

IMPLEMENTASI ALGORITMA PARTICLE SWARM OPTIMIZATION DALAMPENGENALAN HURUF BUGIS

Ramdhan Reza Wardana¹, Kurniati²

Fakultas Ilmu Komputer, Universitas Bina Darma

Email : ramdhanrrw99@gmail.com¹, kurniati6491@gmail.com²

ABSTRAK

Orang Bugis juga mempunyai kesastraan dari segi lisan atau pun tulisan. Macam-macam sastra tulis berkembang seiring dengan tradisi sastra lisan, hingga kini masih tetap dibaca dan disalin ulang. Majunya teknologi pengolahan gambar atau citra juga dapat dimanfaatkan didalam melestarikan budaya serta ilmu bangsa. Di antara penerapannya dapat digunakan didalam melestarikan huruf Bugis yang hingga kini hampir punah karena perkembangan zaman yang semakin modern. Meskipun diajarkan di sekolah dasar dan menengah, tetap saja minim penggunaan untuk kehidupan sehari-hari. Padahal pengguna huruf Bugis semakin bertambah banyak dan sedikit dari kalangan remaja yang mengenal huruf Bugis dengan jelas. Maka dari itu melalui penelitian ini menghadirkan solusi atau solution yaitu membangun dan mengimplementasikan huruf Bugis untuk dikenali melalui metode algoritma particle swarm organization, sehingga hasilnya sistem berhasil mengenali huruf-huruf bugis tersebut dalam bentuk karakter, kata dan teks dengan tingkat akurasi tertentu. Aplikasi yang dibuat melalui MATLAB berhasil menjadi program yang efektif untuk membantu membuat pengenalan huruf Bugis lebih efektif dan mudah didalam mengenal dan melestarikan huruf Bugis.

Kata Kunci: *Sastra Lisan, Bahasa Bugis, Algoritma particle swarm organization*

ABSTRACT

The Bugis also have literature from an oral or written perspective. Various kinds of written literature developed along with the oral literary tradition, until now it is still read and re-copied. The advancement of image or image processing technology can also be used in preserving the nation's culture and knowledge. Among its applications, it can be used in preserving the Bugis letters which until now are almost extinct due to increasingly modern times. Even though it is taught in primary and secondary schools, it is still minimally used for daily life. In fact, the number of users of Bugis letters is increasing and few teenagers know Bugis letters clearly. Therefore, this research presents a solution, namely building and implementing Bugis letters to be recognized through the particle swarm organization algorithm method, so that the results of the system are able to recognize these Buginese letters in the form of characters, words and text with a certain level of accuracy. The application created through MATLAB has succeeded in becoming an effective program to help make Bugis letter recognition more effective and easier in recognizing and preserving Bugis letters.

Keywords: *Oral literature, Buginese language, particle swarm organization algorithm*

1. PENDAHULUAN

Perkembangan aplikasi komputer secara relatif signifikan baik didalam segi hardware (perangkat keras) atau didalam segi software (perangkat lunak). Salah satu penerapan yang

terbentuk dari development software yang terasa sekali manfaatnya yakni aplikasi pengenalan huruf Bugis. Aplikasi pengenalan huruf ini dapat begitu menolong masyarakat didalam memahami huruf Bugis, mengingat di Indonesia terdapat beberapa jenis bahasa daerah seperti bahasa Jawa, Batak, Madura, maupun bahasa orang Bugis yang mempunyai huruf yang berbeda.

Orang Bugis juga mempunyai kesastraan dari segi lisan atau pun tulisan. Macam-macam sastra tulis berkembang seiring dengan tradisi sastra lisan, hingga kini masih tetap dibaca dan disalin ulang. Perpaduan antara tradisi sastra lisan dan tulis itu kemudian menghasilkan salah satu Epos Sastra Terbesar didunia Yakni Las Galigo yang naskahnya lebih panjang dari Epos Mahabharata [3].

Majunya teknologi pengolahan gambar atau citra juga dapat dimanfaatkan didalam melestarikan budaya serta ilmu bangsa. Di antara penerapannya dapat digunakan didalam melestarikan huruf Bugis yang hinggakini hampirpunah karena perkembangan zaman yang semakin modern . Huruf Bugis merupakan bagian dari Bahasa Bugis yang melekat pada kebudayaan masyarakat Bugis, suntuik saat ini hampir mengalami kepunahan. Adapun usahapemerintahan didalam melestarikan huruf Bugis punahnya mencakup tentang pendidikan disekolah dasar dan menengah. Minimnya menggunakan untuk kehidupan sehari-hari. Padahal pengguna huruf Bugis semakin bertambah banyak dan sedikit dari kalangan remaja yang mengenal huruf Bugis dengan jelas.

Sebelumnya ada beberapa penelitian yang melibatkan penggunaan *algoritma particle swarm optimization pararel* yakni oleh Nugroho (2015) untuk proses segmentasi pada karakter Aksara Jawa. Untuk mengatasi beban komputasi yang cukup berat pada proses segmentasi citra, maka diterapkan metode *parallel programing*, dan arsitektur komputer yaitu GPU dengan support CUDA dari NVIDIA. Hasil akhir dari penelitian ini dapat mengimplementasikan segmentasi citra digital dengan metode *Particle Swarm Optimization* pada Aksara Jawa. GPU CUDA mempercepat proses segmentasi citra Aksara Jawa.

Adapun penelitian lain yang menerapkan pengenalan citra yaitu pengenalan kata aksara Jawa melibatkan penggunaan algoritma K-Nearest neighbor oleh Mukhooyar (2015). Pada penelitian tersebut, algoritma K Nearest neighbor diterapkan untuk menentukan thresholding atau nilai kemiripan suatu citra dengan aksara Jawa. Hasil yang didapat yaitu sistem translasi aksara Jawa menjadi alfabet dan dapat digunakan oleh pemula yang ingin belajar untuk mengenal aksara Jawa.

Adapun perbandingan penelitian pada skripsi ini dengan penelitian lain adalah algoritma yang digunakan yaitu Algoritma Particle Swarm Optimization. Lalu dengan huruf Bugis sebagai objeknya, penelitian dilakukan untuk menerapkan algoritma PSO dalam pengenalan huruf Bugis dan tidak berfokus untuk membandingkan kinerja PSO pada jenis-jenis pemrosesan GPU dan CUDA seperti halnya pada penelitian oleh Nugroho (2015) dan Mukhooyar (2015)

Menurut Eberhart dan Kennedy (2016) *Algoritma Particle Swarm Optimization (PSO)* didalam sebuah konferensi jaringan syaraf di Perth, Australia. Algoritma Particle Swarm Optimization (PSO) merupakan teknik optimasi berbasis matlab yang akan mengenal huruf Bugis. Matlab merupakan bahasa pemrograman dikembangkan oleh The Mathworks . Bahasa pemrograman ini banyak digunakan untuk perhitungan numerik keteknikan, komputasi, simbolik , visualisasi , grafis, analisis data matematis, statistika, simulasi, pemodelan, dan desain GUI . Kedua metode tersebut diharapkan dapat menyelesaikan pengenalan huruf Bugis.

Objek pada penelitian ini yaitu huruf Bugis, yang memiliki permasalahan proses pembelajaran huruf Bugis yang kurang memadai. Maka dari itu melalui penelitian ini menghadirkan *solusi atau solution* yaitu membangun mengimplementasikan huruf Bugis untuk dikenali melalui algoritma *particle swarm organization*.

Berdasarkan permasalahan yang telah dirumuskan sebelumnya, maka penulis membatasi masalah didalam penelitian ini, untuk memudahkan pengenalan huruf Bugis dibutuhkan sebuah sistem komputer yang mampu membaca serta mengenalkan huruf tersebut. Maka dari itu munculah sebuah gagasan untuk membuat sebuah aplikasi pengenalan huruf Bugis dengan melibatkan penggunaan metode *Algoritma Particle Swarm Optimization*.

Dari latar belakang di atas, maka penulis memberi judul pada penelitian ini “**Implementasi Algoritma Particle Swarm Optimization Dalam pengenalan Huruf Bugis**”.

2. METODOLOGI PENELITIAN

2.1 Metode Penelitian

Menurut Sugiyono (2013) Metode penelitian pada dasarnya merupakan cara ilmiah untuk mendapatkan data dengan tujuan dan kegunaan tertentu. Berdasarkan hal tersebut terdapat empat kata kunci yang perlu diperhatikan yaitu cara ilmiah, data, tujuan dan kegunaan. Menurut Darmadi (2013), Metode penelitian adalah suatu cara ilmiah untuk mendapatkan data dengan tujuan kegunaan tertentu. Cara ilmiah berarti kegiatan penelitian itu didasarkan pada ciri-ciri keilmuan yaitu rasional, empiris, dan sistematis. Berdasarkan pemaparan di atas dapat disimpulkan bahwa metode penelitian adalah suatu cara ilmiah untuk memperoleh data dengan tujuan tertentu.

Metode yang digunakan oleh penulis adalah metode deskriptif. Metode deskriptif adalah suatu metode dalam meneliti status sekelompok manusia, suatu objek, suatu set kondisi, suatu sistem pemikiran, ataupun suatu peristiwa pada masa sekarang. Tujuan dari penelitian deskriptif ini adalah untuk membuat deskriptif, gambaran atau lukisan secara sistematis, actual dan akurat mengenai fakta-fakta, sifat-sifat serta hubungan antar fenomena yang diselidiki.

2.2 Metode Pengumpulan Data

Menurut Sugiyono (2013:224) metode pengumpulan data adalah langkah yang strategis sekali, sesuai tujuan utama dari penelitian yaitu memperoleh data. Di dalam penyusunan penelitian ini pengumpulan data yang dibutuhkan didalam pengujian pengenalan huruf dilakukan oleh penulis dengan melibatkan metode pengumpulan data sebagai berikut:

1) Pengamatan (*Observation*) .

Peneliti melakukan peninjauan pada itu di pustaka dan buku-buku.

2) Studi kepustakaan (*Literature*) .

Data didapat melalui studi pustaka (*literature*) yaitu dengan menelusuri bahan dari perpustakaan atau journal atau dunia maya internet serta buku yang berkaitan dengan object yang akan diteliti.

2.3 Metode Analisis Data

Menurut Septia & Matlab (2015) MATLAB (*Matrix Laboratory*) merupakan bahasa pemrograman tingkat tinggi yang dikembangkan oleh *MathWorks* dan dikhususkan untuk komputasi numerik, visualisasi, dan pemrograman. Dengan memanfaatkan MATLAB, pengguna dapat melakukan analisis data, mengembangkan algoritma, dan membuat model maupun aplikasi. Bahasa, *tools*, dan fungsi-fungsi *built-in* akan memudahkan pengguna untuk mengeksplorasi berbagai pendekatan dan memperoleh *solusi atau solution* dengan lebih cepat dibandingkan apabila melibatkan penggunaan *spread sheets* atau bahasa pemrograman tradisional, seperti C/C++ atau Java. MATLAB melibatkan penggunaan konsep *array/matrik* menjadi standard variabel elemennya tanpa perlu adanya pendeklarasian array seperti pada bahasa lainnya. Selain itu juga dapat dilakukan integrasi dengan aplikasi dan bahasa pemrograman eksternal seperti C, Java, .NET, dan *Microsoft Excel*. Penggunaan MATLAB meliputi bidang-bidang:

- 1) Matematika dan Komputasi
- 2) Pembentukan Algorithm
- 3) Akuisisi Data
- 4) Pemodelan, simulasi dan Pembuatan
- 5) *Prototype*
- 6) Analisis Data, Explorasi, dan Visualisasi
- 7) Grafik Keilmuan dan Bidang Rekayasa

3. HASIL DAN PEMBAHASAN

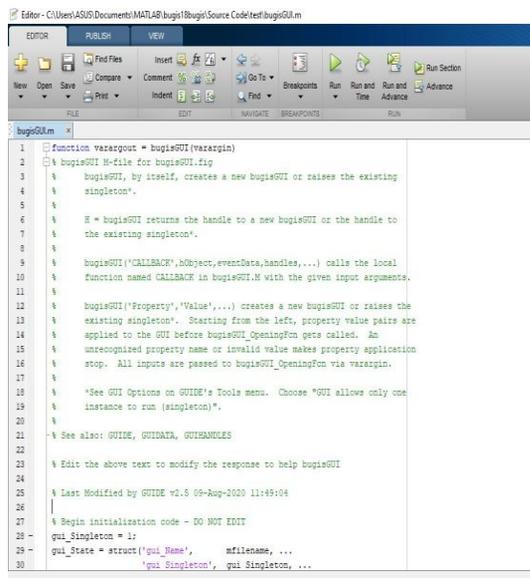
3.1 Hasil

Berdasarkan hasil penelitian dimana dilakukan di dalam meneliti implementasi algoritma *particle swarm optimization* (PSO) dalam pengenalan huruf bugis, maka didapat hasil akhir dari semua kegiatan dan rangkaian fase atau tahapan pengembangan *system* yang telah dilakukan merupakan perancangan dari berbagai desain atau rancangan yang telah dijelaskan pada bab terdahulu yang terdiri dari desain *file*, desain *masukan atau input* dan desain *keluaran atau out-put*. Program yang di gunakan didalam membuat program ini merupakan MATLAB R2013a.

Adapun pada tahapan ini akan di pembuatan tampilan dan di lakukan proses *coding* terhadap sistem yang akan dibuat. Adapun hasil dari tampilan sistem informasi yang telah di buat sebagai berikut :

Tampilan MatLab

Tampilan MatLab merupakan tampilan koding aplikasi implementasi algoritma *particle swarm optimization* (PSO) dalam pengenalan huruf bugis yang telah dibuat pada aplikasi MatLab. Berikut merupakan tampilan koding MatLab yaitu:



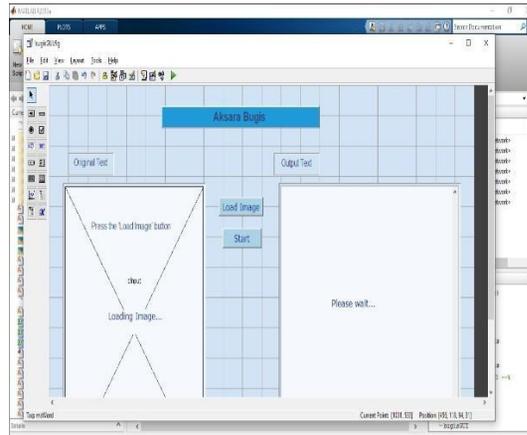
```
1 function varargin = bugisGUI(varargin)
2 % bugisGUI M-file for bugisGUI.fig
3 % bugisGUI, by itself, creates a new bugisGUI or raises the existing
4 % singleton*.
5 %
6 % H = bugisGUI returns the handle to a new bugisGUI or the handle to
7 % the existing singleton*.
8 %
9 % bugisGUI('CALLBACK', hObject,eventData,handles,...) calls the local
10 % function named CALLBACK in bugisGUI.M with the given input arguments.
11 %
12 % bugisGUI('Property','Value',...) creates a new bugisGUI or raises the
13 % existing singleton*. Starting from the left, property value pairs are
14 % applied to the GUI before bugisGUI_OpeningFcn gets called. An
15 % unrecognized property name or invalid value makes property application
16 % stop. All inputs are passed to bugisGUI_OpeningFcn via varargin.
17 %
18 % *See GUI Options on GUIDE's Tools menu. Choose *GUI allows only one
19 % instance to run (singleton)*.
20 %
21 % See also: GUIDE, GUIDATA, GUIHANDLES
22
23 % Edit the above text to modify the response to help bugisGUI
24
25 % Last Modified by GUIDE v2.5 09-Aug-2020 11:49:04
26 |
27 % Begin initialization code - DO NOT EDIT
28 gui_Singleton = 1;
29 gui_State = struct('gui_Name',       mfilename, ...
30                  'gui_Singleton',   gui_Singleton, ...
```

Gambar 1. Tampilan MatLab

Melalui tampilan ini terdapat beberapa menu yang dapat digunakan, antara lain Editor, Publish dan View. Pada editor dapat melakukan perintah seperti membuat, membuka, maupun menyimpan file coding baru pada matlab file opsi insert, comment ataupun indent disertai fitur pencarian kata atau sintaks tertentu. Lalu dapat pula menjalankan program melalui submenu Run.

Tampilan GUIDE MatLab

Tampilan GUIDE MatLab merupakan tampilan edit interface implementasi algoritma *particle swarm optimization* (PSO) dalam pengenalan huruf bugis melalui GUI Development Environment. Pada tampilan ini dapat menentukan bagaimana layout dan fungsional dari aplikasi pengenalan huruf bugis dengan algoritma *particle swarm optimization*. Berikut merupakan tampilan GUIDE MatLab yaitu:

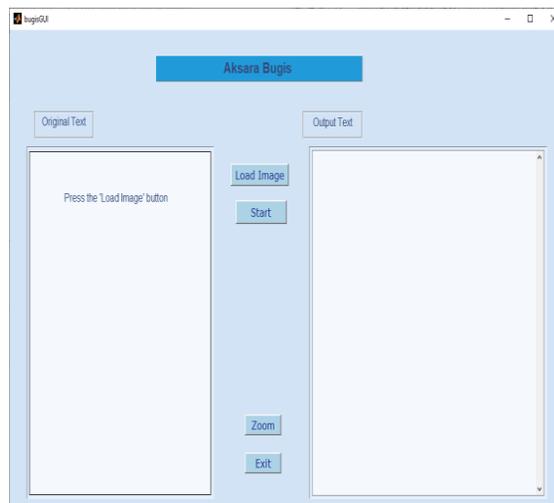


Gambar 2. Tampilan GUIDE MatLab

Pada guide ini, program yang ada dibuat sesuai dengan kebutuhan aplikasi pengenalan huruf bugis dengan algoritma *particle swarm optimization*. Interface aplikasi ini adapun melibatkan penggunaan image, button atau tombol dan disertai label untuk memperjelas fungsi tiap-tiap bagian. Kemudian, dilakukankustomisasi di masing-masing bagian tersebut baik ukuran komponen maupun warnanya pada interface.

Halaman Utama

Halaman utama merupakan halaman paling utama didalam aplikasi implementasi algoritma *particle swarm optimization* (PSO)dalam pengenalan huruf bugis. Pada halaman ini terdapat teks original dan teks keluaran atau out- put. Teks original dapat ditambahkan dengan memasukkan gambar kemudian menekan tombol Start. Berikut merupakan tampilan halaman utama yaitu:



Gambar 3 Halaman Utama

Adapun, fungsi tombol Load Image dapat memasukkan gambar tersebut dan akan tampil pada teks original. Setelahnya, terdapat tombol Start untuk memulai pengenalan huruf Bugis dengan

melibatkan penggunaan *particle swarm optimization*. Hasil atau keluaran atau out-put akan muncul apada teks keluaran atau out-put. Selain itu,terdapat tombol Zoom untuk melakukan zoom in atau zoom out terhadap gambar teks original. Dan terakhir tombol Exit digunakan untuk keluar dari aplikasi.

Proses pengenalan huruf tersebutdilakukan pada latar belakang atau background.Melalui interface ini nantinya dapat dilihatkeluaran atau out-put dari pengenalan huruf.Rangkaian proses dimana di laku kan di latar belakang pada aplikasi ini merupakan baca gambar, segmentasi, binerisasi, grayscaleing , thresholding, pemisahan karakter, normalisasihasil segmentasi, ekstrasi ciri (encode input) yang tahapannya disertai *particle swarm optimization*

Halaman Pengenalan Huruf Bugis

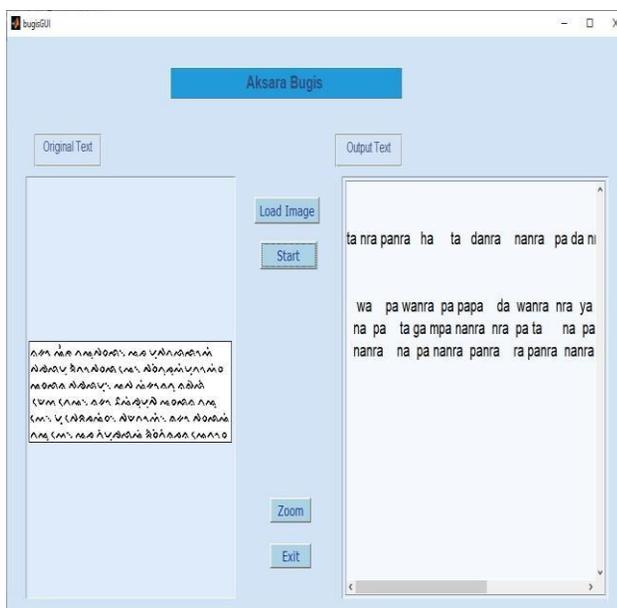
Halaman pengenalan huruf bugis merupakan halaman utama yang telah menerapkan implementasi algoritma *particle swarm optimization* (PSO) dalam pengenalanhuruf bugis. Pada halaman ini gambar citra teks bugis telah diproses dan menghasilkan teks keluaran atau out-put berupa pengenalan huruf bugis tersebut.

Adapun, Menurut Sikki (1991), huruf bugis terdiri dari beberapa macam huruf atau simbol yaitu sebagai berikut:

Tabel 1. Macam Huruf atau Simbol Bugis

No	Bentuk huruf /simbol	Penjelasan
1		dibaca 'ka'
2		dibaca 'ga'
3		dibaca 'nga'
4		dibaca 'ngka'
5		dibaca 'pa'
6		dibaca 'ba'
7		dibaca 'ma'
8		dibaca 'mpa'
9		dibaca 'ta'
10		dibaca 'da'
11		dibaca 'na'
12		dibaca 'nra'

Selanjutnya merupakan tampilan halaman pengenalan huruf bugis pada kalimat paragraf yaitu sebagai berikut:



Gambar 5. Halaman Pengenalan Huruf Bugis Pada Kalimat Paragraf

Gambar yang telah diproses pada halaman ini akan ditampilkan keluaran atau out-put pengenalannya pada teks keluaran atau out-put. Proses pengenalan ini dilakukan secara satu per satu sampa akhirnya teks dapat muncul semua pada keluaran atau out-put. Setelah pengenalan huruf selesai dilakukan, dapat memilih tombol Exit untuk keluar dari aplikasi.

Lalu berikutnya adalah pengujian terhadap beberapa karakter dengan kualitas penulisan yang berbeda:

Tabel 2. Pengujian Karakter

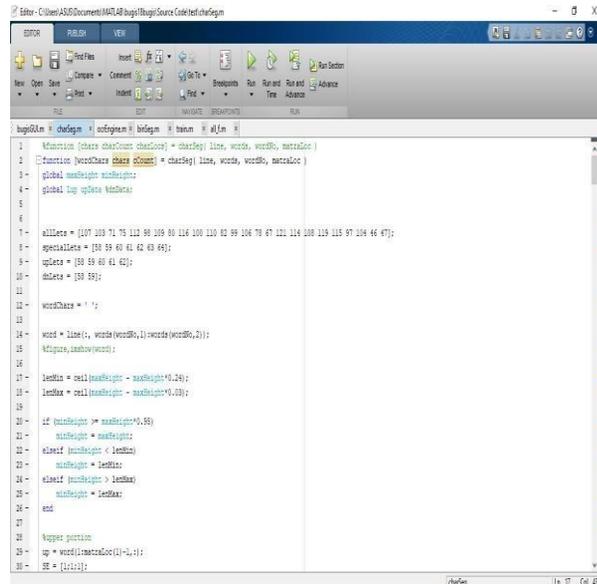
No	Penulisan pertama	Aku rasi	Penulisan kedua	Aku rasi
1		90%		66%
2		94%		46%
3		92%		34%

5.2. Pembahasan

Aplikasi yang telah dibuat berhasil mengimplementasikan algoritma *particle swarm optimization* (PSO) dalam pengenalan huruf bugis. Selain itu, program pada penelitian ini juga dapat digunakan untuk mengubah data huruf didalam format grafis menjadi teks pada hurufBugis. Dengan adanya sistem pengenalan huruf Bugis, akan lebih efektif dan mu dah didalammengenal dan melestarikan huruf Bugis.

Particle Swarm Optimization (PSO) didasarkan pada peniruan perilaku sosial organisme. Perilaku sosial tersebut terdiri dari tindakan individu dan pengaruh dari individu- individu lain didalam suatu kelompok. Katapartikel menunjukkan satu partikel yang berada didalam kelompok atau kawanannya. Tiap-tiap individu atau partikel berperilaku dengan cara melibatkan penggunaan kecerdasannya (intelligence) sendiri dan jugadipengaruhi perilaku kelompok kolektifnya. Dengan demikian, jika satu partikel menemukan path terpendek, sisa kelompok yang lain juga akan dapat untuk segera mengikuti path tersebut meskipun lokasi mereka jauh dikelompok tersebut.

Perilaku partikel didalam kawanannya (swarm) dipengaruhi perilaku partikel individual dan kelompoknya. Didalam Particle Swarm Optimization (PSO), kawanannya diasumsikan mempunyai ukuran tertentu dengan tiap-tiap partikel position/letak awalnya terletak disuatu lokasi yang acak didalam ruang multi dimensi. Tiap-tiap partikel diasumsikan memiliki dua ciri- ciri atau characteristic: position/letak dan kecepatan. Tiap-tiap partikel bergerak didalam ruang/space tertentu dan mengingat position/letak terbaik yang pernah dilalui atau ditemukan terhadap sumber makanan atau nilai fungsi objectif. Tiap-tiap partikel menyampaikaninformasi atau position/letak terbaiknya kepada partikel yang lain dan menyesuaikan position/letak dan kecepatan masing-masing berdasarkan informasi yang diterima mengenai osition/letak tersebut. Particle swarm optimization didasarkan pada kombinasi dari 3 faktor simpel yaitu antara lain kohesi yaitu agar proses dilakukan bersama, separasi yaitu agar tiappartikel jangan terlalu dekat dan penyesuaian (alignment) yaitu agar mengikuti arah bersama. Adapun, penerapan algoritma PSO pada sistem terdapat pada berkas charSeg sebagai berikut:



```
1 #function (char charCode charCode) = charSeg line, word, word0, maraCode
2 #function (wordChar char charCode) = charSeg line, word, word0, maraCode
3
4 global maxSegit maxSegit
5 global up upData maxData
6
7
8 allLeta = [107 109 71 75 112 88 109 80 116 100 110 82 89 106 78 87 121 116 108 119 115 97 104 46 47];
9 specialLeta = [58 59 60 61 62 63 64];
10 upLeta = [58 59 60 61 62];
11 danLeta = [58 59];
12
13 woodShare = " ";
14
15 word = line(i, word(word0,1)) + word(word0,2);
16 #print, show(word);
17
18 lenMin = ceil(maxSegit - maxSegit*0.24);
19 lenMax = ceil(maxSegit - maxSegit*0.10);
20
21 if (maxSegit >= maxSegit*0.95)
22     maxSegit = maxSegit;
23 else if (maxSegit < lenMin)
24     maxSegit = lenMin;
25 else if (maxSegit > lenMax)
26     maxSegit = lenMax;
27
28 #print position
29 up = word(maraCode[1]+1,i);
30 #E = [i,i];
```

Gambar 6. Penerapan Algoritma PSO pada Sistem

Di dalam algoritma PSO terdapat beberapa proses sebagai berikut :

- a. Inisialisasi
 1. Inisialisasi kecepatan awal Pada iterasi ke-

0, dapat dipastikan bahwa nilai kecepatan awal semua partikel merupakan 0.

2. Inisialisasi position/letak awal partikel Pada iterasi ke-0, position/letak awal partikel dibangkitkan dengan persamaan :

$$X = x_{min} + rand [0,1] \times (X_{max} - X_{min})$$

3. Inisialisasi pBest dan gBest Pada iterasi ke-0, pBest akan disamakan dengan nilai position/letak awal partikel. Sedangkan gBest dipilih dari satu pBest dengan fitness tertinggi.

b. *Update Kecepatan*

Untuk melakukan update kecepatan, digunakan rumus berikut :

$$v_{i,j}^{t+1} = w \cdot v_{i,j}^t + c^1 \cdot r^1 (Pbest_{i,j}^t - x_{i,j}^t) + c_2 \cdot r_2 (Gbest_g^t$$

- c. *Update position/letak dan hitung fitness* Untuk melakukan update position/letak, digunakan rumus berikut :

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1}$$

- d. Dilakukan perbandingan antara pBest pada iterasi sebelumnya dengan hasil dari update position/letak. Fitness yang lebih tinggi akan menjadi pBest yang baru. pBest terbaru yang memiliki nilai fitness tertinggi akan menjadi gBest yang baru.

Pada algoritma PSO ini, pencarian *solusi atau solution* dilakukan oleh suatu populasi yang terdiri dari partikel sekian jumlah. Populasi dibangkitkan secara acak dengan batasan nilai terkecil dan terbesar. Masing-masing partikel adalah representasi *solusi atau solution* atau position dari permasalahan atau trouble yang dihadapi. Masing-masing partikel melakukan penelusuran *solusi atau solution* yang optimal dengan adanya melintasi ruang pencarian (search space). Hal ini dilakukan dengan cara masing-masing partikel melakukan penyesuaian terhadap p position/letak terbaik dari partikel tersebut (local best) dan penyesuaian terhadap position/letak partikel terbaik dari seluruh kawanan (global best) selama melintasi ruang pencarian.

Jadi, penyebaran *experience* maupun informasi terjadi pada partikel itu sendiri dan antara suatu partikel dengan p artikel terbaik dari seluruh kawanan selama proses pencarian *solusi atau solution*. Setelah itu, dilakukan proses pencarian untuk mencari position/letak terbaik masing-masing partikel di dalam sejumlah iterasi tertentu sampai didapat atkan position/letak yang relatif steady atau mencapai batas iterasi yang telah ditetapkan.

Di masing-masing iterasi, masing-masing *solusi atau solution* yang direpresentasikan oleh position/letak partikel, dievaluasi performansinya dengan cara yaitu memasukkan *solusi atau solution* tersebut ke dalam fitness function. Masing-masing partikel diperlakukan seperti halnya sebuah titik pada suatu dimensi ruang tertentu.

Kemudian terdapat dua faktor yang memberikan karakter terhadap status partikel pada ruang pencarian yaitu position/letak partikel dan kecepatan partikel. Masing-masing partikel diperlakukan seperti sebuah titik pada suatu dimensi ruang tertentu. Kemudian terdapat dua faktor yang memberikan karakter terhadap status partikel pada ruang pencarian yaitu position/letak partikel dan kecepatan partikel.

Pada proses grayscale citra masukan atau input yang berwarna dapat diubah menjadi gambar yang terdiri dari warna putih dan gradasi warna hitam dengan melibatkan penggunaan representasi warna RGB atau red green blue. Pengubahan gambar ke dalam bentuk grayscale atau hitam putih ini dilakukan dengan mengambil nilai piksel dari suatu gambar masukan atau input.

Proses thresholding yaitu dengan melalui pemeriksaan nilai intensitas dari sebuah piksel apakah ada di bawah atau di atas sebuah nilai intensity threshold yang telah diatur. Apabila nilai piksel tersebut berada di atas batas nilai yang telah diatur, maka piksel tersebut akan diubah menjadi putih yang berarti bahwa piksel tersebut merupakan background, dan sebaliknya bila piksel tersebut berada di bawah batas nilai yang ditentukan maka piksel tersebut akan diubah menjadi berwarna

hitam yang berarti dianggap sebuah karakter.

Selanjutnya yaitu pemisahan karakter yang dilakukan dengan mapping atau kegiatan memetakan jumlah titik hitam masing-masing baris pada gambar ke sumbu y (Y-Mapping) dan masing-masing baris karakter hasil mapping tersebut di-mapping atau dipetakan lagi ke sumbu x (X-Mapping). Juga ada normalisasi karakter yang diperlukan karena data yang dihasilkan dari proses pemisahan karakter dapat bervariasi dimensinya. Oleh karena itu untuk mendapatkan data yang seragam, akurat dan konsisten dari masing-masing sampel, data gambar hasil segmentasi tersebut akan dinormalisasikan (stretch/shrink) dengan size atau ukuran tertentu. Dengan demikian jumlah area yang ada di masing-masing sampel akan disesuaikan dengan jumlah partikel yang akan dipakai.

Lalu, setelah didapatkan informasi matrik hasil proses normalisasi masing-masing karakter didalam dimensi matrik, selanjutnya merupakan mengekstraksi masing-masing piksel dari citra ke didalam sebuah vektor hal ini dilakukan agar dapat dihasilkan kumpulan data yang seragam di masing-masing sampel yang akan diamati. Ciri-ciri citra merupakan piksel-piksel yang memiliki nilai satu dan nol. Piksel bernilai nol merupakan piksel berwarna hitam atau object, sebaliknya piksel bernilai satu merupakan piksel berwarna putih atau background. Proses ekstraksi bertujuan untuk menangkap ciri tertentu dari citra masukan atau input. Kemudian, dimulailah proses pelatihan (training process) dimana pada proses ini jaringan akan dilatih berdasarkan vektor masukan atau input gambar huruf dan angka yang sudah disiapkan sebelumnya untuk proses pelatihan. Didalam proses ini huruf dan angka dilatihkan satu per satu sesuai dengan masukan atau input vektornya. Adapun keluaran atau output dari modul ini ialah file data informasi yang berisi bobot koneksi antar neuron dari semua pola vektor yang terbentuk dan vektor pola yang dipanggil secara stabil (konvergen).

Keluaran atau output dari proses pelatihan ialah sekumpulan nilai bobot jaringan dan vektor pola stabil yang disimpan ke didalam sebuah file pelatihan, file pelatihan tersebut dapat dibuka kembali bila akan diupdate ataupun digunakan didalam proses pengenalan. Sedangkan keluaran atau output dari proses pengenalan ialah konversi dari citra yang berisi angka-angka bipolar menjadi huruf-huruf atau angka yang dapat disimpan ke didalam file teks.

Di dalam hal ini perlu diperhatikan bahwa sebelum melakukan proses pengenalan, terlebih dahulu dilakukan pengecekan apakah sudah ada data pelatihan (training) yang dapat dijadikan acuan didalam melakukan pengenalan. Hal tersebut penting diperhatikan agar proses pengenalan huruf Bugis dengan disertai tahapan *particle swarm optimization* dapat dilakukan.

5.2.1 Pengujian Algoritma PSO

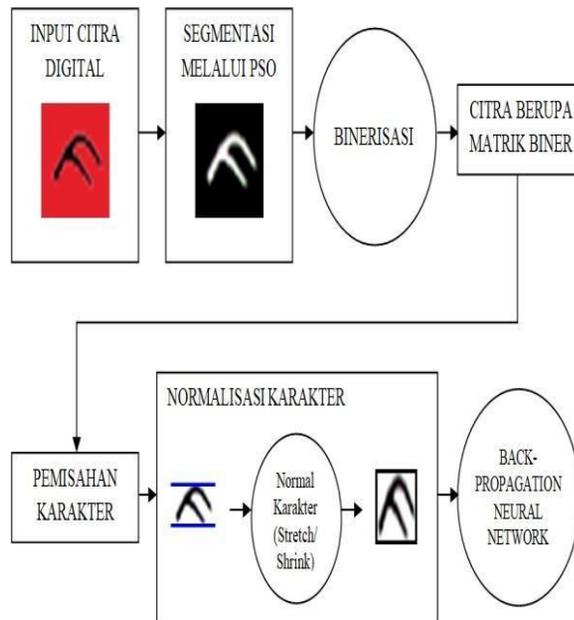
Rangkaian proses yang akan dilakukan oleh aplikasi ini ialah baca gambar, segmentasi dengan PSO, binerisasi, grayscale, thresholding, pemisahan karakter, normalisasi hasil segmentasi, ekstraksi ciri (encode input) dan backpropagation neural network. Proses backpropagation neural network sendiri membutuhkan proses pelatihan (training) agar keluaran atau output yang dihasilkan benar dan akurat.

Jadi secara garis besar dapat dikatakan bahwa alur aplikasi ini dimulai dari dimasukkannya atau inputkannya sebuah gambar dari sebuah gambar teks dengan latar belakang berwarna. Dimana gambar tersebut akan melalui serangkaian manipulasi citra didalam proses image preprocessing diantaranya proses segmentasi PSO, grayscale, thresholding, pemisahan karakter, normalisasi dan ekstraksi ciri yang akan mengubah gambar tersebut menjadi serangkaian nilai-nilai yang memiliki nilai guna untuk proses utama berikutnya yaitu : proses pelatihan (training process) ataupun proses pengenalan (recognition process).

Keluaran atau output dari proses pelatihan ialah sekumpulan nilai bobot jaringan dan vektor pola stabil yang disimpan ke didalam sebuah file pelatihan, file pelatihan tersebut dapat dibuka kembali bila akan diupdate ataupun digunakan didalam proses pengenalan. Sedangkan keluaran atau output dari proses pengenalan ialah konversi dari citra yang berisi angka-angka bipolar menjadi huruf-huruf atau angka yang dapat disimpan ke dalam file teks.

Di dalam hal ini perlu diperhatikan bahwa sebelum melakukan proses pengenalan,

terlebih dahulu dilakukan pengecekan apakah sudah ada data pelatihan (training) yang dapat dijadikan acuan dalam melakukan pengenalan. Bila tidak, maka proses pengenalan tidak bisa dilakukan.



Gambar 7. Proses Pengenalan Karakter

Pengujian Algoritma PSO Pengujian dilakukan dengan dengan komputer Processor Celeron N3060 Dual Core, Memory RAM 2.0GB, Video Card Intel HD Graphics Integrated.

Terdapat dua puluh empat citra aksarabugis (aksara lontara) untuk masing masing implementasi (CPU, Naif, full device) yang dibagi menjadi 4 buah citra uji. Keempat citra digital tersebut diuji dengan nilai cluster sebanyak 2. Jumlah partikel 20, 100 dan 1000. Jumlah iterasi sebanyak 20, 40, dan 60. Keduapuluhlima citra aksara bugis (aksara lontara) yang digunakan sebagai bahan uji salah satunya citra berformat JPG dan memiliki ukuran 612 X202 pixel.

Dari hasil pengujian diatas didapatkan bahwa simplementasi segmentasi citra dengan melibatkan penggunaan PSO secara paralel memberikan peningkatan performa pemrosesan yang signifikan dibandingkan dengan segmentasi citra dengan PSO yang tidak paralel.

Jika diketahui n ialah jumlah partikel dan i ialah jumlah iterasi, maka kompleksitas waktu worst case dari implementasi diatas ialah :

- 1) $O(i * 2n)$ pada implementasi CPU, karena terdapat 2 buah loop sebanyak n elemen pada loop i .
- 2) $O(i * n)$ pada implementasi naif, karena loop yang digunakan untuk mengupdate partikel diparalelkan sehingga kompleksitas loop tersebut menjadi $O(1)$ atau konstan.
- 3) $O(i * \lg n)$ pada implementasi full device, karena loop untuk mengupdate partikel dan mencari $gBest$ diparalelkan sehingga menjadi $O(1)$ (konstan), dan untuk mencari nilai fitness terkecil digunakan algoritma parallel reduction yang memiliki kompleksitas $O(\lg n)$.

Dilihat dari kompleksitas waktu masing-masing implementasi diatas, maka secara teoritis, untuk jumlah partikel cukup banyak, implementasi full device ialah yang tercepat karena waktu pemrosesannya relatif terhadap logaritma basis dua dari jumlah partikel, diikuti oleh implementasi naif, dan implementasi pada CPU ialah implementasi yang paling lambat.

Untuk jumlah partikel yang cukup sedikit (20 partikel), implementasi pada CPU ialah implementasi yang berjalan paling cepat. Hal ini disebabkan karena peningkatan kecepatan

kecepatannya tertutupi oleh overhead pada bagian lain, misalnya inisialisasi memori, copy data dan inisialisasi kernel.

Pada jumlah partikel yang cukup banyak (100 dan 1000 partikel), implementasi pada GPU berjalan lebih cepat daripada implementasi pada CPU karena overhead pada pemrosesan paralel pada CUDA tidak signifikan dibandingkan peningkatan pemrosesannya. Implementasi full device ialah implementasi yang paling cepat, yaitu hampir setengah dari waktu pemrosesan pada implementasi CPU, diikuti oleh implementasi naif

5.2.2 Analisis Proses Pelatihan

Pada proses ini sistem melakukan proses pelatihan di mana tiap aksara bugis dilatihkan tiap karakter sesuai dengan nilai masukan atau input vektornya. Jenis karakter yang dilatihkan ialah BugisA 30pt. Berikut nilai-nilai parameter yang digunakan didalam proses pelatihan :

Maksimum Iterasi tiap polas: 10000

Sedangkan parameter yang digunakan sebagai arsitektur JST ialah sebagai berikut

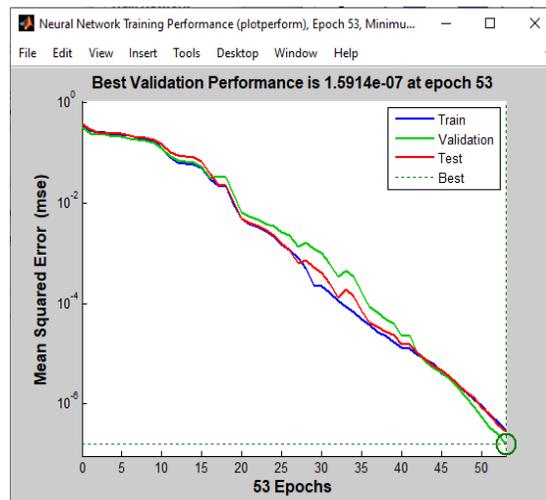
Pola Masukan atau input	: Tinggi = 20, Lebar = 20
Pola Keluaran atau out-put	: Tinggi = 20, Lebar = 20
Jumlah Neuron	: 1600

Proses yang terjadi pada pelatihan ialah citra masukan tersebut diubah ke didalam citra biner dengan melibatkan penggunaan metode grayscaleing dan thresholding. Intensitas tiap-tiap pixel citra yang diubah ke didalam citra biner yang kemudian dirubah ke didalam bentuk bipolar 1 dan 0 Di dalam sebuah matrik vektor yang sudah disiapkan, kemudian vektor masukan atau input di encode untuk menghasilkan sebuah masukan

5.2.3 Pengujian Sistem Pengenalan Karakter

Pengujian dilakukan dengan komputer Processor Celeron N3060 Dual Core, Memory RAM 2.0 GB, Video Card Intel HD Graphics Integrated. Untuk pengujian pertama dilakukan dengan citra yang berisi karakter yang merepresentasikan tiap-tiap huruf dan angka dengan ukuran 30pt, 40pt dan 50pt. Selain itu, pengujian dilakukan juga pada komputer Processor Intel Core i5, Memory RAM 8.0 GB, Video Card NVIDIA GeForce 840M. Hal ini dilakukan agar dapat diketahui bagaimana pengujian pada perangkat dengan spesifikasi yang berbeda.

Berikut ialah grafik hasil pengujian iterasi yang pernah diuji coba:



Gambar 8. Grafik Hasil Pengujian Iterasi

Di dalam pengujian ini dilakukan dengan tujuan apakah sistem bisa mengenali dengan baik tiap pola karakter dengan membandingkan terhadap ukuran karakter yang sama dengan yang

dilatihkan. Untuk beberapa karakter untuk ukuran font tertentu sistem tidak bisa mengenali karakter tersebut dengan benar, ini dimungkinkan disebabkan oleh proses normalisasi dimana dilakukan oleh sistem, karena pada proses normalisasi (stretch) semua karakter diseragamkan ukurannya menjadi 20 x 20, hal inilah yang kemudian menjadikan sistem salah didalam mengenali pola yang dimaksud dikarenakan terjadinya perubahan nilai vektor masukan atau input suatu karakter akibat proses normalisasi tersebut sehinggajaringan stabil pada pola lain .

4. KESIMPULAN

Berdasarkan bab sebelumnya yang telah membahas hasil dan pembahasan, maka dari itu penulis mencoba untuk menarik kesimpulan serta saran yang kemungkinan dapat menjadi pertimbangan yaitu berikut:

- 1) Penelitian ini menerapkan algoritma *particle swarm optimization* (PSO) dalam pengenalan huruf bugis. Penelitian bertujuan mengenali huruf Bugis melibatkan penggunaan algoritma *particle swarm optimization*.
- 2) Aplikasi yang dibuat berhasil menjadi program yang efektif untuk membantu membuat pengenalan huruf Bugis lebih efektif dan mudah didalam mengenal dan melestarikan huruf Bugis.

DAFTAR PUSTAKA

- [1] Nugroho. (2015). *Segmentasi Citra Aksara Jawa Menggunakan Algoritma Particle Swarm Optimization Paralel*. Banyumas: STIK Yos Sudarso.
- [2] Septia. (2015). *Perhitungan Kurva Dispersi dari Vibrasi Foton untuk Kisi Satu Dimensi Secara Numerik dengan Matlab*. Bandung: Institut Teknologi Bandung.
- [3] Sikki, M. (1991). *Tata Bahasa Bugis*. Jakarta: Departemen Pendidikan dan Kebudayaan.