

Pengenalan Tulisan Tangan Huruf Hijaiyah Sambung Menggunakan Algoritma *Template Matching Correlation*

Nina Retno Angraheni¹, Rusdi Efendi², Endina Putri Purwandari³

^{1,2,3}Program Studi Teknik Informatika, Fakultas Teknik, Universitas Bengkulu.

Jl. W.R. Supratman Kandang Limun Bengkulu 38371A INDONESIA
(telp: 0736-341022; fax: 0736-341022)

¹ninaangraheni@gmail.com

²r_efendi@gmail.com

³endinaputrip@gmail.com

Abstrak. Huruf hijaiyah merupakan huruf atau karakter yang digunakan dalam bahasa Arab yang umumnya diketahui berjumlah 28 huruf, dan memiliki bentuk yang berbeda sesuai dengan posisi sebuah huruf pada sebuah kata yaitu di awal, di tengah, di akhir, dan huruf asli yang berdiri sendiri. Penelitian ini bertujuan membuat suatu aplikasi yang dapat mengenali tulisan tangan huruf hijaiyah sambung dan mengetahui nilai akurasinya menggunakan algoritma *Template Matching Correlation*. Database citra *template* yang digunakan sebanyak 28 untuk huruf awal, 28 untuk huruf tengah, dan 28 untuk huruf akhir dengan sampel yang diambil dari tulisan tangan seseorang sebanyak 28 untuk setiap posisi huruf dengan ukuran sebesar 100X150 piksel. Sistem dibangun dengan menggunakan Matlab R2013a. Sistem dibagi menjadi 2 tahapan yaitu, prapemrosesan dengan 3 tahapan berupa *image thresholding*, penipisan, dan segmentasi, serta tahapan kedua yaitu proses pengenalan. Penelitian ini memiliki tingkat akurasi sebesar 56,41% berdasarkan 26 citra uji.

Kata kunci: Huruf Hijaiyah, Pengenalan Tulisan Tangan, *Template Matching Correlation*

***Abstract.* The hijaiyah letters are the letters or characters used in Arabic which are generally known to be 28 letters, and have different shapes according to the position of a letter in a word that is at the beginning, in the middle, at the end, and the original standalone letter. The goal in this reasearch is to create an application that can recognize handwriting of hijaiyah cursive letters and know the value of accuracy by using the *Template Matching Correlation* algorithm. The template image database used is 28 for the initial letter, 28 for the middle letter, and 28 for the final letter with samples taken from one's handwriting of 28 for each letter position with size of 100X150 pixels. The system is built using Matlab R2013a. The system is divided into 2 phases namely, prapemrosesan with 3 stages of image thresholding, thinning, and segmentation, and the second stage is the process of recognition. This study has an accuracy of 56,41% based 26 trials.**

Keywords: Hijaiyah Letters, Handwriting Recognition, Template Matching Correlation.

I. PENDAHULUAN

Huruf hijaiyah merupakan huruf atau karakter yang digunakan dalam bahasa Arab yang umumnya diketahui berjumlah 28 huruf [1]. Pada tulisan huruf Arab setiap kata ditulis dalam bentuk yang bersambung, yang artinya huruf - huruf dalam sebuah kata digabung menjadi satu untuk membentuk suatu kata yang memiliki makna. Banyaknya bentuk pada huruf Arab sambung membuat orang yang baru belajar sulit membedakan setiap huruf yang ada dan pengenalan huruf Arab sambung masih dilakukan dengan menggunakan penglihatan. Cara tersebut memiliki kelemahan, yaitu pembaca harus memiliki sumber yang menunjukkan tentang bentuk-bentuk huruf arab dalam berbagai bentuk atau pengajar yang sudah memahami bentuk-bentuk huruf arab. Jika tidak, maka bagi pembaca yang baru belajar kesulitan untuk mengenali huruf tersebut [2].

Perkembangan teknologi mempengaruhi perkembangan ilmu pengetahuan dalam hal pengenalan pola (*pattern recognition*). Terdapat macam-macam pengenalan pola yaitu *Voice Recognition*, *Fingerprint Identification*, *Face Identification*, *Optical Character Recognition (OCR)*, *Robot Vision*, dan *Handwriting Identification* (Pengenalan Tulisan Tangan). Pengenalan huruf tulisan tangan adalah sebuah teknik dimana *input* data yang berupa lembaran kertas hasil *scan* menggunakan scanner dan menghasilkan gambar pada komputer yang dikenali sebagai titik-titik (*bitmap*).

Tulisan tangan memiliki banyak variasi dari setiap penulis sehingga pengenalan ini sangat

dibutuhkan agar dapat mengidentifikasi huruf hijaiyah pada sebuah tulisan tangan huruf hijaiyan sambung. Jika salah pasangan ketika menyambung huruf hijaiyah dapat menimbulkan perbedaan pada bacaan dan arti.

Algoritma yang sudah digunakan untuk proses pengenalan antara lain, *Template Matching Correlation*, *Feature Extraction*, *Participal Component Analysis*, dan *Chan Code*. Pada penelitian ini penulis menggunakan metode *Template Matching Correlation*. Algoritma *Template Matching Correlation* dinilai cukup sederhana dan dapat diterapkan tanpa harus melakukan proses ekstraksi terlebih dahulu. Oleh karena itulah algoritma *Template Matching Correlation* dipilih sebagai metode yang akan digunakan dalam mengatasi masalah pemisahan huruf arab sambung.

Berdasarkan penelitian terkait, penelitian ini mengembangkan suatu aplikasi berupa pengenalan huruf Hijaiyah sambung tulisan tangan sebagai modelnya dengan mengajukan suatu metode alternatif yaitu metode *Template Matching Correlation*, dengan judul "Pengenalan Tulisan Tangan Huruf Hijaiyah Sambung Menggunakan Algoritma *Template Matching Correlation*".

II. LANDASAN TEORI

A. Pengolahan Citra Digital

Pengolahan citra digital adalah teknik untuk memanipulasi dan memodifikasi citra [3]. Sementara menurut Wijaya [4] pengolahan citra digital dapat diartikan sebagai pemrosesan gambar berdimensi-dua melalui komputer digital. Ada dua prinsip dalam pengolahan citra digital: peningkatan informasi piktorial untuk interpretasi manusia; dan pengolahan data citra untuk penyimpanan, transmisi dan representasi bagi peralatan persepsi (*perception*). Pengolahan citra

merupakan bagian penting yang mendasari berbagai aplikasi nyata, seperti pengenalan pola, penginderaan jarak jauh melalui satelit atau pesawat udara, dan *machine vision*. Pengenalan citra sebagai pengenalan pola berperan memisahkan objek dari latar belakang secara otomatis. Selanjutnya, objek akan diproses oleh pengklasifikasi pola.

B. Pengenalan Pola

Pengenalan pola adalah suatu ilmu untuk mengklasifikasikan atau menggambarkan sesuatu berdasarkan pengukuran kuantitatif fitur (ciri) atau sifat utama dari suatu objek [3]. Pola adalah suatu entitas yang terdefinisi dan dapat di definisikan serta diberi nama. Sidik jari adalah suatu contoh pola. Pola bisa merupakan kumpulan hasil pengukuran atau pemantauan dan bisa dinyatakan dalam notasi vektor atau matriks. Terdapat macam-macam pengenalan pola yaitu *Voice Recognition*, *Fingerprint Identification*, *Face Identification*, *Optical Character Recognition* (OCR), *Robot Vision*, dan *Handwriting Identification* (Pengenalan Tulisan Tangan). Pengenalan pola yang digunakan pada penelitian ini adalah *Handwriting Identification* (Pengenalan Tulisan Tangan).

C. Pengenalan Tulisan Tangan

Pengenalan tulisan tangan dapat dilakukan secara *offline* maupun *online*. Pengenalan huruf tulisan tangan secara *offline* adalah sebuah teknik dimana *input* data yang berupa lembaran kertas hasil *scan* menggunakan gambar pada komputer yang dikenali sebagai titik-titik (*bitmap*), *bitmap* inilah yang kemudian diproses lebih lanjut dengan menggunakan algoritma tertentu menjadi karakter sehingga dapat dikenali dan diolah menjadi informasi [5].

D. Template Matching

Template Matching adalah salah satu teknik dalam pengolahan citra digital yang berfungsi untuk mencocokkan tiap-tiap bagian dari suatu citra dengan citra yang menjadi *template* (acuan) [3]. Citra masukkan dibandingkan dengan citra *template* yang ada di dalam *database*, kemudian dicari kesamaannya dengan menggunakan aturan tertentu [6]. *Template Matching Correlation* adalah teknik statistik yang digunakan untuk mencari dua variabel/matriks atau lebih yang sifatnya kuantitatif [3].

Kesamaan antar dua buah matriks citra dapat dihitung dengan menghitung nilai korelasinya. Nilai korelasi dua buah matriks dapat dihitung dengan menggunakan rumus 2.1.

$$r = \frac{\sum_{k=1}^n (x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j)}{\sqrt{\sum_{k=1}^n (x_{ik} - \bar{x}_i)^2 \sum_{k=1}^n (x_{jk} - \bar{x}_j)^2}} \quad (2.1)$$

Dimana \bar{x}_i dirumuskan dengan persamaan 2.2 dan \bar{x}_j dirumuskan dengan persamaan 2.3.

$$\bar{x}_i = \frac{1}{n} \sum_{k=1}^n x_{ik} \quad (2.2)$$

$$\bar{x}_j = \frac{1}{n} \sum_{k=1}^n x_{jk} \quad (2.3)$$

Keterangan:

r adalah nilai korelasi antara dua buah matriks (nilainya antara -1 dan +1)

x_{ik} adalah nilai pixel ke- k dalam matriks i

x_{jk} adalah nilai pixel ke- k dalam matriks j

\bar{x}_i adalah rata-rata dari nilai matriks i

\bar{x}_j adalah nilai rata-rata dari matriks j

n menyatakan jumlah pixel dalam suatu matriks

E. Prapemrosesan

Prapemrosesan merupakan tahapan awal sebelum dilakukan segmentasi citra terhadap huruf yang akan dikenali. Prapemrosesan yang digunakan dalam penelitian ini meliputi *Image Thresholding*, *Thinning*, *Segmentasi*.

F. Pengkodean Karakter (*Feature Extraction*)

Feature Extraction merupakan salah satu cara untuk mengenali karakter dengan melihat ciri-ciri khusus yang dimiliki oleh karakter tersebut. Oleh karena itu, setelah melakukan proses *scalling* proses selanjutnya adalah *Feature Extraction*.

G. Huruf Hijaiyah

Huruf Hijaiyah atau huruf arab merupakan huruf yang sudah ada sejak dahulu yang digunakan oleh orang-orang muslim di seluruh penjuru dunia untuk membaca Al-Quran. Jumlah Huruf Hijaiyah yang umumnya diketahui berjumlah 28 huruf, yaitu : س ز ر ذ د خ ح ج ث ت ب ا . Huruf-huruf Hijaiyah yang berjumlah 28 itu terbagi menjadi dua, yaitu huruf “Qamariyah” dan huruf “Syamsiyah”. Huruf Qamariyah atau huruf bulan adalah huruf yang dibaca secara jelas namun tetap mempertegas pembacaan dari huruf lam. Huruf Qamariyah ada 14 huruf, yaitu : ق ف غ ع خ ح ج ب ا : ك م ل ن و ه ي . Huruf Syamsiyah atau huruf matahari adalah huruf yang menghilangkan pembacaan dari huruf lam. Huruf Syamsiyah ada 14 huruf, yaitu : [1]. ن ل ظ ط ض ص ش س ز ر ذ د ث ت :

III. METODOLOGI

Langkah-langkah yang dilakukan dalam penelitian ini adalah :

1) Jenis Penelitian

Jenis penelitian yang dilakukan adalah penelitian terapan (*applied research*). Penelitian terapan merupakan penelitian yang dilakukan dengan tujuan menerapkan, menguji, dan mengevaluasi kemampuan suatu teori yang diterapkan dalam memecahkan masalah-masalah praktis [7].

2) Populasi dan Sampel

a) Populasi

Populasi yang digunakan dalam penelitian ini yaitu citra huruf Arab sambung tulisan tangan sebanyak 26 kata. Pengambilan citra huruf Arab ini dilakukan dengan cara sejumlah huruf Arab sambung tulisan tangan dibubuhi pada kertas HVS dengan background putih yang kemudian tulisan tangan pada kertas tadi akan dijadikan citra tulisan tangan dengan cara ditangkap (*capture*) melalui media *scanner* Canon MG2500.

b) Sampel

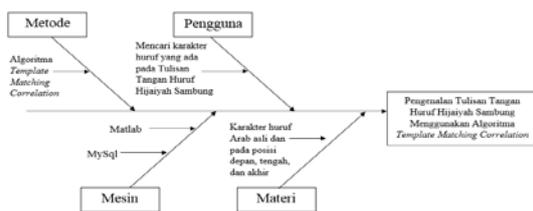
Sampel yang digunakan pada penelitian ini adalah citra huruf Arab sambung tulisan tangan dari seseorang, setiap orang melakukan pembubuhan huruf Arab sambung tulisan tangan menggunakan 1 buah alat tulis berbeda masing-masing sebanyak 1 kali pada setiap kata, sesuai dengan jumlah huruf untuk posisi di depan, tengah, akhir, dan huruf asli. Huruf Arab yang berjumlah 26 kata.

IV. ANALISIS DATA DAN PERANCANGAN

A. Identifikasi Masalah

Dalam huruf Arab terdapat 28 huruf dan setiap huruf memiliki 4 bentuk yang berbeda sesuai dengan letak huruf pada sebuah kata. Untuk mengurangi kesalahan dari pembacaan huruf pada setiap kata, diperlukan langkah-langkah pemisahan (*segmentasi*) yang kemudian akan dicocokkan dengan huruf Arab acuan dengan menghitung nilai korelasinya. Teknik pencocokan yang digunakan adalah *Template Matching Correlation*.

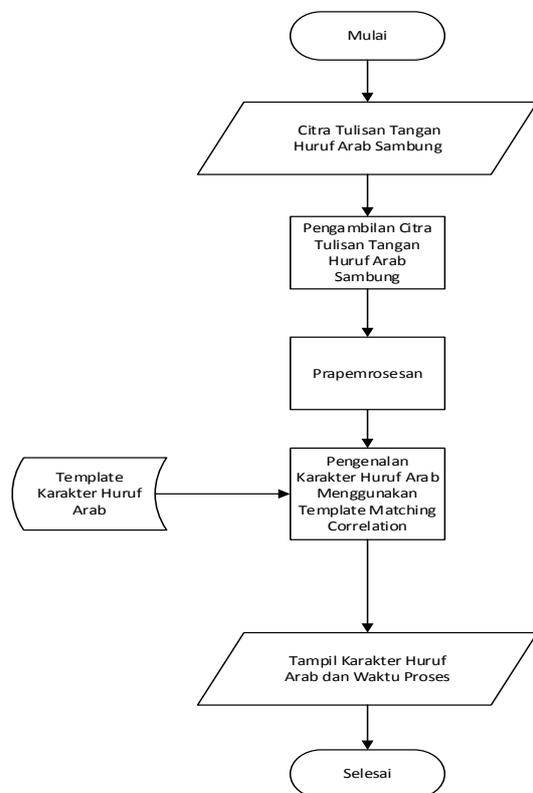
Masalah pada penelitian ini diidentifikasi dengan menggunakan diagram Ishikawa. Diagram Ishikawa atau yang sering dikenal dengan *Cause and Effect Diagram* adalah diagram yang digunakan untuk memahami dan mengidentifikasi serta menggambarkan beberapa masalah yang terjadi pada sistem dan akibat yang ditimbulkan oleh masalah. Permasalahan pada penelitian ini secara umum dapat ditunjukkan pada gambar 4.1.



GAMBAR 4. 1 DIAGRAM ISHIKAWA

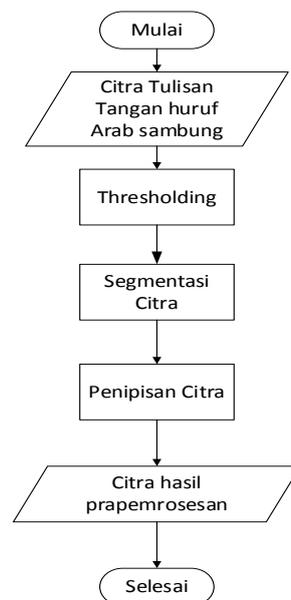
B. Analisis Alur Sistem

Alur sistem merupakan analisis tahapan kerja sistem yang akan dibangun. Secara garis besar tahapan perancangan sistem yang akan dibangun dapat dilihat pada gambar 4.2.



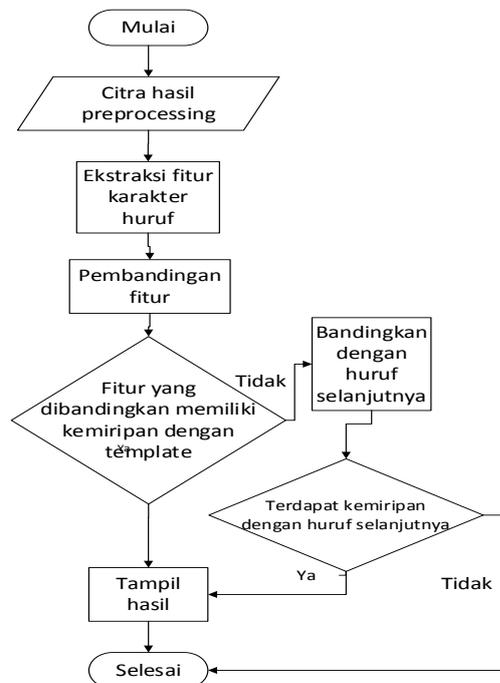
GAMBAR 4. 2 DIAGRAM ALIR SISTEM

Proses prapemrosesan merupakan proses yang harus dilakukan sebelum melakukan proses pengenalan. Proses prapemrosesan terdiri dari thresholding, segmentasi, dan penipisan. *Flowchart* prapemrosesan dapat dilihat pada Gambar 4.3.



GAMBAR 4. 3 DIAGRAM ALIR PRAPEMROSESAN

Proses *Template Matching Correlation* dapat dilihat pada gambar 4.4.

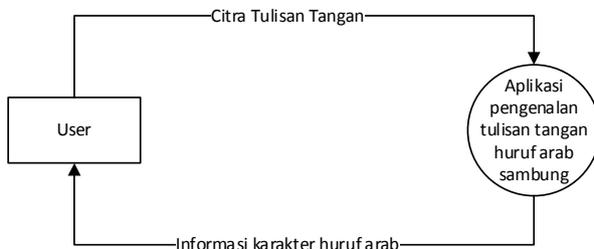


GAMBAR 4. 4 DIAGRAM ALIR PROSES TEMPLATE MATCHING CORRELATION

C. Perancangan Sistem

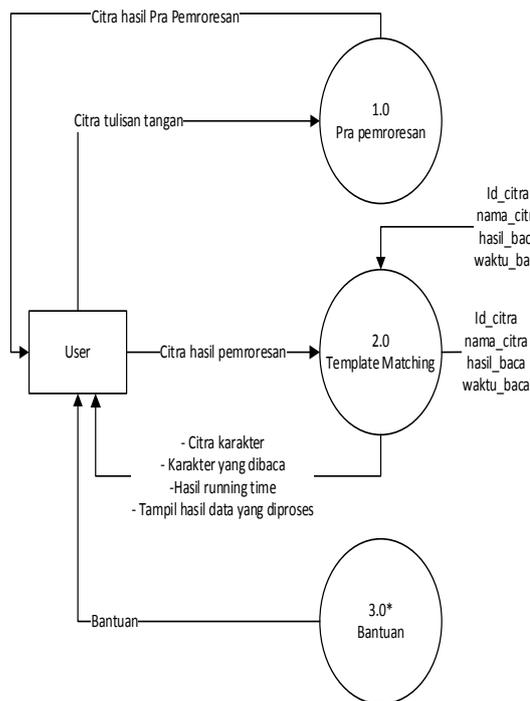
Perancangan yang dilakukan dalam sistem ini adalah perancangan *Data Flow Diagram* (DFD).

1. Perancangan Data Flow Diagram (Diagram Konteks)



GAMBAR 4. 5 DIAGRAM KONTEKS APLIKASI PENGENALAN TULISAN TANGAN HURUF ARAB SAMBUNG

2. Perancangan Data Flow Diagram Level 1



Gambar 4. 6 Diagram Level 1

V. PEMBAHASAN

A. Implementasi Antarmuka

Pada tahapan implementasi antar muka ini, sistem akan diimplementasikan menggunakan bahasa pemrograman bahasa pemrograman MATLAB R2013a.

1. Halaman Home

Halaman utama merupakan halaman pertama yang akan diakses saat aplikasi dijalankan. Pada halaman utama ini terdapat judul dari aplikasi yang telah dibuat ini yaitu “Pengenalan Tulisan Huruf Hijaiyah Sambung Menggunakan Algoritma *Template Matching Correlation*”. Halaman Utama dapat dilihat pada Gambar 5.1.



GAMBAR 5. 1 HALAMAN UTAMA SISTEM

2. Halaman Pengenalan Huruf Hijaiyah

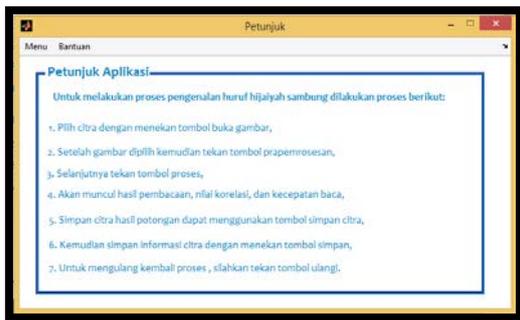
Halaman pengenalan huruf hijaiyah merupakan submenu dari menu Menu. Pada halaman ini terdapat tiga panel, yaitu panel untuk citra yang akan diproses, panel untuk citra yang sudah diproses, dan panel untuk *database* informasi citra yang sudah disimpan yang dapat dilihat pada Gambar 5.2.



GAMBAR 5. 2 HALAMAN PENGENALAN HURUF HIJAIYAH

3. Halaman Petunjuk

Halaman petunjuk merupakan submenu dari menu bantuan. Halaman ini menampilkan petunjuk penggunaan aplikasi untuk melakukan pengenalan tulisan tangan huruf hijaiyah sambung. Halaman petunjuk penggunaa aplikasi pengenalan huruf hijaiyah dapat dilihat pada Gambar 5.3.



GAMBAR 5. 3 HALAMAN PETUNJUK

4. Halaman Tentang

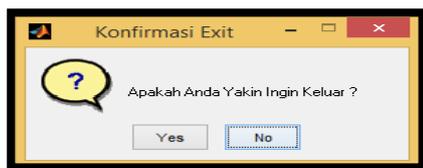
Halaman tentang merupakan submenu dari menu bantuan. Halaman ini menampilkan judul aplikasi, tujuan dibuatnya aplikasi ini, dan nama pembuat aplikasi. Halaman tentang pengenalan huruf hijaiyah dapat dilihat pada Gambar 5.4.



GAMBAR 5. 4 HALAMAN TENTANG

5. Halaman Keluar

Halaman keluar merupakan submenu dari menu Menu. Halaman ini menampilkan kotak dialog yang berguna untuk menanyakan kepada pengguna apakah benar-benar ingin keluar dari aplikasi ini. Halaman keluar dari aplikasi pengenalan tulisan tangan huruf hijaiyah sambung dapat dilihat pada Gambar 5.5.



Gambar 5.5 Halaman Keluar

B. Implementasi Sistem

Pada subbab ini akan dibahas tentang implementasi sistem pada proses pengenalan dan potongan coding untuk masing-masing proses. Proses yang pertama kali dilakukan untuk pengenalan huruf hijaiyah adalah memilih citra yang akan diproses.

Untuk memulai pengenalan pertama yang harus dilakukan adalah menekan tombol Buka Gambar pada Gambar 5.6. *Coding* yang digunakan untuk Buka Gambar ini adalah sebagai berikut

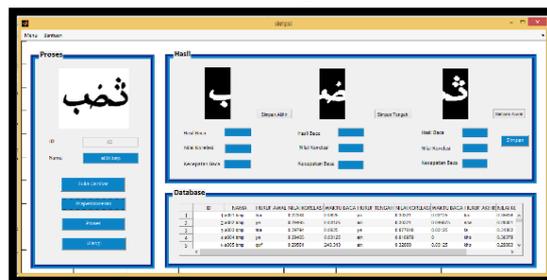
```
[filename,pathname,filterindex]=uigetfile(
{'*.bmp'}, 'Buka foto', '');
```



GAMBAR 5. 6 TAMPILAN BUKA GAMBAR

Kemudian setelah itu akan muncul *window* seperti yang ada pada Gambar 5.6, pada proses ini pengguna dapat memilih citra dengan format *.bmp. Sebagai contoh pengguna memilih citra dengan nama a001.bmp, kemudian tekan tombol *open*. Setelah citra dipilih maka citra akan ditampilkan pada *axes* yang terdapat pada *panel* proses, dapat dilihat pada Gambar 5.7. *Coding* yang digunakan untuk menampilkan gambar yang dipilih adalah sebagai berikut

```
guidata(hObject,handles);
handles.current_data1=handles.data1;
axes(handles.axes4)
imshow(handles.current_data1);
```



GAMBAR 5. 7 GAMBAR YANG TAMPIL PADA PANEL PROSES

Setelah citra berhasil dimuat pada *axes panel* proses, tahap selanjutnya adalah prapemrosesan. Pada tahap ini citra a001.bmp akan melewati tiga proses sekaligus yaitu, *image thresholding*, penipisan, dan segmentasi. Pada proses *image thresholding* citra a001.bmp akan diubah menjadi citra dengan format skala keabuan(*greyscale*), yang mempunyai kemungkinan nilai lebih dari 2 ke citra biner yang hanya memiliki 2 buah nilai (0 dan 1). *Coding* yang digunakan untuk proses *image thresholding* adalah sebagai berikut

```
for i=1:m %baris dari matriks citra
    masukan dari 1 sampai n
    for j=1:n %kolom dari matriks citra
        masukan dari 1 sampai n
        if a(i,j)<=128 %jika nilainya kurang dari
            sama dengan 128
            bw(i,j)=0; %maka nilai a dari baris i dan
            kolom j akan samadengan 0
        else
            bw(i,j)=255; %jika tidak maka nilainya
            samadengan 1
        end
    end
end
```

Setelah proses *image thresholding* proses selanjutnya adalah penipisan. Pada proses ini citra akan ditipiskan dengan menggunakan metode erosi karena pada metode ini menggunakan *strel* untuk membuat berbagai bentuk elemen penstruktur penentu bentuk. *Coding* erosi adalah sebagai berikut

```
ge=imerode(BW,se); %melakukan erosi
```

Proses terakhir pada prapemrosesan adalah segmentasi, dimana citra yang telah diproses sebelumnya akan dipotong menjadi tiga bagian dengan menggunakan operasi *cropping*. Pemotongan yang dilakukan berjumlah tiga bagian karena pada penelitian ini setiap citra uji memiliki tiga huruf yang disambung menjadi satu kata sehingga bila citra dibagi menjadi tiga bagian maka diharapkan akan mendapatkan huruf pada bagian awal, tengah, dan akhir. Karena citra

masukannya berukuran sebesar 150 x 100 piksel maka dilakukan pemotongan yang sama besar yaitu 50 x 100 piksel untuk setiap bagian huruf. *Coding cropping* yang digunakan adalah sebagai berikut

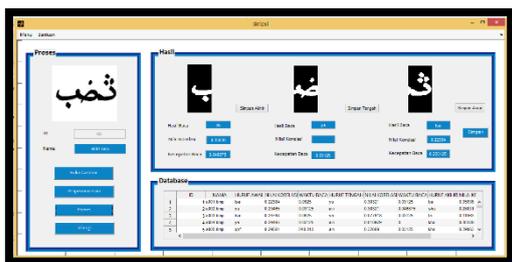
```
cr1=imcrop(gdiff,[1 1 50 100]); %memotong
dari kolom ke 1 baris 1 dengan panjang 50
piksel dan lebar 100 piksel
cr2=imcrop(gdiff,[51 1 50 100]); %memotong
dari kolom ke 51 baris 1 dengan panjang 50
piksel dan lebar 100 piksel
cr3=imcrop(gdiff,[101 1 50
100]); %memotong dari kolom ke 101 baris 1
dengan panjang 50 piksel dan lebar 100
piksel
```

Setelah proses *cropping* selesai dilakukan hasilnya akan ditampilkan pada tiga buah *axes* yang terdapat pada *panel* hasil. Pada *panel* hasil terdapat tiga buah *axes* untuk menampilkan hasil potongan huruf untuk masing-masing bagian yaitu bagian awal, tengah, dan bagian akhir dapat dilihat pada Gambar 5.8. *Coding* yang digunakan untuk menampilkan hasil pemotongan adalah sebagai berikut

```
axes(handles.axes_akhir);
imshow(cr1); %menampilkan hasil potong
akhir
axes(handles.axes_tengah);
imshow(cr2); %menampilkan hasil potong
tengah
axes(handles.axes1);
imshow(cr3); %menampilkan hasil potong
awal
```

GAMBAR 5.8 MENAMPILKAN HASIL DARI PRAPEMROSESAN

Proses pengenalan dapat dilakukan ketika semua prapemrosesan telah dilakukan. Munculnya tiga buah citra pada *axes panel* hasil menandakan bahwa prapemrosesan telah dilakukan. Kemudian lakukan proses pengenalan dengan menekan tombol Proses pada *panel* proses. Setelah itu hasil baca, nilai korelasi, dan kecepatan baca akan ditampilkan pada *panel* hasil, hasil dari proses pengenalan dapat dilihat pada Gambar 5.9.



GAMBAR 5.9 MENAMPILKAN HASIL DARI PENGENALAN HURUF HIJAIYAH SAMBUNG

Hasil baca pada *panel* hasil didapatkan berdasarkan kemiripan antara citra masukkan dengan citra yang ada pada *database template* untuk masing-masing posisi huruf yang menggunakan dari hasil ekstraksi fitur bentuk dengan metode *eccentricity*, dimana kemiripan ini didapatkan dengan menghitung nilai korelasinya menggunakan algoritma *Template Matching Correlation*. Pada sistem terdapat tiga buah *database template* sesuai dengan posisi dari huruf hijaiyah yaitu, posisi awal, tengah, dan akhir dengan citra masing-masing berjumlah 28 citra. Untuk citra hasil potong pada posisi awal, pertama sistem akan melakukan ekstraksi bentuk yang akan menghasilkan matriks 24 x 42 piksel setelah itu sistem akan menghitung nilai korelasinya dengan semua huruf yang ada pada *database template* untuk posisi awal.

Kecepatan baca merupakan hasil pengurangan waktu cpu pada saat sesudah dengan waktu cpu pada saat sebelum proses selesai dilakukan. Begitu juga untuk potongan huruf yang terdapat pada posisi tengah dan akhir masing-masing akan dihitung nilai korelasinya dengan semua huruf yang ada pada *database template* posisi huruf tengah untuk potongan huruf tengah dan *database template* posisi huruf akhir untuk potongan huruf akhir. Nilai korelasi tertinggi dari masing-masing posisi akan ditampilkan pada *field* nilai korelasi, begitu juga dengan kecepatan bacanya. Coding

yang digunakan pada proses ini adalah sebagai berikut

```
[fl re]=lines(re); %melakukan ekstraksi
[L Ne] = bwlabel(imgn);
for n=1:Ne
    [r,c] = find(L==n);
n1=imgn(min(r):max(r),min(c):max(c));
img_r=imresize(n1,[42 24]);
letter=baca(img_r,num_letras);
word=[letter];
end
sem=corr2(templates{1,n},imgn); %menghitung
nilai korelasi dengan semua huruf yang
ada pada templates
vd=find(comp==max(comp)); %mencari nilai
tertinggi
set(handles.nilai_rata_awal,'String',sem);
%menampilkan nilai korelasi
set(handles.hasilbacal,'string',word); %me
nampilkan hasil
runtimeproses1=cputime-
runtimeproses1; %menghitung waktu
kecepatan baca
set(handles.runtime,'String',runtimeproses
1); %menampilkan kecepatan baca
```

Informasi dari hasil proses seperti hasil baca, nilai korelasi, kecepatan baca serta nama file dapat disimpan ke dalam *database Mysql* dengan menekan tombol Simpan yang terdapat di kanan pada *panel* hasil, jika informasi berhasil disimpan maka sistem akan menampilkan kotak dialog yang menyatakan bahwa data berhasil disimpan, kotak dialog dapat dilihat pada Gambar 5.10. Coding untuk menyimpan informasi ke dalam *database Mysql* adalah sebagai berikut

```
datainsert(conn,'tbl_arab',{'nama','huruf_
awal','wb_awal','huruf_tengah','wb_tengah'
,'huruf_akhir','wb_akhir'},{nama,
hasilbacal1, runtime, hasilbacal2, runtime2,
hasilbacal3, runtime3}); %memasukkan data
ke dalam tabel tbl_arab dengan field nama,
huruf_awal, wb_awal, huruf_tengah,
wb_tengah, huruf_akhir, dan wb_akhir
```



GAMBAR 5.10 KOTAK DIALOG BERHASIL SIMPAN

C. Pengujian Sistem

1. Pengujian *Black Box*

Berikut adalah kasus untuk menguji perangkat lunak yang telah dibangun menggunakan metode *black box*.

TABEL 5. 1 TABEL BLACKBOX TESTING

Halaman	Skenario	Harapan	Hasil
Halaman Utama	Klik menu Menu	Sistem menampilkan submenu Pengenalan Huruf Hijaiyah dan submenu Keluar	Sukses
	Klik menu Bantuan	Sistem menampilkan submenu Petunjuk dan submenu Tentang	Sukses
Halaman Pengenalan Huruf Hijaiyah	Klik tombol Buka Gambar	Sistem akan menampilkan dokumen dari komputer dan membuka gambar dengan ekstensi *.bmp. Nama file terletak di sebelah lebel Nama	Sukses
	Klik tombol Prapemrosesan	Sistem akan melakukan proses awal yaitu <i>thresholding</i> kemudian penipisan citra, dan segmentasi citra dengan melakukan pemotongan citra menjadi tiga bagian dan menampilkan a ke dalam panel Hasil	Sukses
	Klik tombol Proses	Sistem akan melakukan proses pencocokan citra masukan dengan citra yang ada dalam <i>template</i> kemudian sistem akan menampilkan hasil pengenalan, nilai korelasi, dan kecepatan baca dari masing-masing bagian citra	Sukses

	Klik tombol Ulangi	Sistem akan memuat ulang (<i>reset</i>) semua panel baik panel Proses maupun panel Hasil	Sukses
	Klik tombol Simpan Awal	Sistem akan menyimpan citra potongan bagian awal dalam ekstensi *.bmp	Sukses
	Klik tombol Simpan Tengah	Sistem akan menyimpan citra potongan bagian awal dalam ekstensi *.bmp	Sukses
	Klik tombol Simpan	Sistem akan menyimpan informasi yang ada pada panel hasil ke dalam <i>database</i> MySQL	Sukses
	Klik menu Menu	Sistem menampilkan submenu Halaman Utama dan submenu Keluar	Sukses
	Klik menu Bantuan	Sistem menampilkan submenu Petunjuk dan Tentang	Sukses
Halaman Keluar	Klik tombol Yes	Sistem akan tertutup	Sukses
	Klik tombol No	Sistem akan kembali ke halaman sebelumnya	Sukses
Halaman Petunjuk	Klik menu Menu	Sistem menampilkan submenu Halaman Utama, Halaman Pengenalan Huruf Hijaiyah dan submenu Keluar	Sukses
	Klik menu Bantuan	Sistem menampilkan submenu Tentang	Sukses
Halaman Tentang	Klik menu Menu	Sistem menampilkan submenu Halaman Utama, Halaman Pengenalan Huruf Hijaiyah dan submenu Keluar	Sukses
	Klik menu Bantuan	Sistem menampilkan submenu Bantuan	Sukses

2. Uji Kelayakan Sistem

Uji kelayakan sistem dilakukan dengan cara mengukur persentase pengenalan sistem dalam mengenali huruf Hijaiyah sambung tulisan tangan dan hasil keluaran sistem dengan menghitung persentase dari jumlah data uji yang dikenali/jumlah data uji total. Hasil pengujian dapat dilihat pada tabel 5.2.

TABEL 5.2 HASIL PENGUJIAN SAMPEL

No.	Nama Citra	Posisi Huruf Yang Dikenali Sistem			Akurasi Data
		Awal	Tengah	Akhir	
1.	A001	Tsa	Ya	Ba	66,67%
2.	A002	Ya	Ain	Sha	66,67%
3.	A003	Tsa	Ya	Ta	33,33%
4.	A004	Ya	Ain	Kho	66,67%
5.	A005	Qof	Ain	Kho	33,33%
6.	A006	Ya	Ain	To	100%
7.	A007	Ha	Tsa	Kaf	33,33%
8.	A008	Jim	Ta	Nun	33,33%
9.	A009	Sha	Zho	Za	66,67%
10.	A010	Sha	Ain	Rha	33,33%
11.	A011	To	Ain	Ha	66,67%
12.	A012	Qof	Ya	Ain	66,67%
13.	A013	Rha	Ya	Lam	66,67%
14.	A014	Ya	Jim	Jim	33,33%
15.	A015	Kaf	Ain	Rha	33,33%
16.	A016	Sha	Jim	Zho	100%
17.	A017	Ya	Mim	Za	66,67%
18.	A018	Qof	Jim	Za	66,67%
19.	A019	Tsa	Ya	Zho	33,33%
20.	A020	Qof	Ya	Ha	66,67%
21.	A021	Ain	Jim	Ha	33,33%
22.	A024	Ain	Kaf	Alif	66,67%
23.	A025	Rha	Mim	Za	33,33%
24.	A026	Kaf	Ya	Kaf	33,33%
25.	A027	Tsa	Ya	Alif	100%
26.	A028	Sha	Zho	Kho	66,67%
Rata-rata akurasi					56,41%

Tabel 5.2 menunjukkan bahwa di dalam sebuah citra terdapat tiga buah huruf, dimana setiap citra dibagi menjadi 3 bagian yaitu awal, tengah, dan akhir.

Pada tabel 5.2 dapat dilihat bahwa presentase rata-rata akurasi adalah sebesar 56,41%. Tingkat akurasi ini didapatkan karena tulisan tangan seseorang mempunyai bentuk yang berbeda untuk setiap kali penulisan, sehingga bentuk dari satu huruf bisa memiliki perbedaan dengan huruf yang sama. Hal itu menyebabkan hasil dari ekstraksi ciri bentuk yang menggunakan parameter *eccentricity* pada huruf tersebut berbeda, hasil yang didapatkan pada saat dilakukan proses pencarian kemiripan menggunakan algoritma *Template Matching Correlation* dengan citra yang ada pada template menjadi berbeda dengan huruf yang seharusnya.

VI. KESIMPULAN

Berdasarkan hasil dan pembahasan yang telah dijabarkan sebelumnya, maka kesimpulan yang dapat diambil adalah sebagai berikut.

1. Penelitian ini telah menghasilkan sebuah aplikasi pengenalan pola tulisan tangan berbasis desktop dengan studi kasus Huruf Hijaiyah Sambung Menggunakan Algoritma *Template Matching Correlation*.
2. Aplikasi Pengenalan Tulisan Tangan Huruf Hijaiyah Sambung Menggunakan Algoritma *Template Matching Correlation* memiliki tingkat akurasi yang rendah yaitu sebesar 56,41% berdasarkan 26 uji coba sedangkan untuk rata-rata akurasi pada tiga tulisan tangan yang berbeda adalah sebesar 4,44% dari total 30 uji coba.
3. Kelemahan dari sistem ini adalah metode *cropping* yang digunakan hanya dapat membagi citra menjadi tiga bagian yang sama besar berdasarkan nilai pikselnya, sistem

tidak dapat melakukan *cropping* dengan ukuran yang dinamis sesuai dengan ukuran masing-masing huruf pada posisi tertentu.

4. Faktor yang menyebabkan tingkat akurasi yang rendah adalah algoritma yang digunakan terlalu sederhana dimana semua nilai dijumlahkan dan dihitung nilai akurasinya berdasarkan pikselnya, sedangkan bisa saja terjadi jumlah nilai korelasi yang sama antar satu huruf dengan huruf yang lainnya, dan juga data yang digunakan merupakan data yang tergolong sulit karena untuk setiap karakter dari huruf hijaiyah mempunyai bentuk yang berbeda dan juga terdapat beberapa huruf yang memiliki ciri yang hampir mirip misalnya jumlah titik pada suatu huruf.

VII. SARAN

Berdasarkan analisa dan perancangan sistem, implementasi, dan pengujian sistem, maka saran yang dapat diberikan untuk pengembangan penelitian selanjutnya adalah sebagai berikut:

1. Penggunaan algoritma *Template Matching Correlation* kurang efektif digunakan untuk mengenali huruf hijaiyah sambung karena tingkat keberhasilan yang didapatkan rendah. Maka dari itu disarankan untuk menggunakan ekstraksi fitur yang lain seperti fitur tekstur yang dipadukan dengan metode pencocokan lain agar didapatkan hasil yang lebih akurat.
2. Untuk metode pemotongan huruf untuk tiap huruf sebaiknya berdasarkan ukuran huruf yang dinamis sesuai dengan ukuran yang ada

pada citra uji bukan berdasarkan ukuran piksel yang ditetapkan.

3. Pengembangan sistem yang selanjutnya diharapkan mampu mengatasi perbedaan ketebalan setiap tulisan.
4. Pengembangan sistem untuk pengenalan huruf secara real time dengan mengenali huruf yang ditulis manual di layar komputer

REFERENSI

- [1] A. I. Musa, "AL DAN HURUF HIJAIYAH," 2012.
- [2] R. Efendi, E. P. Purwandari dan M. A. Aziz, "APLIKASI PENGENALAN HURUF HIJAIYAH BERBASIS MARKER AUGMENTED REALITY PADA PLATFORM ANDROID," *Jurnal Pseudocode*, vol. 2, no. 2, pp. 124-134, 2015.
- [3] D. Putra, "Pengolahan Citra Digital," Penerbit Andi, Yogyakarta, 2010.
- [4] M. C. Wijaya dan A. Prijono, Pengolahan Citra Digital Menggunakan Matlab, Bandung: Penerbit Informatika, 2007.
- [5] C. N. Santosh K.C., A Comprehensive Survey On On-Line Handwriting Recognition Technology and Its Real Application to The Nepalese Natural Handwriting, Kathmandu University Journal of Science, Engineering, and Technology, 2009.
- [6] R. Brunelli, "Template Matching Techniques In Computer Vision," John Willey & Sons. Inc, 2006.
- [7] Sugiyono, Metode Penelitian Kuantitatif, Kualitatif, dan R&D, Bandung: Alfabeta, 2014.