

Pengenalan Karakter Tulisan Menggunakan Metode Backpropagation Neural Network

Ni'mah Moham¹, Felix Andika Dwiyanto², Herman Santoso Pakpahan³, Islamiyah⁴, Hario Jati Setyadi⁵

^{1,3,4,5}Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Mulawarman

²Pascasarjana Universitas Negeri Malang

nimahmoham@gmail.com; ayikgugun@gmail.com; herman.pakpahan@fkti.unmul.ac.id; islamiyah@fkti.unmul.ac.id; hario.setyadi@fkti.unmul.ac.id

INFORMASI ARTIKEL

Histori Artikel

Diterima : 2 Juli 2019

Direvisi : 17 Juli 2019

Diterbitkan : 30 Agustus 2019

Kata Kunci:

Aksara Lontara
Pengenalan Pola
BPNN
MSE

ABSTRAK

Artikel ini bertujuan untuk menjelaskan langkah-langkah kerja metode *Backpropagation Neural Network* (BPNN) dalam mengenali pola Aksara Lontara Bugis Makassar dan menjelaskan seberapa akurat dalam mengenali pola aksara Lontara Bugis Makassar. Dari hasil pengujian, diperoleh tingkat akurasi sebesar 76.08%, dengan parameter *learning rate* sebesar 0.02, *epoch* maksimum sebesar 50 *epoch* dan *hidden layer* sebanyak 90 *neuron* berdasarkan ciri 8. Adapun, performa *mean square error* (MSE) sebesar 0.00424 telah diperoleh. Namun demikian, waktu yang dibutuhkan saat proses pembelajaran terbilang cukup lama yaitu 16 menit 56 detik. Berdasarkan hasil pengujian metode BPNN dapat direkomendasikan untuk mengenali pola aksara Lontara Bugis Makassar dalam rangka menunjang pembelajaran kepada masyarakat.

2019 SAKTI – Sains, Aplikasi, Komputasi dan Teknologi Informasi.

Hak Cipta.

I. Pendahuluan

Warisan kekayaan budaya Indonesia sangatlah beragam salah satunya adalah sebuah perwujudan berupa bahasa daerah yang memiliki aksara atau tulisan asli di wilayah nusantara. Aksara Lontara Bugis Makassar merupakan bagian dari aksara nusantara yang berasal dari wilayah Sulawesi Selatan. aksara Lontara sebagai salah satu kebudayaan asli Bugis Makassar selayaknya menjadi titik perhatian masyarakat umum khususnya suku Bugis di Sulawesi Selatan. Jika tidak diterapkan maka arus modernisasi dapat menyingkirkannya, untuk itu aksara Lontara pada saat ini perlu mendapat perhatian khusus karena berada pada posisi terancam akibat keterbatasan data dan informasi. Menurut Profesor Nurhayati Rahman (2008), ada tiga hal yang menyebabkan aksara Lontara Bugis Makassar mulai ditinggalkan. Pertama, masyarakat merasa rendah diri, takut dianggap kolot kalau berbahasa daerah. Padahal, untuk tetap melestarikan aksara Lontara, masyarakat semestinya intens berbahasa Bugis Makassar. Penyebab kedua adalah masuknya teknologi industri pop, yang membuat masyarakat lebih memilih perkembangan budaya asing. Yang terakhir adalah pengaruh globalisasi, di mana budaya luar begitu mudah masuk dan menggantikan budaya lokal. Masyarakat lebih condong menirukan gaya Barat dibanding bangga atas kearifan lokal tempat mereka menetap. Sebagai upaya pelestarian kembali aksara Lontara Bugis Makassar, penelitian tersebut menyarankan agar aksara Lontara Bugis Makassar dapat kembali diajarkan dengan semangat yang penuh akan antusias di sekolah-sekolah dasar hingga sekolah menengah pertama, sehingga dapat memudahkan pelajar untuk kembali mengenal lebih dekat dengan aksara Lontara Bugis Makassar (Abidin, 2006; Ahmad, 2014; Rahman, 2008).

Dalam rangka membantu masyarakat terutama dalam mengenali pola aksara Lontara Bugis Makassar berbagai pendekatan telah dilakukan. Seiring perkembangan teknologi yang cukup pesat maka pendekatan saat ini cenderung memanfaatkan kemampuan teknologi. Salah satu pendekatan teknologi pengenalan pola dapat dilakukan dengan bidang keilmuan kecerdasan buatan (Artificial Intelligence) yang merupakan salah satu dari bagian keilmuan bertujuan membuat mesin (Komputer) dapat melakukan atau menirukan pekerjaan seperti yang dilakukan manusia. Agar komputer bisa bertindak seperti manusia tersebut, maka komputer juga harus diberikan pengetahuan sehingga mempunyai kemampuan untuk meniru perilaku manusia.

Artikel ini bertujuan untuk menerapkan metode *Backpropagation Neural Network* (BPNN) dalam mengenali pola aksara Lontara Bugis Makassar dengan tulisan tangan. Hasil penelitian diharapkan dapat membantu masyarakat dalam mempelajari aksara Lontara Bugis Makassar. Paper ini terdiri dari empat bagian.

Bagian pertama adalah motivasi penelitian ini. Bagian kedua adalah menjelaskan metodologi dan teknik yang digunakan. Bagian ketiga adalah menerangkan hasil pengujian dengan metode BPNN. Dan, bagian terakhir adalah kesimpulan dan saran yang diperoleh setelah pengujian, serta rencana penelitian selanjutnya.

II. Metodologi

Pada bagian ini, akan dijelaskan secara singkat pola aksara lontara, metode BPNN dan pengolahan citra yang digunakan dalam penelitian ini.

A. Aksara Lontara

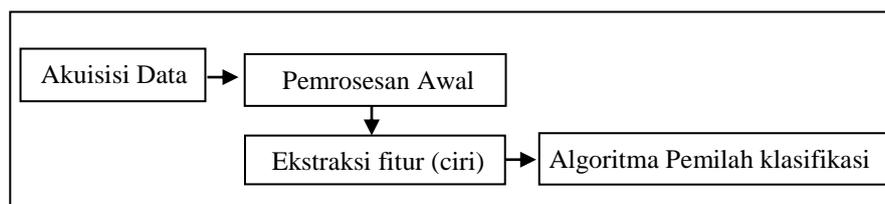
Lontara adalah aksara tradisional masyarakat Bugis-Makassar. Bentuk aksara berasal dari "sulapa eppa wala suji". Wala suji berasal dari kata wala yang artinya pemisah/pagar/penjaga dan suji yang berarti putri. Wala Suji adalah sejenis pagar bambu dalam acara ritual yang berbentuk belah ketupat. Sulapa eppa (empat sisi) adalah bentuk mistis kepercayaan Bugis-Makassar klasik yang menyimbolkan susunan semesta, api-air-angin-tanah. Huruf lontara ini pada umumnya dipakai untuk menulis tata aturan pemerintahan dan kemasyarakatan. Naskah ditulis pada daun lontar menggunakan lidi atau kalam yang terbuat dari ijuk kasar (kira-kira sebesar lidi)(Drs. Mattulada, 1974).

B. Pengenalan Pola

Pengenalan pola adalah suatu cabang keilmuan kecerdasan buatan. Beberapa penulis (Fatta, 2009) mengutip beberapa definisi pengenalan pola dari beberapa peneliti sebelumnya, yaitu penentuan suatu objek fisik atau kejadian ke dalam salah satu atau beberapa kategori (Dua dan Hart, 1973), ilmu pengetahuan yang menitik-beratkan pada deskripsi dan klasifikasi (pengenalan) dari suatu pengukuran (Schalkoff, 1992).

C. Pengenalan Tulisan Tangan

Menurut Plamondon dan Srihari (Wu dan Yu, 2008), pengenalan tulisan tangan adalah proses perubahan suatu bahasa yang dihadirkan dalam bentuk ruang melalui tulisan menjadi representasi simbolik. Tahapan umum yang dilakukan pada sistem pengenalan tulisan tangan dapat dilihat pada Gambar 1.(Fatta, 2009):



Gambar 1. Tahapan pengolahan tulisan tangan

D. Pengolahan Citra

Pengolahan citra adalah pemrosesan citra yang bertujuan agar kualitas citra menjadi lebih baik dan sesuai dengan keinginan pemakai sehingga mudah diinterpretasi oleh manusia atau mesin. Pengolahan citra adalah proses pengolahan sinyal yang *input* nya adalah citra. *Output*nya dapat berupa citra atau sekumpulan karakteristik atau parameter yang berhubungan dengan citra. Pengolahan citra pada dasarnya adalah suatu sistem yang tujuannya adalah mengklasifikasikan objek-objek ke dalam kategori-kategori atau kelas-kelas berdasarkan baik pada apriori pengetahuan atau pada informasi statistik yang diambil dari pola.

1. Akuisisi Citra

Akuisisi Citra atau *Image Acquisition* merupakan proses menangkap (*capture*) atau memindai (*scan*) suatu citra analog sehingga diperoleh sebuah citra digital. Pada tahap ini diawali dengan melakukan pengambilan gambar pada objek dengan menggunakan media *scanner*. Objek berupa citra aksara Lontara Bugis Makassar yang diambil dari 16 responden yang berbeda dengan kertas berukuran A4.

2. Peningkatan Kualitas Citra

a. *Cropping*

Cropping objek sendiri merupakan proses pemotongan area citra sehingga hanya tersisa area citra yang berisi objek saja. Proses ini dilakukan untuk membuang area kosong yang tidak berguna di sekeliling objek dan menghindari kesalahan pembelajaran karena letak objek (tulisan huruf) yang berbeda-beda posisinya. *Cropping* yang akan dilakukan adalah dengan memotong 23 titik citra secara bersamaan dan menghasilkan 23 citra aksara yang siap diolah oleh tahap selanjutnya. Pada dasarnya mesin *cropping* ini akan dibuat terpisah dengan bagian-bagian lain pada proses pengolahan citra.

b. Konversi citra RGB ke citra *grayscale*

Grayscale atau skala keabuan merupakan citra yang nilai intensitas pikselnya berdasarkan derajat keabuan. Pada tahap ini citra RGB akan dikonversi menjadi citra *grayscale* sehingga dihasilkan hanya satu kanal warna.

3. Perbaikan Kualitas Citra

Penajaman citra atau biasa disebut dengan transformasi ini adalah salah bagian dari proses perbaikan kualitas citra yang mana digunakan dalam meningkatkan kontras warna dan cahaya pada suatu citra. Proses ini bertujuan untuk mempermudah dalam proses interpretasi dan analisis citra. Penajaman kontras dalam citra sendiri merupakan sebuah cara untuk memperbaiki tampilan dengan memaksimalkan kontras antara pencahayaan dan penggelapan suatu citra.

4. Segmentasi Citra

Thresholding merupakan salah satu metode segmentasi citra yang memisahkan objek dengan background dalam suatu citra berdasarkan tingkat kecerahannya atau gelap terangnya. *Thresholding* merupakan proses mengubah citra berwarna atau citra keabuan menjadi citra biner (hitam-putih). Proses ini menggunakan konsep *thresholding* (pengambangan). Proses pengambangan ini mengambil nilai warna pada tiap piksel citra lalu dibandingkan dengan nilai *threshold* (ambang batas). Tiap piksel citra lalu diubah warnanya menjadi warna putih jika nilai warna atau keabuan nya diatas ambang batas. Sebaliknya jika nilai warna atau keabuan nya dibawah ambang batas maka akan diubah menjadi warna hitam.

5. Ekstraksi Ciri Ukuran

a. *Resizing* citra

Resizing citra adalah proses pengecilan ukuran citra berdasarkan jumlah piksel nya. Misalnya citra dengan ukuran 186 x 186 piksel dan dikecilkan menjadi 42 x 42 piksel. Proses *resizing* citra diperlukan untuk mengurangi jumlah piksel citra namun tetap dengan mempertimbangkan bentuk objek supaya tidak mengalami perubahan signifikan. Jumlah piksel citra dikurangi agar jumlah *node* atau *neuron* pada *layer input* jaringan syaraf tiruan yang digunakan tidak terlalu banyak sehingga mengakibatkan waktu pemrosesan terlalu lama.

b. Penipisan citra

Penipisan citra atau *thinning* menggunakan proses iteratif yang menghilangkan piksel-piksel hitam (mengubahnya menjadi piksel putih) pada tepi-tepi pola. Tujuan penipisan adalah mengurangi bagian yang tidak perlu (*redundant*) sehingga hanya dihasilkan informasi yang penting saja. Adapun, tahap proses pengolahan citra dapat dilihat pada Gambar 2.



Gambar. 1. Flowchart proses pengolahan citra

E. Metode *Backpropagation Neural Network* (BPNN)

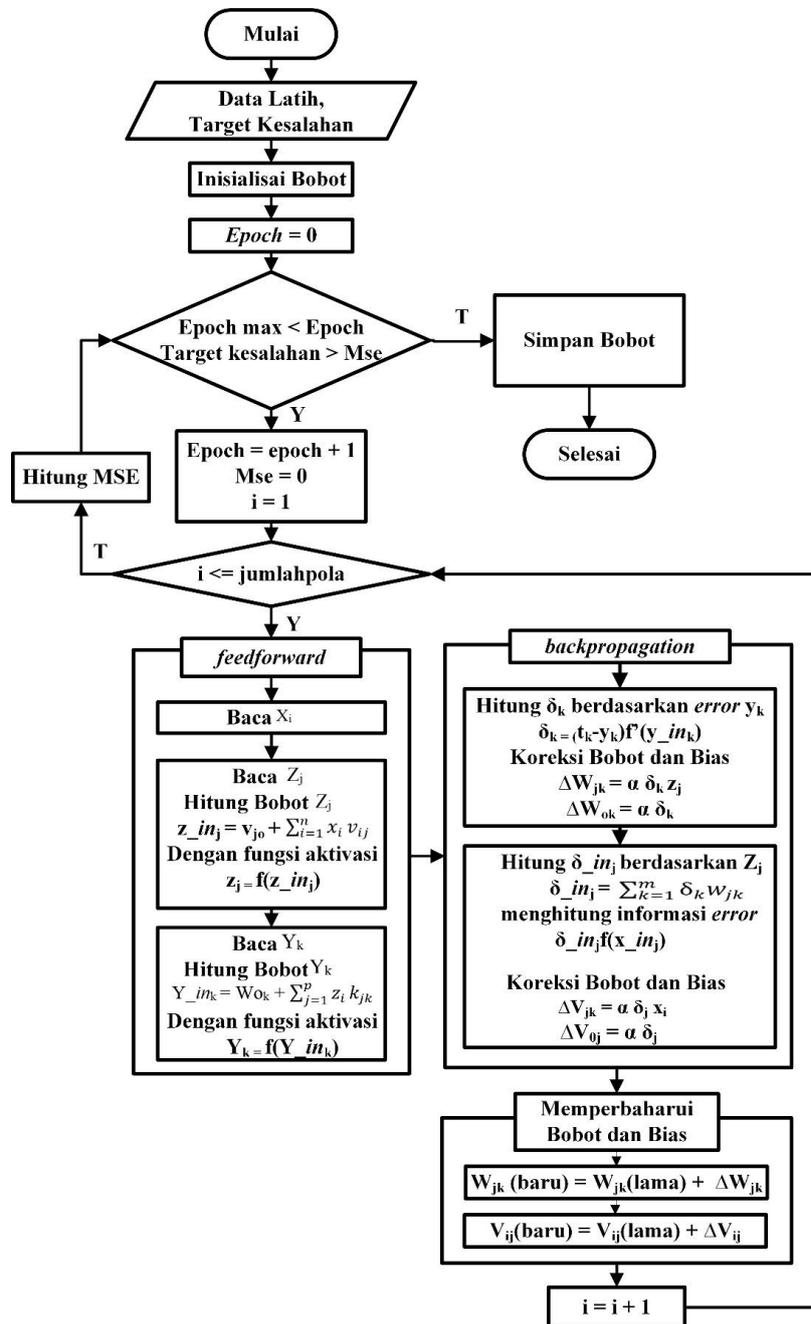
Metode BPNN pertama kali dirumuskan oleh Paul Werbos pada tahun 1974, kemudian dikembangkan oleh David Parker pada tahun 1982. Selanjutnya, tahun 1986 dipopulerkan lagi oleh Rumelhart dan McClland (Dharwal & Kaur, 2016; Gaffar, Gani, Havaluddin, Gaffar, & Alfred, 2019). Metode BPNN banyak digunakan dalam memecahkan berbagai persoalan simulasi termasuk prediksi. Terdapat tiga tahapan dalam BPNN, yaitu Umpan maju (Feed forward), umpan mundur (*backward*) dan pembaharuan bobot. Sebelum memasuki ketiga

tahapan tersebut, diperlukan untuk menginisialisasi bobot dengan nilai acak dengan interval 0 hingga 1. Kemudian pola-pola data akan dinormalisasi dengan tujuan memperoleh nilai bobot yang diinginkan. Nilai bobot akan terus mengalami perubahan tiap putaran hingga mencapai batas pelatihan setelah melalui umpan maju, umpan mundur dan pembaharuan bobot. Setelah bobot diperoleh, maka akan masuk ke tahapan akhir, yaitu tahapan pengujian. Hasil pengujian yang diperoleh kemudian akan dibandingkan dengan data sesungguhnya (Graves, 2011; Hagan & Menhaj, 1994; Kurniawan, Pamungkas, Wibawa, & Haviluddin, 2016; Kurniawan et al., 2015; Lehtokangas, 1999; Rojas & Rojas, 2011).

Metode BPNN merupakan algoritma pelatihan terbimbing yang mempunyai banyak lapisan. BPNN menggunakan *error output* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* ini, tahap perambatan maju (*forward*) harus dikerjakan terlebih dahulu. Syarat fungsi aktivasi dalam BPNN adalah bersifat kontinu, terdiferensial dengan mudah, dan merupakan fungsi yang tidak turun. Fungsi aktivasi yang dapat memenuhi ketiga syarat tersebut adalah logsig, tansig, dan purelin. Tahapan BPNN dijelaskan sebagai berikut:

1. Dimulai dengan memasukkan data latih, data *input* yang ada merupakan hasil dari proses pengolahan citra, jadi data latih ini sendiri merupakan sebuah citra digital yang sudah siap diproses oleh Sistem pengenalan pola dan menentukan target kesalahan yang akan dicapai.
2. Tahap kedua adalah inisialisasi terhadap semua bobot dengan bilangan acak kecil.
3. *Epoch* = 0 berarti belum adanya proses yang terjadi atau dapat dikatakan awal pemrosesan. *Epoch* sendiri dapat disebut sebagai iterasi.
4. Merupakan sebuah kondisi yang mana Jika *error* lebih kecil atau sama dengan target kesalahan dan atau *epoch* lebih besar atau sama dengan *epoch* maksimum maka bobot yang terbentuk disimpan dalam suatu file bobot. Namun sebagai awal kondisi yang mana belum ada *epoch* yang terjadi dan secara otomatis nilai *error* juga belum dapat ditentukan, Jadi kondisi yang ada adalah bersifat benar. Maka akan dilanjutkan dengan tahap berikutnya.
5. Inisialisasi $epoch = epoch + 1$, $error = 0$ dan $i = 1$. Inisialisasi ini menandakan akan dilakukannya sebuah *epoch* yang merupakan satu kesatuan dari kondisi yang berada di atasnya. Apabila telah dilakukan sebuah *epoch* dan kondisi yang ada bersifat benar maka *epoch* akan kembali berulang dengan kembali menginisialisasi $error = 0$ dan $i = 1$ hingga nilai dari *Epoch* dan atau Mse sesuai dengan kondisi yang ditentukan yaitu "T" atau kondisi yang tidak sesuai pada *flowchart* tersebut.
6. Setelah itu terdapat kondisi yang berisi $i = \text{jumlah pola yang memiliki dua kemungkinan apabila kondisi itu bersifat benar atau "Y"}$ maka akan memulai proses BPNN yang diawali dengan tahap Umpan Maju (*feedforward*), Umpan Mundur (*backward*) kemudian memperaruhi bobot dan bias. Apabila tahap tersebut telah selesai dikerjakan maka akan terjadinya perulangan dengan $i = i + 1$ yang digunakan sebagai acuan pada kondisi ini dan secara otomatis kondisi akan menjadi "T" karena nilai i sudah tidak lebih kecil sama dengan jumlah pola, kemudian Hitung *error* dengan rumus yang ada. Setelah *error* dihitung maka akan masuk kedalam kondisi pertama yaitu $Epoch \max < Epoch$ dan atau Target kesalahan $> error$.
7. Tahap Umpan Maju (*feedforward*) yang bermula pada indentifikasi X_i yang merupakan unit masukan yang menerima sinyal dan meneruskannya ke unit tersembunyi yang berada di atasnya. Setelah itu menghitung semua keluaran pada unit tersembunyi yaitu Z_j yang dikalikan dengan bobotnya dan dijumlahkan serta ditambahkan dengan biasnya. Kemudian hitung fungsi aktivasi untuk menghitung sinyal keluaran, $z_j = f(z_{inj})$, lalu mengirimkan sinyal ini kesemua unit pada *layer* di atasnya (unit keluaran). Selanjutnya menghitung semua keluaran jaringan di unit Y_k yang dikalikan dengan bobotnya dan dijumlahkan serta ditambahkan dengan biasnya dan hitung fungsi aktivasi untuk menghitung sinyal keluaran, $Y_k = f(Y_{ink})$ dan mengirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit *output*). Keluaran dari jari jaringan Y_k dibandingkan dengan target yang harus dicapai, lalu selisihnya adalah kesalahan yang terjadi.
8. Masuk kedalam tahap Umpan Mundur (*backward*) yang diawali dengan menghitung faktor δ unit keluaran berdasarkan kesalahan disetiap unit keluaran yaitu Y_k . δ_k merupakan unit kesalahan yang nantinya akan digunakan dalam perubahan bobot layar dibawahnya (unit Z_j). Hitung suku perubahan bobot W_{jk} (yang akan dipakai nanti untuk merubah bobot W_{jk}) dengan laju percepatan α . Hitung juga koreksi bias (yang akan digunakan untuk memperbaiki nilai W_{jk}). Dan akhirnya δ_k dikirimkan ke unit-unit yang ada dilapisan bawahnya. Selanjutnya hitung faktor δ unit tersembunyi Z_j berdasarkan kesalahan di setiap unit. Setelah itu hitung informasi *error*, hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai V_{1j}) dan juga hitung koreksi bias (digunakan untuk memperbaiki nilai V_{0j}).
9. Kemudian tahap terakhir dalam proses pelatihan ini adalah mempengaruhi bobot dan bias dimana Semua perubahan bobot akan dihitung termasuk perubahan bobot garis yang menuju ke unit keluaran dan perubahan bobot garis yang menuju ke unit tersembunyi.
10. Apabila tahap BPNN telah selesai dikerjakan maka akan terjadinya perulangan dengan $i = i + 1$ yang digunakan sebagai acuan pada kondisi sebelumnya dan secara otomatis kondisi $i \leq \text{jumlahpola}$ akan menjadi "T" karena nilai i sudah tidak lebih kecil sama dengan jumlahpola, kemudian hitung *error*

dengan rumus yang ada. Setelah *error* dihitung maka akan masuk kedalam kondisi pertama yaitu $Epoch\ max < Epoch$ dan $Target\ kesalahan > error$. Jika *error* lebih kecil atau sama dengan target kesalahan dan atau *epoch* lebih besar atau sama dengan *epoch* maksimum maka bobot yang terbentuk disimpan dalam suatu file bobot dan jika sebaliknya maka akan kembali melakukan *epoch* hingga kondisi sesuai dengan apa yang diinginkan. Adapun, *flowchart* BPNN dapat dilihat pada Gambar 2.



Gambar. 2. Flowchart BPNN

F. Performa Akurasi

Beberapa metode dalam statistik untuk melakukan pengukuran akurasi suatu algoritma seperti *mean absolute error* (MAE), *mean square error* (MSE), *root mean squared error* (RMSE), dan *mean absolute percentage error* (MAPE). Pengukuran algoritma bertujuan untuk mendapatkan nilai terbaik (Haviluddin, Alfred, Obit, Hijazi, & Ibrahim, 2015; Rojas & Rojas, 2011; Susanti et al., 2018). Dalam penelitian ini, metode MSE dipilih untuk mengukur akurasi dengan menggunakan persamaan (1).

$$MSE = \frac{1}{M} \sum_{t=1}^M (x_t - \hat{x}_t)^2 \tag{1}$$

Dimana, x_t adalah nilai data; $x_t - \hat{x}_t$ adalah nilai hasil; M adalah nilai pola.

G. Sampel Pola Karakter

Jenis data yang digunakan dalam penelitian ini adalah data primer. Data primer merupakan data yang berasal dari sumber asli atau pertama. Data ini tidak tersedia dalam bentuk terkompilasi ataupun dalam bentuk file-file. Data ini harus dicari melalui narasumber atau dalam istilah teknisnya responden, yaitu orang yang kita jadikan objek penelitian atau orang yang kita jadikan sarana mendapatkan informasi ataupun data (Nurmila, Sugiharto, & Sarwoko, 2016; Widoretno, Sarosa, & Muslim, 2013). Sampel data dari 23 aksara Lontara Bugis Makassar dapat dilihat pada Tabel 1.

Tabel 1. Sampel Pola Karakter

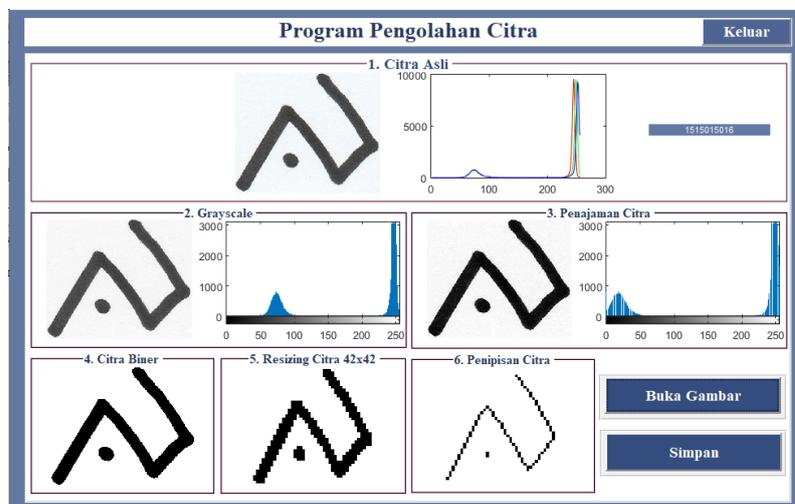
Aksara Lontara	Latin	Aksara Lontara	Latin	Aksara Lontara	Latin
	Ka		Ta		Ya
	Ga		Da		Ra
	Nga		Na		La
	Ngka		Nra		Wa
	Pa		Ca		Sa
	Ba		Ja		A
	Ma		Nya		Ha
	Mpa		Nca		

III. Hasil dan Pembahasan

Bagian ini adalah menerangkan hasil pengolahan citra, ekstraksi ciri yang dilanjutkan dengan pengujian menggunakan metode *Backpropagation Neural Network* (BPNN).

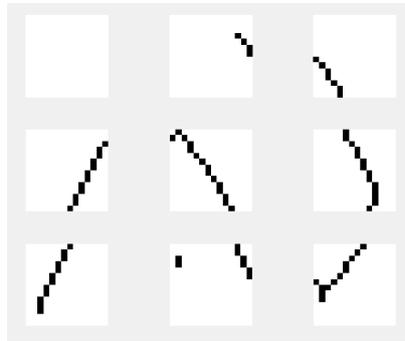
A. Hasil Pengolahan Citra

Pengolahan citra adalah pemrosesan citra yang bertujuan agar kualitas citra menjadi lebih baik dan sesuai dengan keinginan pemakai sehingga mudah diinterpretasi oleh manusia atau mesin. Berdasarkan tahapan proses pengolahan citra (Gambar 1), Tahap Pertama adalah proses cropping seperti akuisisi citra (*convert* citra asli menjadi citra *grayscale*), peningkatan kualitas citra, dan perbaikan kualitas citra (*penajaman citra*) telah dilakukan. Tahap kedua, proses segmentasi citra dengan mengubah citra *grayscale* yang telah ditajamkan diubah menjadi citra biner menggunakan metode *threshold*. Setelah itu dilakukan ekstraksi ciri ukuran dengan *resizing* menjadi citra berukuran 42 x 42 dan penipisan citra. Tahap ketiga adalah proses ekstraksi ciri untuk mendapatkan data yang dibutuhkan. Adapun, hasil proses cropping dapat dilihat pada Gambar 3.



Gambar. 3. Proses Pengolahan Citra

Selanjutnya, Tahap keempat adalah proses segmentasi. Pada penelitian ini, metode *Cheesboard* yang membagi citra menjadi objek-objek seperti papan catur telah digunakan. Objek-objek yang terbentuk berupa persegi dengan ukuran tertentu. Pada penelitian ini citra dibagi menjadi 9 segmen yang mana akan digunakan IoC untuk menghitung jumlah nilai piksel hitam (*black*) pada masing-masing segmen atau bagian. Hasil segmentasi dapat dilihat pada Gambar 4.



Gambar. 4. Segmentasi dengan membagi Citra menjadi 9 segmen

Tahap terakhir adalah ekstraksi ciri pada masing-masing segmen dengan menggunakan Mark Direction untuk menghitung beberapa nilai yang mencakup masking vertikal (*vert*), horizontal (*horz*), diagonal kiri (*dig1*) dan diagonal kanan (*dig2*). Proses ekstraksi ciri dilakukan pada data sebanyak 16 x 23 huruf yang kemudian terbentuk 8 variasi ciri pada masing-masing huruf aksara Lontara Bugis Makassar. Ciri inilah yang akan digunakan sebagai *input* pada jaringan. Untuk jumlah data yang digunakan dapat dilihat pada Tabel 2.

Tabel 2. Ekstraksi Ciri

Ciri	Kombinasi	Jumlah Data Ciri
1	[<i>Black</i>]	9
2	[<i>dig1</i>]	9
3	[<i>dig2</i>]	9
4	[<i>Black + dig1</i>]	$9 + 9 = 18$
5	[<i>Black + dig2</i>]	$9 + 9 = 18$
6	[<i>horz + vert</i>]	$9 + 9 = 18$
7	[<i>dig1 + dig2</i>]	$9 + 9 = 18$
8	[<i>Black + dig1 + dig2 + horz + vert</i>]	$9 + 18 + 18 = 45$

Dalam percobaan ini, hasil dari proses ekstraksi ciri disimpan ke dalam dataset dengan ekstensi file yaitu .xlsx. Sehingga, kumpulan data ciri dari ciri ke-1 hingga ciri ke-8 untuk dianalisa dengan metode BPNN sebagai data training dan testing telah digunakan.

B. Perhitungan Citra dengan Metode Backpropagation Neural Network (BPNN)

Dalam percobaan ini, arsitektur jaringan BPNN yang digunakan terdiri dari *input layer*, *hidden layer* dan *output layer*. Jumlah dari *hidden layer* dapat diubah sesuai kebutuhan. *Input layer* terdiri dari 9, 18 dan 45 *neuron* yang merupakan kombinasi ciri terdiri dari 8 ciri berbeda yang telah diperoleh dari proses ekstraksi ciri. Sedangkan *output layer* terdiri dari 23 *neuron* karena target *output* yang akan dicapai merupakan jumlah dari huruf aksara Lontara yang terdiri dari 23 huruf. Pelatihan jaringan dengan metode BPNN menggunakan 230 data latih dan 138 data uji. Pada percobaan ini, pelatihan jaringan dengan metode BPNN dipengaruhi oleh *learning rate*, *epoch* dan jumlah *neuron* pada *hidden layer*. Simulasi dilakukan 6 kali dengan 3 parameter yang berbeda yaitu pengaruh *learning rate*, kemudian jumlah *epoch* dengan batas maksimum dan *hidden layer*. Variasi masing-masing parameter dapat dilihat pada Tabel 3.

Tabel 3. Parameter pengujian BPNN

Parameter	Nilai
<i>Learning rate</i>	0.02; 0.03; 0.04
<i>Epoch</i>	10; 25; 50; 5000
<i>Hidden layer</i>	90; 120
Fungsi pembelajaran	trainlm

Berdasarkan simulasi yang dilakukan sebanyak 6 kali, simulasi ke-4 merupakan simulasi terbaik pada perhitungan menggunakan metode BPNN dengan satu *hidden layer* dengan *neuron* berjumlah 90 dengan fungsi aktivasi *tansig*, jumlah *epoch* maksimum adalah 50 dan target kesalahan adalah 0. Fungsi pembelajaran yang digunakan adalah trainlm dan jumlah lapisan keluaran terdapat 32 *neuron*. *Learning rate*

yang diberikan pada simulasi ini yaitu 0.02. Pola jaringan ini akan diterapkan pada kombinasi ciri 1 sampai 8. Hasil simulasi ke-3 dapat dilihat pada Tabel 3.

Tabel 4. Hasil simulasi 3 pada jaringan BPNN

Kombinasi Ciri	Hidden layer	Data Uji	Data Benar	Akurasi	Performa	Best Epoch	Waktu
Ciri 1	90	138	88	63.76%	0,00478	50	03:53
Ciri 2	90	138	44	31.88%	0,00666	50	03:48
Ciri 3	90	138	65	47.10%	0,00522	50	03:52
Ciri 4	90	138	103	74.63%	0,00475	50	06:07
Ciri 5	90	138	101	73.18%	0,00478	50	06:07
Ciri 6	90	138	35	25.36%	0,00891	50	06:09
Ciri 7	90	138	66	47.82%	0,00488	50	05:26
Ciri 8	90	138	105	76.08%	0,00424	50	16:56

Tabel 4 dapat dilihat bahwa dari 138 huruf terdapat pengenalan huruf paling banyak adalah 105 huruf yang dapat terbaca dengan baik dan tepat dengan akurasi sebesar 76.08% menggunakan kombinasi ciri 8. Adapun, kombinasi ciri 8 dihitung menggunakan MSE yang menghasilkan nilai terbaik sebesar 0.00424. Pembelajaran berhenti pada best *epoch* 50 sesuai dengan penetapan jumlah *epoch* pada simulasi ini. Namun waktu pembelajaran yang dihabiskan dapat dikatakan cukup lama yaitu 16 menit 56 detik.

C. Analisa metode BPNN

Variasi *learning rate* pada metode BPNN dengan menggunakan fungsi *trainlm* yang terdapat pada simulasi ke-1 dan ke-2 tidak berpengaruh terhadap perubahan akurasi. Secara umum hasil MSE yang diperoleh oleh kedua simulasi tersebut tidak berubah terkecuali pada ciri 6 yang dapat dilihat pada simulasi ke-1 memiliki MSE sebesar 0.075 dan pada simulasi ke-2 memiliki MSE sebesar 0.0108. Dan perubahan yang paling terlihat yaitu pada waktu yang dihabiskan pada proses pembelajaran. Berdasarkan hasil tersebut, parameter *learning rate* dapat dikatakan berpengaruh terhadap proses pembelajaran dalam penelitian ini meski perubahan hanya terlihat pada MSE dan waktu pembelajaran.

Pemberian *epoch* maksimum pada simulasi ke-3 dan ke-4 dapat dikatakan berpengaruh terhadap akurasi yang didapatkan karena kenaikan akurasi dapat dilihat dengan jelas pada ciri 2,3,4,5,6 dan ciri 8. Perubahan akurasi dapat dilihat pada Tabel 4.

Tabel 5. Peningkatan akurasi terhadap pengaruh *epoch* maksimum

Kombinasi Ciri	Akurasi Simulasi 3	Akurasi Simulasi 4
Ciri 1	64.49%	63.76%
Ciri 2	28.98%	31.88%
Ciri 3	44.20%	47.10%
Ciri 4	68.11%	74.63%
Ciri 5	71.01%	73.18%
Ciri 6	23.91%	25.36%
Ciri 7	47.82%	47.82%
Ciri 8	71.01%	76.08%

Apabila akurasi meningkat maka juga akan berpengaruh pada MSE yang di hasilkan. Seluruh pembelajaran pada variasi ini berhenti pada best *epoch* sesuai dengan *epoch* maksimum yang diberikan yaitu 10 *epoch* dan 50 *epoch*. Namun waktu yang di gunakan pada proses pembelajaran juga meningkat secara keseluruhan. Semakin besar *epoch* yang diberikan semakin lama waktu yang dihabiskan untuk proses pembelajaran. Jadi *epoch* maksimum pada penelitian ini juga berpengaruh besar terhadap MSE dan waktu pembelajaran. Namun jumlah perubahan dari unit *neuron* sendiri tidak memberi peningkatan yang dapat melampaui nilai tertinggi dari nilai default atau titik acuan yakni simulasi ke-5 dengan akurasi tertinggi 76.08%.

Dalam percobaan ini, hasil terbaik metode BPNN dalam mengenali suatu pola dapat dilihat pada Tabel 4, dengan memperhatikan banyaknya huruf yang terbaca diikuti waktu pembelajaran tersingkat terdapat pada simulasi ke-4 sebanyak 105 huruf dari 138 huruf dengan akurasi sebesar 76.08%. MSE 0.00424, best *epoch* 50 dengan waktu pembelajaran 16 menit 56 detik.

IV. Kesimpulan

Analisa pengenalan pola aksara Lontara Bugis Makassar menggunakan metode BPNN telah diimplementasikan. Berdasarkan hasil percobaan, metode BPNN memiliki tingkat akurasi terbaik sebesar 76.08% yang diperoleh dari simulasi ke-4 ciri 8 dengan data yang dapat dikenali sebanyak 105 data dari total

data *input* sebanyak 138 data. Sedangkan, waktu pengujian yang dibutuhkan adalah selama 16 menit 56 detik. Dapat dikatakan bahwa, parameter seperti *learning rate*, jumlah *neuron* pada *hidden layer* dan banyaknya *epoch* maksimum, sangat mempengaruhi hasil data yang dapat dikenali dan tingkat akurasi hasil pengenalan. Dalam penelitian ini, parameter terbaik dari metode BPNN yaitu *learning rate* = 0.02, jumlah *neuron* pada *hidden layer* = 90, banyak *epoch* = 50 telah digunakan untuk mendapatkan akurasi yang baik. Sehingga, metode BPNN telah menghasilkan tingkat akurasi pengenalan yang diukur dengan MSE sebesar 0.00424.

Daftar Pustaka

- Abidin, A. Z. (2006). Notes on the Lontara' as Historical Sources. *Indonesia*. <https://doi.org/10.2307/3350664>
- Ahmad, A. A. (2014). MELESTARIKAN BUDAYA TULIS NUSANTARA: Kajian tentang Aksara Lontara. *Jurnal Budaya Nusantara*.
- Dharwal, R., & Kaur, L. (2016). Applications of Artificial Neural Networks: A Review. *Indian J. Sci. Technol*, 9 No. 47, 1–8.
- Drs. Mattulada. (1974). *Bugis-Makassar Manusia dan Kebudayaan* (Nomor 16). BERITA ANTROPOLOGI.
- Fatta, H. al. (2009). *Rekayasa Sistem Pengenalan Wajah*. Yogyakarta: ANDI OFFSETYOGY.
- Gaffar, E. U. A., Gani, I., Haviluddin, Gaffar, A. F. O., & Alfred, R. (2019). A Heuristic Network for Predicting the Percentage of Gross Domestic Product Distribution. *Proceeding - 2018 International Symposium on Advanced Intelligent Informatics: Revolutionize Intelligent Informatics Spectrum for Humanity, SAIN 2018*. <https://doi.org/10.1109/SAIN.2018.8673349>
- Graves, A. (2011). Practical Variational Inference for Neural Networks. *Advances in Neural Information Processing Systems*, 2348–2356.
- Hagan, M. T., & Menhaj, M. B. (1994). Training Feedforward Networks With The Marquardt Algorithm. *IEEE Transaction on Neural Networks*, 5(6), 989 – 993.
- Haviluddin, H., Alfred, R., Obit, J. H., Hijazi, M. H. A., & Ibrahim, A. A. A. (2015). A Performance Comparison of Statistical and Machine Learning Techniques in Learning Time Series Data. *Advanced Science Letters*, Vol. 21(Number 10, October 2015), 3037–3041. <https://doi.org/10.1166/asl.2015.6490>
- Kurniawan, F., Pamungkas, Z. S., Wibawa, A. P., & Haviluddin. (2016). Peramalan Forex Syariah Menggunakan Jaringan Saraf Tiruan Backpropagation. *Seminar Nasional Riset Ilmu Komputer (SNRIK 2016)*, 302–305.
- Kurniawan, F., Pamungkas, Z. S., Wibawa, A. P., Haviluddin, H., Graves, A., Dharwal, R., ... Pauzi, G. A. (2015). Pengenalan karakter sandi rumput pramuka menggunakan jaringan saraf tiruan dengan metode backpropagation. *Jurnal Teknologi Informasi*.
- Lehtokangas, M. (1999). Modelling with constructive backpropagation. *Neural Networks*. [https://doi.org/10.1016/S0893-6080\(99\)00018-0](https://doi.org/10.1016/S0893-6080(99)00018-0)
- Nurmila, N., Sugiharto, A., & Sarwoko, E. A. (2016). ALGORITMA BACK PROPAGATION NEURAL NETWORK UNTUK PENGENALAN POLA KARAKTER HURUF JAWA. *JURNAL MASYARAKAT INFORMATIKA*. <https://doi.org/10.14710/jmasif.1.1.1>
- Rahman, N. (2008). Agama, Tradisi dan Kesenian dalam Manuskrip La Galigo. *Sari*.
- Rojas, R., & Rojas, R. (2011). The Backpropagation Algorithm. In *Neural Networks*. https://doi.org/10.1007/978-3-642-61068-4_7
- Susanti, A., Suhartono, Setyadi, H. J., Taruk, M., Haviluddin, & Widagdo, P. P. (2018). Forecasting Inflow and Outflow of Money Currency in East Java Using a Hybrid Exponential Smoothing and Calendar Variation Model. *Journal of Physics: Conference Series*, 979, 012096. <https://doi.org/10.1088/1742-6596/979/1/012096>
- Widoretno, S., Sarosa, M., & Muslim, M. A. (2013). Implementasi Pengenalan Karakter Seseorang Berdasarkan Pola Tulisan Tangan. *Jurnal EECCIS*.