

PENERAPAN ALGORITMA *MODIFIED DIRECTION FEATURE* (MDF) DAN *SUPPORT VECTOR MACHINE* (SVM) PADA SISTEM PENGENALAN AKSARA LONTARA

Asyraful Insan Asry¹

¹Politeknik ATI Makassar

asyraful@atim.ac.id¹

ABSTRAK

Sistem pengenalan aksara Lontara Bugis Makassar pada penelitian ini menggunakan metode *Modified Direction Feature* (MDF) pada ekstraksi fitur dan metode *Support Vector Machine* (SVM) pada klasifikasi. Data masukan berupa gambar dengan ukuran 150x120 piksel, dimana jumlah data uji 25 dan jumlah data latih 50. Pada proses klasifikasi, SVM menggunakan nilai C sebagai parameter penentu *hyperplane* terbaik dan nilai kernel (σ) pada *Radial Basis Function* (RBF) yang menjadi variable. Nilai C yang digunakan dari 1 sampai 4 dan Nilai σ adalah 1 sampai 16. Hasil penelitian menunjukkan bahwa akurasi tertinggi yang diperoleh sebesar 84% dengan proses *thinning* pada tahap preprocessing untuk kondisi C = 4 dan $\sigma = 10$.

Kata kunci: Pengenalan pola, aksara lontara bugis makassar, *thinning*, ekstraksi fitur, *modified direction feature*.

ABSTRACT

Aksara Lontara Bugis Makassar recognition system using Modified Direction Feature (MDF) method as feature extraction and Support Vector Machine (SVM) method as Classification. Data input using image with size 150x120 pixels. The number of data test 25 and data training 50 images of word aksara lontara. Classification method using C value parameter as the best hyperplane and kernel value (σ) parameter with radial basis function (RBF). The Value of C from 1 to 4 and the kernel value from 1 to 16. The result of this research, best accuracy values is 84% using thinning process method in step preprocessing with the value of C is 4 and $\sigma = 10$.

Keywords: Pattern recognition, aksara lontara bugis makassar, thinning, feature extraction, modified direction feature.

PENDAHULUAN

Penerapan teknologi sebagai alat bantu komunikasi berkembang sangat pesat termasuk dalam penggunaan kamus untuk menterjemahkan suatu bahasa. Salah satu bidang teknologi yang diterapkan pada sistem penerjemah yaitu pada bidang pengenalan pola yang menjadi dasar dari teknologi *Optical Character Recognition* (OCR). Teknologi OCR berguna untuk mengkonversi gambar ke tulisan sehingga dapat diolah oleh komputer untuk dapat diterjemahkan[1]. Saat ini, tidak hanya tulisan latin yang dapat dikenali namun telah dilakukan penelitian untuk jenis tulisan yang lain, seperti tulisan arab[2], kanji-Jepang[3], kanji-China hingga aksara lontara.

Aksara Lontara adalah aksara tradisional masyarakat Bugis-Makassar di daerah Sulawesi Selatan, Indonesia yang merupakan bagian dari warisan budaya. Untuk melestarikan tulisan bahasa daerah dilakukan upaya pengajaran di bidang pendidikan dengan menerapkan unsur teknologi sebagai bahan ajar yang interaktif dan komunikatif, sehingga meningkatkan minat siswa untuk mempelajarinya. Sistem penerjemah termasuk media ajar di bidang pendidikan untuk membantu memahami makna dan arti tulisan suatu bahasa. Dalam bidang pariwisata membantu untuk memperkenalkan tulisan aksara Lontara dan penerapannya untuk membantu mendokumentasikan naskah yang bertuliskan aksara Lontara.

Pengenalan aksara lontara Bugis-Makassar yang merupakan salah satu warisan budaya di Indonesia telah diterapkan pada teknologi pengenalan pola dengan menggunakan metode JST *backpropagation* untuk mengklasifikasi bentuk setiap huruf

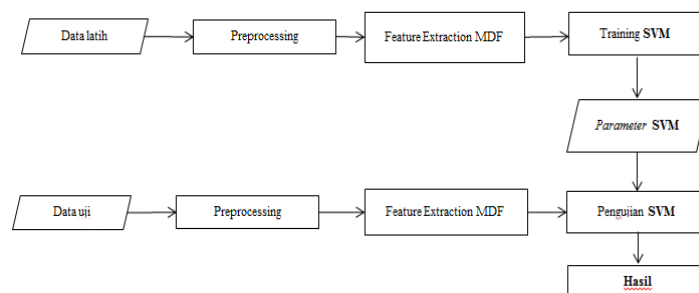
aksara Lontara. Proses penelitian dilakukan dengan memasukkan data citra aksara tiap karakter. Pada tahap pra proses dengan menggunakan metode Otsu dan dilanjutkan tahap normalisasi pada ekstraksi fitur. Metode jaringan syaraf tiruan Backpropagation diterapkan untuk klasifikasi nilai vektor dari tahapan normalisasi. Hasil dari penelitiannya dari sejumlah percobaan yang dilakukan untuk mengenali huruf-huruf Bugis-Makassar, sebanyak 19 percobaan hanya dua diantaranya yang menghasilkan perilaku *Mean Square Error* (MSE) yang tidak konvergen [4].

Selain itu, penelitian sistem penerjemah bahasa Indonesia ke aksara lontara telah dilakukan dengan menggunakan metode binary search dan parsing tree. Pada tahap awal dilakukan pembacaan *string* per karakternya dan kata tersebut langsung dibandingkan terhadap kosakata contohnya masukan kata yaitu “manre”, dimana kata tersebut dibentuk dari susunan huruf m-a-n-r-e yang diawali dengan huruf “m” yang ada pada database, selanjutnya karakter “ma” diubah ke lontara bugis menjadi “m” dan karakter “nre” diubah menjadi huruf “eR”. Dari hasil penelitian tersebut bahwa kata berawalan “B” memiliki nilai propabilitas paling kecil yaitu 0.0085 dengan jumlah kata yang berawalan huruf B sebanyak 113 kosakata dari seluruh jumlah kosakata sebanyak 702 kosakata[5]. Untuk membantu memperkenalkan kosa kata bahasa Indonesia dan bahasa bugis beserta penulisannya kepada masyarakat. Namun, saat ini belum ada penerjemah aksara lontara berbasis gambar, khususnya untuk aksara Lontara Bugis-Makassar[6], sehingga penelitian ini dilakukan proses klasifikasi kata pada sistem terjemahan dengan metode SVM.

Penelitian ini proses klasifikasi dengan SVM dilakukan untuk mengenali jenis kata aksara berbasis gambar, dimana setiap gambar kata memiliki nilai fitur yang diperoleh dari proses ekstraksi fitur dengan metode MDF. Sebelumnya, proses awal dilakukan pada tahap preprocessing untuk mendapatkan nilai piksel yang membentuk pola objek setiap gambar kata aksara.

METODE PENELITIAN

Penelitian ini proses klasifikasi kata aksara Lontara dilakukan 2 proses yaitu training dan pengujian, pada proses *training* dilakukan untuk mendapatkan parameter terbaik dari hasil *training* seluruh nilai fitur yang didapatkan dari proses ekstraksi fitur dengan metode MDF. Ketika proses *training* telah dilakukan, parameter terbaik yang diperoleh dari proses tersebut digunakan pada proses pengujian pada saat klasifikasi data uji. Alur diagram dapat dilihat pada gambar 1



Gambar 1. Alur diagram sistem

Data yang digunakan pada penelitian ini merupakan gambar kata aksara Lontara dengan format .JPG dengan ukuran 150 x 120 piksel. Tulisan kata aksara Lontara menggunakan font BugisA.ttf dengan ukuran 14 pt. Kata yang digunakan untuk data latih dan data uji yaitu kata kerja dengan dua suku kata. Data uji yang digunakan diperlihatkan pada tabel 1.

Tabel 1. Kata kerja aksara lontara

No	Kata aksara	Arti	No	Kata aksara	Arti	No	Kata aksara	Arti
1	Jamma	Adu	11	Walek	Balas	21	Gerrak	Bentak
2	Bissa	Basuh	12	Jallo	Amuk	22	Engka	Ada
3	Akka	Angkat	13	Tahang	Tahan	23	Layya	Takut
4	Tunu	Bakar	14	Bici	Bisik	24	Lingu	Bingung
5	Dince	Tancap	15	Baji	Baik	25	Sare	Beri
6	Giling	Balik	16	Lannyi	Bersih			
7	Mala	Ambil	17	Pau	Bicara			
8	Lurang	Angkut	18	Ekbu	Buat			
9	Tojang	Ayun	19	Bampa	Pukul			
10	Era	Ajak	20	Manre	Makan			

Preprocessing

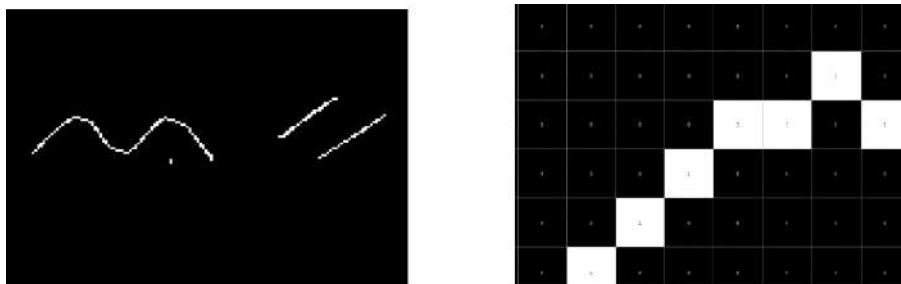
Pada tahap ini dilakukan proses *grayscale*, binerisasi dan *thinning*. *Grayscale* adalah proses mengubah nilai piksel gambar RGB ke nilai piksel skala keabuan berdasarkan persamaan :

$$\text{Gray} = wR + wG + wB \tag{1}$$

dimana wR , wG dan wB adalah nilai dari masing-masing bobot sistem elemen warna merah, hijau dan biru secara berurut. *National Television System Committee* (NTSC) mendefinisikan nilai bobot untuk konversi citra warna ke skala keabuan dengan nilai 0.289, 0.5870 dan 0.1140. Setelah dilakukan proses *grayscale* nilai setiap piksel pembentuk objek gambar aksara Lontara dirubah menjadi nilai piksel berkisar 0 – 255. Nilai *grayscale* pada setiap piksel dirubah menjadi nilai biner pada proses binerisasi dengan nilai ambang (T) sebesar 210 dan berdasarkan persamaan (2) berikut.

$$B(x,y) = \begin{cases} 1 & \text{if } G(x,y) \leq T \\ 0 & \text{if } G(x,y) \geq T \end{cases} \tag{2}$$

Selanjutnya dilakukan proses *thinning* untuk mereduksi nilai piksel pada objek citra tanpa menghilangkan bentuk asli dari objek citra seperti diperlihatkan pada gambar 2.



Gambar 2. Hasil proses *thinning* dari gambar biner

Ekstraksi Fitur

Setelah dilakukan proses *thinning* pada tahap preprocessing maka dilanjutkan ke tahap ekstraksi fitur dengan metode *Modified direction Feature*[7]. Metode ini menggabungkan nilai *direction feature* dan *transition feature*. Citra biner hasil dari tahap proses *thinning* pada preprocessing, memiliki nilai piksel 1 untuk objek yang berwarna putih sebagai *foreground* dan nilai piksel 0 untuk *background*. Setiap piksel bernilai 1 diberikan label arah pada masing-masing piksel sesuai posisi piksel pada gambar aksara Lontara. Gambar 3 ditampilkan label arah setiap piksel pembentuk objek berdasarkan posisi setiap piksel.

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	2	0
0	0	0	0	3	4	4	0	4
0	0	0	3	0	0	0	0	0
0	0	3	0	0	0	0	0	0
0	3	0	0	0	0	0	0	0

Gambar 3. Nilai label arah setiap piksel pembentuk objek

Setelah pemberian nilai arah pada setiap piksel pembentuk objek, selanjutnya menghitung nilai TF (*Transition Feature*) dengan rumus $TF = 1 - (\text{posisi piksel}/\text{Lebar citra})$. Hasilnya ditampilkan pada gambar dibawah ini. Untuk pemberian nilai DF (*Direction Feature*) dilakukan dengan rumus $DF = \text{Nilai arah}/10$. pada gambar 4 ditampilkan nilai TF dan DF.

Pemberian nilai TF dan DF dilakukan dari empat arah gambar objek yaitu dari kiri ke kanan, atas ke bawah, kanan ke kiri dan bawah ke atas. Setelah mendapatkan nilai TF dan DF di setiap arah objek gambar, dilakukan normalisasi nilai TF dan DF di setiap arah dengan ukuran matriks 5x3.

Setelah dilakukan normalisasi nilai TF dan DF, dilakukan penggabungan nilai TF dan DF dari semua arah, dan menghasilkan panjang nilai vector $5 \times 3 \times 2 \times 4 = 120$, 5x3 adalah ukuran matriks dan angka 2 adalah nilai dari TF dan DF dan angka 4 adalah jumlah arah pengambilan nilai TF dan DF, yaitu arah kiri-kanan, kanan-kiri, atas-bawah dan bawah-atas.

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0.97	0
0	0	0	0	0.96	0.95	0.95	0	0.95
0	0	0	0.96	0	0	0	0	0
0	0	0.96	0	0	0	0	0	0
0	0.96	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0.2	0
0	0	0	0	0.3	0.4	0.4	0	0.4
0	0	0	0.3	0	0	0	0	0
0	0	0.3	0	0	0	0	0	0
0	0.3	0	0	0	0	0	0	0

Gambar 4. (a) Nilai TF (b) Nilai BF

Klasifikasi

Proses klasifikasi dilakukan setelah proses ekstraksi fitur pada setiap data latih, nilai fitur yang dihasilkan pada semua data latih digunakan sebagai data input pada proses training dengan metode SVM (*Support Vector Machine*). Proses metode SVM bertujuan untuk mendapatkan hyperplane terbaik agar mendapatkan jalur pemisah dari dua kelas yang berbeda[8].

Data latih yang telah melewati proses sebelumnya akan dinyatakan sebagai (x_i, y_i) , dengan $i = 1, 2, 3, \dots, N$, yang berarti N adalah banyaknya data dan dinotasikan dengan model $x_i = \{x_{i1}, x_{i2}, x_{i3}, \dots, x_{iq}\}^T$ yang merupakan atribut untuk data latih ke-i. Untuk pelabelan kelas dinotasikan dengan $y_i \in \{-1, +1\}$ dimana $i = 1, 2, 3, \dots, N$ yang merupakan kelas dari sampel data yang terpisah dengan fungsi pemisah (*hyperplane*), dinotasikan pada persamaan (3).

$$f(x) = w x + b \tag{3}$$

dari notasi hyperlane tersebut diperoleh persamaan

$$[(w \cdot x_i) + b] \geq 1 \text{ untuk } y_i = +1$$

$$[(w \cdot x_i) + b] \leq -1 \text{ untuk } y_i = -1$$

Untuk menentukan kelas dari data testing, ditentukan dengan nilai fungsi berikut.

$$F(x_t) = \sum_{s=1}^{ns} a_s y_s x_s \cdot x_t + b \tag{4}$$

Dengan x_t = data yang akan diprediksi kelasnya (data testing), x_s = data support vector, $s = 1, 2, \dots, ns$ dan ns = banyak support vector. Untuk mengatasi data nonlinier, digunakan fungsi kernel RBF (Radial Basis Function) dengan persamaan berikut.

$$K(u, v) = \exp(-\gamma \|u - v\|^2), \gamma > 0 \tag{5}$$

Untuk mengetahui tingkat akurasi pengujian dilakukan dengan membagi data yang benar dengan jumlah data uji seluruhnya dan dikalikan dengan nilai 100.

$$Akurasi = \frac{\text{jumlah data benar}}{\text{jumlah seluruh data}} \times 100\% \tag{6}$$

HASIL DAN PEMBAHASAN

Setelah dilakukan proses pengujian dengan menggunakan jumlah data latih sebanyak 25 data dan data uji sebanyak 50 data dengan menggunakan parameter C dan parameter sigma pada kernel RBF. Proses ini juga membandingkan pengaruh proses thinning dan proses dilasi pada tahap preprocessing. Dihilangkan akurasi terbaik pada parameter C = 4 dan sigma = 10 dengan akurasi 84% dengan proses thinning seperti pada tabel 2

Tabel 2. Nilai akurasi proses *thinning*

Sigma	C			
	1	2	3	4
1	52 %	40 %	56 %	52%
2	64 %	56 %	72 %	56 %
4	72%	76 %	80 %	72 %
6	76 %	76 %	80 %	80 %
8	76 %	80 %	84 %	84 %
10	73 %	80 %	84 %	84 %
12	68 %	72 %	80 %	80 %
14	68 %	68 %	76 %	76 %
16	64 %	86 %	72 %	76 %

Pada tabel 3 ditampilkan hasil percobaan dilakukan tanpa proses *thinning* memiliki nilai akurasi 80% sebagai nilai terbaik.

Tabel 3. Nilai akurasi tanpa proses *thinning*

Sigma	C			
	1	2	3	4
1	40 %	40 %	40 %	40 %
2	64 %	56 %	44 %	56 %
4	72 %	72 %	76 %	72 %
6	76 %	76 %	80 %	80 %
8	76 %	80 %	80 %	80 %
10	72 %	80 %	76 %	76 %
12	68 %	72 %	76 %	76 %
14	68 %	68 %	76 %	76 %
16	64 %	68 %	72 %	72 %

KESIMPULAN

Proses Klasifikasi menggunakan SVM (*Support Vector Machine*) telah diterapkan pada kata aksara Lontara Bugis-Makassar dengan menggunakan kata kerja sebagai data penelitian. Jumlah data yang diteliti berjumlah 75 data dengan data latih sebanyak 50 dan data uji sebanyak 25. Proses klasifikasi digunakan parameter C dan parameter kernel RBF untuk mengetahui akurasi terbaik untuk mengenali setiap kata yang diuji. Setelah dilakukan percobaan pengaruh proses *thinning* terhadap proses klasifikasi data yang diuji, dihasilkan nilai akurasi semakin baik jika pada tahap preprocessing dilakukan proses *thinning* karena mempengaruhi nilai fitur sebagai data masukan pada tahap klasifikasi.

DAFTAR PUSTAKA

- [1] Mori, S., C.Y. Suen, and K. Yamamoto. 1992. "Historical Review of OCR Research and Development." *Proceedings of the IEEE* 80 (7): 1029–58. doi:10.1109/5.156468.
- [2] Lutf, Mohammed, Xinge You, Yiu-ming Cheung, and C.L. Philip Chen. 2014. "Arabic Font Recognition Based on Diacritics Features." *Pattern Recognition* 47 (2): 672–84. doi:10.1016/j.patcog.2013.07.015
- [3] Ino, Tadashi, Ryusuke Nakai, Takashi Azuma, Toru Kimura, and Hidenao Fukuyama. 2009. "Recognition and Reading Aloud of Kana and Kanji Word: An fMRI Study." *Brain Research Bulletin* 78 (4-5): 232–39. doi:10.1016/j.brainresbull.2008.11.008.
- [4] A.Alwi, 2010, Pengenalan Pola Huruf-huruf Lontara Bugis-Makassar dengan Menggunakan Jaringan Syaraf Tiruan Metode Backpropagation. Yogyakarta: MIPA Pascasarjana Universitas Gadjah Mada.
- [5] Nangi, J, dkk,2013,Sistem Aplikasi Kamus Penerjemah Bahasa Indonesia-Aksara Lontara Bugis dengan Menggunakan Metode Binary Search dan parsing Tree. Makassar: Fakultas Teknik Universitas Hasanuddin.
- [6] Mattulada. 1991a. Manusia dan Kebudayaan Bugis-Makassar dan Kaili di Sulawesi. *Jurnal Antropologi Sosial dan Budaya Indonesia* No. 43 Th. XV Januari-April 1991.

- [7] Liu, X., Y., Blumeinstein, M., Experimental Analysis of the Modified Direction Feature for Cursive Character Recognition. Australia: Griffith University, 2003.
- [8] Jayadeva, Khemchandani, R. And Chandra, S., Twin Support Vector Machines for Pattern Classification. Pattern Analysis And Machine Intelligence, 2007 Vol. 29, No. 5, hal 905-910.