi ciri, dan *Backpropagation*.

Abstract - Rapidly evolving technologies impact on public awareness on cultural heritage. Many people who have forgotten the existing culture, one of which is the script of the archipelago. At the end of the previous task discusses literacy Sundanese identifier using Learning Vector Quantization (LVQ) on android with an accuracy rate of 60.90%. At the other end of the task javanese script using the Self-Organizing Map (SOM) in matlab with 96.25% accuracy rate with 7.9 seconds of computing time.

From the thesis above, then on this last task designed an application identifier javanese script based on Android. In this final input javanese script taken using a camera android, and then processed in MATLAB software. The parameters used include, among others, the type of edge detection Canny, Sobel, Prewitt, and Robert, as well as the type of extraction characteristics include Sum, DCT, DWT, and DFT. While the classification method used is backpropagation. The output of this application in the form of written Indonesian. To assess the accuracy of these applications be tested on parameters used in the back propagation neural network.

From the testing that has been done can be concluded that the best accuracy was 80% with the amount of training data 40 parameters of data, Sobel edge detection, feature extraction DCT, number of hidden layers 120 layers, learning rate 0.01, function activation used is purelin, and training algorithm using trainrp.

Keyword: Javanese script, image processing, android, MATLAB, feature extraction, and *Backpropagation*.

I. PENDAHULUAN

Pengenalan huruf merupakan salah satu bidang pengenalan pola yang telah memberikan kontribusi yang berpengaruh pada perkembangan teknologi, dan lebih bermanfaat untuk menghubungkan antara manusia dan komputer yang bisa diterapkan pada berbagai bidang aplikasi.

Pada penelitian kali ini yang menjadi objek penelitian adalah aksara jawa dimana saat ini budaya daerah sudah perlahan ditinggalkan dan dilupakan. Maka dari itu dilakukan sebuah penelitian untuk membuat sistem penerjemah aksara jawa untuk mempermudah penduduk dalam mengenal kata dalam bentuk aksara.

Pada referensi [1] telah dilakukan penelitian tentang pengidentifikasian kepribadian seseorang melalui tulisan tangan dengan hasil akurasi optimal 91.11% dengan jarak 21 cm dan menggunakan jenis size strel diasi bernilai 8. Referensi selanjutnya [2] merancang sistem ANPR menggunakan raspberry pi sebagai alat pengenalan plat nomor menggunakan metode deteksi tepi sobel dan metode *template matching* dengan hasil akurasi 94.4% dengan waktu komputasi maksimal 5.6 detik. Pada referensi [3] telah dilakukan penelitian tentang pengenalan kata dalam aksara sunda menggunakan metode deteksi tepi dan *Learning Vector Quantization (LVQ)* berbasis pengolahan citra pada android. Pada penelitian tersebut didapatkan akurasi sistem paling tinggi 60.90 pada pengujian terhadap 10 kata sunda. Sementara pada referensi [4] dibahas penelitian tentang penerjemah aksara jawa ke huruf latin berbasis pengolahan citra digital dan jaringan syaraf tiruan *Self-Organization Map (SOM)*. Pada penelitian tersebut didapatkan

akurasi sistem pada aksara dasar mencapai 96.25% dengan waktu komputasi 7.9 detik, pada aksara tanda baca mencapai 53.1625% dengan waktu komputasi 6.4 detik, dan pada aksara murda mencapai 98.9583% dengan waktu komputasi 2 detik.

Pada tugas akhir ini, metode klasifikasi yang digunakan adalah metode *backpropagation*. Mengacu pada referensi [7] dan [12] akurasi yang didapat dengan menggunakan metode ini adalah lebih dari 80 %. Metode ini pun memiliki beberapa kelebihan bahwa metode *backpropagation* memiliki satu atau lebih *hidden layer*, tetapi karena menggunakan banyak layer untuk pelatihan maka dalam pelatihan memerlukan waktu komputasi yang lama [6].

Penulisan penelitian ini dibagi menjadi beberapa bagian yaitu bagian pertama membahas tentang latar belakang dari penelitian ini. Kemudian bagian ke dua membahas tentang dasar-dasar yang digunakan dalam penelitian ini. Bagian ke tiga membahas tentang pemodelan dari metode yang digunakan. Bagian ke empat membahas tentang hasil dari simulasi atau performansi system hasil dari kesimpulan. Bagian terakhir dari paper ini merupakan kesimpulan dari penelitian ini dan saran untuk pengembangan kedepan.

II. LANDASAN TEORI

A. Deteksi Tepi

Deteksi Tepi (Edge Detection) pada suatu citra adalah suatu proses yang menghasilkan tepi-tepi dari objek citra. Tujuan dari Deteksi Tepi adalah :

- Untuk menandai bagian yang menjadidi detail citra,
- Untuk memperbaiki detail dari citra yang kabur, yang terjadi karena *error* atau adanya efek dari proses akuisisi citra.
- Untuk mengubah citra 2D menjadi bentuk kurva.

Macam-macam metode untuk proses deteksi tepi yang sering digunakan [3], antara lain:

a. Metode Robert

Metode Robert merupakan teknik differensial pada arah horisontal dan vertikal, dengan ditambahkan proses konversi biner setelah dilakukan differensial. Teknik konversi biner yang disarankan adalah konversi biner dengan meratakan distribusi warna hitam dan putih.

Kernel yang digunakan dalam metode Robert adalah :

$$H = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \quad (2.1)$$

b. Metode Prewitt

Metode Prewitt merupakan metode pengembangan dari metode Robert dengan menggunakan filter HPF yang diberi satu angka nol penyangga.

Kernel filter yang digunakan dalam metode Prewitt ini adalah :

$$H = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (2.2)$$

c. Metode Sobel

Metode Sobel memiliki sensitivitas lebih terhadap tepian diagonal daripada tepian vertikal dan horisontal. Metode ini diterapkan secara terpisah untuk memperoleh gradien horisontal dan vertikalnya.

Kernel filter yang digunakan dalam metode Prewitt ini adalah:

$$H = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (2.3)$$

B. Discrete Cosine Transform (DCT)

Discrete Cosine Transform merupakan sebuah teknik yang digunakan untuk mengkonversi sinyal masukan kedalam komponen frekuensi pembentuknya, dengan menghitung nilai riil dari hasil transformasinya. Persamaan DCT yang digunakan yaitu :

$$X(k) = \sum_{n=0}^{N-1} x(n) \cos\left(\frac{(2n+1)(k-1)\pi}{2N}\right), \quad k = 1, 2, 3, \dots, N$$

Dimana $C(k) = \begin{cases} \frac{1}{\sqrt{N}}, & k = 1 \\ \sqrt{\frac{2}{N}}, & 2 \leq k \leq N \end{cases}$

N merupakan panjang dari x, serta x dan y mempunyai ukuran yang sama. Jika x adalah sebuah matriks maka dct mentransformasikan kolom dari matriks.

C. Discrete Fourier Transform (DFT)

Discrete Fourier Transform (DFT) merupakan transformasi sinyal masukan kedalam bentuk Fourier. Persamaan dari DFT yang digunakan adalah sebagai berikut.

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}$$

DFT memiliki basis sinyal sinusoidal dan berbentuk kompleks. Domain frekuensi yang dihasilkan akan memiliki bentuk kompleks. Dengan demikian hasil dari DFT akan direpresentasikan kedalam bentuk nilai *absolute*.

D. Discrete Wavelet Transform (DWT)

Discrete Wavelet Transform (DWT) merupakan teknik untuk mentransformasikan sinyal masukan kedalam bentuk wavelet diskrit. Wavelet digunakan untuk menganalisis bentuk suatu gelombang (waktu dan frekuensi). Prinsip dari DWT merupakan cara untuk mendapatkan representasi waktu dan skala dari sinyal masukan dengan teknik filter digital dan operasi sampling.

E. Backpropagation

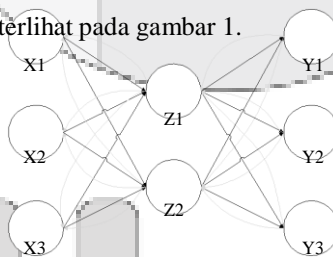
Backpropagation merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyinya. Algoritma backpropagation menggunakan error output untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan error ini, tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu. Arsitektur pada backpropagation berupa *multilayer* dengan beberapa *hidden layer* [6]. Pada saat perambatan maju, neuron-neuron diaktifkan dengan menggunakan fungsi aktivasi sigmoid, yaitu:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2.4}$$

Algoritma Backpropagation melakukan dua tahap perhitungan yaitu :

- a. Perhitungan maju untuk menghitung *error* antara *output actual* dan *target*
- b. Perhitungan mundur yang mempropagasikan balik *error* tersebut untuk memperbaiki bobot sinaptik pada semua neutron yang ada.

Arsitektur dari jaringan backpropagation seperti terlihat pada gambar 1.



Gambar.1 Arsitektur Jaringan Backpropagation

III. PERANCANGAN SISTEM

A. Pemodelan Sistem

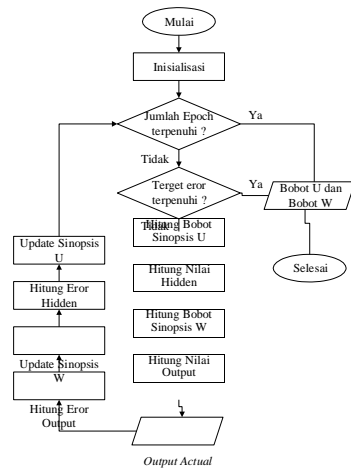
Dalam penelitian ini pemodelan sistem dibagi menjadi beberapa tahap antara lain : Tahap Akuisisi citra, Tahap *preprocessing*, Tahap ekstraksi ciri, dan tahap klasifikasi. Berikut blok diagram sistem.



Gambar.2 Blok Diagram Sistem

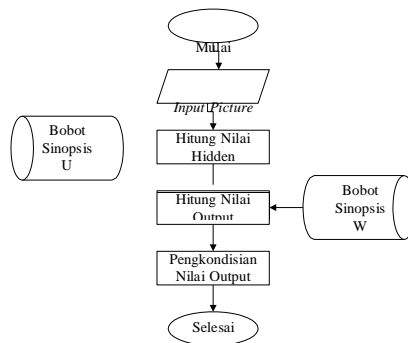
B. Backpropagation

Pada penelitian ini proses klasifikasi dibagi menjadi 2 proses yaitu proses *training* dan proses *testing*. Metode yang digunakan dalam penelitian akhir ini adalah metode backpropagation. Untuk proses *training* dilakukan untuk mencari bobot tiap synopsis dengan nilai *error* terkecil. Berikut merupakan diagram alur dalam proses *training*.



Gambar.3 Diagram Alir Proses *Training*

Untuk proses *testing*, dilakukan dengan menggunakan bobot yang telah dilatih sebelumnya. Berikut merupakan diagram alur dalam proses *testing*.



Gambar.4 Diagram Alir Proses *Testing*

IV. PENGUJIAN DAN ANALISIS

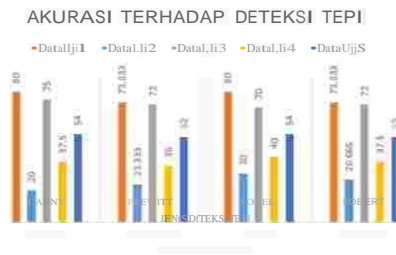
A. Pengujian terhadap Deteksi Tepi menggunakan metode *JST Backpropagation*

Pada tahap *preprocessing* dilakukan proses segmentasi citra. Proses segmentasi yang digunakan adalah metode deteksi tepi (*edge detection*). Ada pula beberapa metode dari deteksi tepi yang digunakan dalam penelitian ini antara lain Canny, Sobel, Prewitt, dan Robert. Pengujian kali ini dilakukan untuk mendapatkan metode deteksi tepi terbaik untuk proses klasifikasi. Parameter yang digunakan pada pengujian antara lain data latih yang digunakan adalah data latih dengan jumlah 40 data latih. Parameter pada *Backpropagation* yang digunakan antara lain jumlah *neuron* 120 layer, batas toleransi *error* atau *goal* adalah $1e-5$, *learning rate* 0.001, algoritma *training* adalah *trainlm*, dan fungsi aktivasi *purelin*. Berikut hasil pengujian.

Tabel 1 Hasil Akurasi terhadap Jenis Deteksi Tepi

Deteksi Tepi	Akurasi (%)					
	Latih	Uji 1	Uji 2	Uji 3	Uji 4	Uji 5
Canny	75	75	20	75	37.5	54
Sobel	77.5	80	30	70	40	54
Prewitt	72.5	73.333	23.333	72	35	52
Robert	72.5	73.333	26.666	72	37.5	52

Pada pengujian ini dapat dilihat bahwa deteksi tepi yang baik untuk penelitian ini adalah Sobel karena deteksi tepi Sobel memiliki akurasi data latih yang terbesar yaitu 77.5 %. Pengujian data uji yang dilakukan mendapatkan hasil akurasi data uji terbesar adalah 80 % dengan menggunakan data uji ke-1.



Gambar 5 Grafik Hasil Akurasi terhadap Jenis Deteksi Tepi

B. Pengujian terhadap Ukuran Normalisasi menggunakan JST *Backpropagation*

Pada tahap *preprocessing* dilakukan proses normalisasi citra. Proses normalisasi yang diuji meliputi ukuran normalisasi 50x125, 100x250, 150x375 dan 200x500. Pengujian kali ini dilakukan untuk mendapatkan ukuran citra terbaik untuk proses klasifikasi.

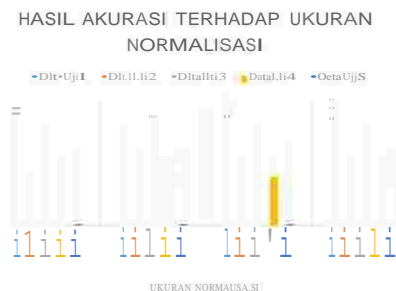
Parameter yang digunakan pada pengujian antara lain data latih yang digunakan adalah data latih dengan jumlah 40 data latih dan metode deteksi tepi Sobel. Parameter pada *Backpropagation* yang digunakan antara lain jumlah *neuron* 120 layer, batas toleransi *error* atau *goal* adalah $1e-5$, *learning rate* 0.001, algoritma *training* adalah *trainlm*, dan fungsi aktivasi *purelin*.

Berikut hasil dari pengujian terhadap ukuran normalisasi yang digunakan.

Tabel 2 Hasil Akurasi terhadap Ukuran Normalisasi

Ukuran Normalisasi	Akurasi (%)					
	Latih	Uji 1	Uji 2	Uji 3	Uji4	Uji 5
50x125	70	63.333	30	70	32.5	46
100x250	75	70	33.333	75	40	52
150x375	72.5	76.666	23.333	70	40	52
200x500	70	73.333	23.333	70	37.5	48

Pada pengujian ini dapat dilihat bahwa ukuran normalisasi yang baik untuk penelitian ini adalah 100x250 karena ukuran normalisasi 100x250 memiliki akurasi data latih yang terbesar yaitu 75 %. Pengujian data uji yang dilakukan mendapatkan hasil akurasi data uji terbesar adalah 76.666 % dengan menggunakan data uji ke-1 pada ukuran normalisasi 150x375.



Gambar 6 Grafik Akurasi terhadap Ukuran Normalisasi

C. Pengujian terhadap Ekstraksi Ciri menggunakan JST *Backpropagation*

Proses ekstraksi ciri yang diuji antara lain *sum* 5x5, DCT, DWT, dan DFT. Pada setiap ekstraksi ciri yang dilakukan menghasilkan jumlah data latih yang berbeda-beda, sehingga pengujian ini dilakukan untuk mengetahui pengaruh jenis ekstraksi ciri terhadap akurasi yang didapatkan.

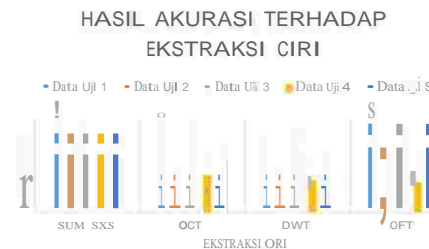
Parameter yang digunakan pada pengujian antara lain data latih yang digunakan adalah data latih dengan jumlah 40 data latih, deteksi tepi sobel, dan normalisasi 100x250. Parameter pada *Backpropagation* yang digunakan antara lain jumlah *neuron* 120 layer, batas toleransi *error* atau *goal* adalah $1e-5$, *learning rate* 0.001, algoritma *training* adalah *trainlm*, dan fungsi aktivasi *purelin*.

Berikut hasil dari pengujian terhadap jenis ekstraksi ciri yang digunakan.

Tabel 3 Hasil Akurasi terhadap Ekstraksi Ciri

Ekstraksi	Akurasi (%)					
	Latih	Uji 1	Uji2	Uji 3	Uji4	Uji 5
Sum 5x5	75	76.666	26.666	75	45	58
DCT	77.5	80	30	75	40	54
DWT	65	60	30	65	35	44
DFT	70	73.333	23.333	70	32.5	52

Pada pengujian ini dapat dilihat bahwa ekstraksi ciri yang baik untuk penelitian ini adalah DCT karena ekstraksi ciri DCT memiliki akurasi data latih yang terbesar yaitu 77.5 %. Pengujian data uji yang dilakukan mendapatkan hasil akurasi data uji terbesar adalah 80 % dengan menggunakan data uji ke-1.



Gambar 7 Grafik Hasil Akurasi Terhadap Ekstraksi Ciri

D. Pengujian Pengaruh *Learning Rate*

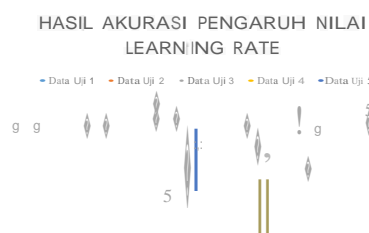
Pada *JST Backpropagation* salah satu parameternya adalah *learning rate*. *Learning rate* ini digunakan untuk menentukan laju pemahaman, dimana semakin besar nilai *learning rate* semakin cepat pula proses pelatihannya. Tetapi nilai *learning rate* yang tinggi berpengaruh pada kestabilan algoritma yang digunakan. Adapun nilai *learning rate* yang diuji antara lain 0.001, 0.005, 0.01, 0.05, 0.1 dan 0.5.

Parameter yang digunakan pada pengujian antara lain data latih yang digunakan adalah data latih dengan jumlah 40 data latih, deteksi tepi Sobel, normalisasi 100x125, dan ekstraksi ciri DCT. Parameter pada *Backpropagation* yang digunakan antara lain jumlah *neuron* 120 layer, batas toleransi *error* atau *goal* adalah 1e-5, algoritma *training* adalah *trainlm*, dan fungsi aktivasi *purelin*.

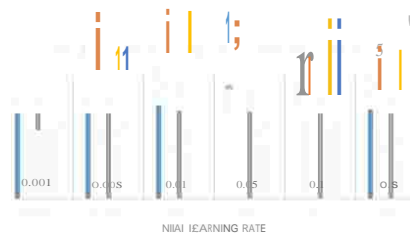
Tabel 4 Hasil Akurasi terhadap Pengaruh *Learning Rate*

Learning Rate	Akurasi (%)					
	Latih	Uji 1	Uji2	Uji3	Uji4	Uji 5
0.001	75	70	30	70	40	50
0.005	75	70	33.333	70	45	50
0.01	72.5	76.666	23.333	72.5	37.5	54
0.05	72.5	63.333	33.333	72	45	48
0.1	72.5	66.666	33.333	70	45	50
0.5	70	73.333	23.333	70	35	48

Pada pengujian ini dapat dilihat bahwa nilai *learning rate* yang baik untuk penelitian ini adalah 0.001 dan 0.005 karena nilai ini memiliki akurasi data latih yang terbesar yaitu 75 %. Pengujian data uji yang dilakukan mendapatkan hasil akurasi data uji terbesar adalah 76.666 % dengan menggunakan data uji ke-1.



Gambar 8 Grafik Hasil Akurasi terhadap Pengaruh *Learning Rate*



Gambar 8 Grafik Hasil Akurasi terhadap Pengaruh *Learning Rate*

E. Pengujian Pengaruh Fungsi Aktivasi

Pada JST *Backpropagation* salah satu parameternya adalah fungsi aktivasi. Fungsi aktivasi ini digunakan untuk megnaktifkan *neuron-neuron* pada saat proses *forward propagation*. Adapun jenis fungsi aktivasi yang diuji antara lain *tansig*, *logsig*, dan *purelin*.

Parameter yang digunakan pada pengujian antara lain data latih yang digunakan adalah data latih dengan jumlah 40 data latih, deteksi tepi Sobel, normalisasi 100x125, dan ekstraksi ciri DCT. Parameter pada *Backpropagation* yang digunakan antara lain jumlah *neuron* 120 layer, nilai *goal* 1e-5, nilai *learning rate* 0.001, dan algoritma *training* adalah *trainlm*.

Tabel 5 Hasil Akurasi terhadap Pengaruh Fungsi Aktivasi

Ekstraksi Ciri	Akurasi (%)					
	Latih	Uji 1	Uji2	Uji 3	Uji4	Uji5
<i>Tansig</i>	37.5	43.333	13.333	37.5	20	28
<i>Logsig</i>	2.5	3.333	3.333	2.5	5	2
<i>Pure/in</i>	75	73.333	30	70	45	52

Pada pengujian ini dapat dilihat bahwa fungsi aktivasi yang baik untuk penelitian ini adalah *purelin* karena *purelin* memiliki akurasi data latih yang terbesar yaitu 75 %. Pengujian data uji yang dilakukan mendapatkan hasil akurasi data uji terbesar adalah 73.333 % dengan menggunakan data uji ke-1.



Gambar 9 Grafik Hasil Akurasi terhadap Pengaruh Fungsi Aktivasi

F. Pengujian Pengaruh Algoritma Training

Pada pengujian ini diuji pengaruh algoritma *training* terhadap akurasi yang dihasilkan. Adapun jenis algoritma *training* yang diuji antara lain *trainlm*, *traingd*, *traindm*, dan *trainrp*.

Parameter yang digunakan pada pengujian antara lain data latih yang digunakan adalah data latih dengan jumlah 40 data latih, deteksi tepi Sobel, normalisasi 100x125, dan ekstraksi ciri DCT. Parameter pada *Backpropagation* yang digunakan antara lain jumlah *neuron* 120 layer, nilai *goal* 1e-5, nilai *learning rate* 0.001, dan dan fungsi aktivasi *purelin*.

Tabel 6 Hasil Akurasi terhadap Pengaruh Algoritma Training

Ekstraksi Ciri	Akurasi (%)					
	Latih	Uji 1	Uji2	Uji 3	Uji4	Uji5
<i>Trainlm</i>	75	73.333	23.333	70	30	52
<i>Traingd</i>	5	6.666	6.666	5	5	6
<i>Trainrp</i>	75	76.666	3.333	72	45	54

Pada pengujian ini dapat dilihat bahwa algoritma *training* yang baik untuk penelitian ini adalah *Trainlm* dan *Trainrp* karena kedua algoritma memiliki akurasi data latih yang terbesar yaitu 75 %. Pengujian data uji yang dilakukan mendapatkan hasil akurasi data uji terbesar adalah 76.666 % dengan menggunakan data uji ke-1.



Gambar 10 Grafik Hasil Akurasi terhadap Pengaruh Algoritma Training

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berikut ini adalah kesimpulan dari penelitian yang telah dilakukan.

1. Dari pengujian pada data latih didapatkan kata yang sering tidak memenuhi target adalah kata alamat, bagaimana, hari, pulang, ruang, dan tahun. Hal tersebut dikarenakan terdapat *noise* pada gambar yang menyebabkan karakter tulisan sulit untuk diklasifikasikan atau terklasifikasi pada *class* yang berbeda.
2. Berdasarkan pengujian yang telah dilakukan didapatkan akurasi terbaik sebesar 80 % dengan menggunakan deteksi tepi Sobel dan ekstraksi ciri DCT.
3. Dari pengujian deteksi tepi yang digunakan didapatkan bahwa deteksi tepi yang baik untuk penelitian ini adalah deteksi tepi Sobel karena akurasi dari deteksi tepi ini mencapai akurasi tertinggi sebesar 80 %.
4. Dari pengujian ukuran normalisasi didapatkan bahwa ukuran optimal untuk penelitian ini adalah ukuran 150x375. Hal ini disebabkan karena ukuran tersebut memperoleh akurasi yang tertinggi yaitu 76.666 %.
5. Dari pengujian ekstraksi ciri didapatkan bahwa ekstraksi ciri yang baik untuk penelitian ini adalah ekstraksi ciri DCT, karena DCT memiliki akurasi yang lebih tinggi dari ekstraksi ciri lainnya yaitu sebesar 80 %.
6. Pengujian parameter *Backpropagation*, untuk nilai *learning rate* yang digunakan adalah 0.01 karena pada nilai tersebut akurasi yang didapat paling tinggi yaitu sebesar 76.666 %.
7. Fungsi aktivasi yang digunakan pada penelitian ini adalah *purelin*, karena dari hasil pengujian akurasi dari fungsi aktivasi *purelin* memiliki nilai tertinggi yaitu sebesar 73.333 %.
8. Sedangkan algoritma *training* yang digunakan adalah *trainrp*, karena akurasi yang didapatkan sebesar 76.666 % yang merupakan akurasi tertinggi dari algoritma yang lain.

B. Saran

Berikut ini adalah saran yang diberikan penulis untuk dilakukan pada penelitian selanjutnya.

1. Dapat dilakukan pengujian terhadap jenis kertas yang digunakan untuk melihat performansi dari sistem.
2. Dapat dilakukan pengujian terhadap noise pada gambar masukan untuk dapat meningkatkan performansi sistem.
3. Dapat mengubah gambar masukan menjadi tulisan per kata sehingga banyak kata yang dapat diterjemahkan.

REFERENSI

- [1] N. Safaat H., Pemrograman Aplikasi Mobeli Smartphone dan Tablet PC berbasis Android Edisi Revisi, Bandung: Penerbit Informatika, 2012.
- [2] J. Kunci, "Sejarah Singkat Lahirnya Huruf Jawa," 10 2012. [Online]. Available: <http://www.jurukunci.net/2012/10/sejarah-singkat-lahirnya-huruf-jawa-ha.html>. [Accessed 18 February 2016].
- [3] Anonim, "Jaringan Saraf Tiruan," [Online]. Available: <http://entin.lecturer.pens.ac.id/Kecerdasan%20Buatan/Buku/Bab%208%20Jaringan%20Syaraf%20Tiruan.pdf>. [Accessed 30 Maret 2016].
- [4] Anonim, "Huruf Jawa," [Online]. Available: http://www.gimonca.com/indonesia/huruf_jawa.html. [Accessed 18 Februari 2016].
- [5] Anonim, "Algoritma Pelatihan Backpropagation," Februari 2013. [Online]. Available: <http://www.metode-algoritma.com/2013/02/backpropagation.html>. [Accessed 4 April 2016].
- [6] J. J. Siang, Jaringan Saraf Tiruan dan Pemrogramannya menggunakan MATLAB, Bandung: Penerbit Andi, 2009.
- [7] R. Y. N. Fu'adah, Analisis Deteksi Fraktur Batang (Diaffisis) pada Tulang Tibia dan Fibula berbasis Pengolahan Citra Digital dan Jaringan Saraf Backpropagation, Bandung: Universitas Telkom, 2012.
- [8] F. M. Hasibuan, Desain dan Implementasi Sistem Penerjemah Aksara Jawa ke Huruf Latin Berbasis Pengolahan Citra Digital dan Jaringan Syaraf Tiruan Self-Organizing Map (SOM), Bandung: Universitas Telkom, 2011.
- [9] D. D. Lestari, Perancangan Pengenal Kata dalam Aksara Sunda menggunakan Metode Deteksi Tepi dan LVQ berbasis Pengolahan Citra pada Android, Bandung: Universitas Telkom, 2015.
- [10] T. P. Tampubolon, Simulasi dan Analisis Keamanan Teks menggunakan Metode Stenografi Discrete Wavelet Transform (DWT) dan Metode Enkripsi Cellular Automata, Bandung: Universitas Telkom, 2016.
- [11] J. Vilda, Pengenal Pola Tanda Tangan menggunakan Metode Ekstraksi Ciri DCT, DFT dan Filter 2D Gabor Wavelet, Bandung: Universitas Telkom, 2009.
- [12] I. S. M. L. Tambunan, Identifikasi Plat Nomor Kendaraan Reguler atau Nonreguler berbasis Pengolahan Citra menggunakan Jaringan Syaraf Tiruan Backpropagation, Bandung: Universitas Telkom, 2012.