

# PENGENALAN AKSARA BALI DENGAN PENDEKATAN METODE DIRECTION FEATURE DAN AREA BINARY OBJECT FEATURE

Ni Made Ari Pratiwi<sup>1</sup>  
aripratiwi@gmail.com

Widi Hapsari<sup>2</sup>  
widi@ukdw.ac.id

Theresia Herlina R.<sup>3</sup>  
herlina@ukdw.ac.id

## *Abstract*

*The rise of technology has been contributing advances to science, and to human being in making jobs far much easier to do, including pattern recognition. This research focused on character recognition by developing a system capable to recognise images of printed Balinese traditional character, which has a distinct feature of having perceptually similar characters, where each others are often differentiated only by a small stroke or a curve.*

*The system itself took several processes to recognise a character. First, the image containing Balinese characters is preprocessed. Afterwards, two object features are extracted from the image: Direction and Binary Object Area. Both features then tested for similarity using Euclidean distance with the same features already obtained from the control images.*

*From 573 characters tested to the system, 559 are recognized as characters and 526 are correctly recognized as the right character, which yields an overall accuracy of 91.8%. Recognition results are dependent to character spacing condition.*

**Kata kunci :** *Pengenalan Pola, Direction Feature, Area Binary Object Feature, Preprocessing.*

## **1. Pendahuluan**

Perkembangan teknologi sangat banyak memberi pengaruh terhadap perkembangan ilmu pengetahuan. Dan kebutuhan manusia akan teknologi untuk mempermudah melakukan suatu pekerjaan begitu tinggi. Salah satunya adalah dalam hal pengenalan pola. Sistem pengenalan saat ini banyak berkembang dan juga dimanfaatkan seperti pengenalan sidik jari, pengenalan wajah yang berupa image, pengenalan suara, ataupun tulisan. Salah satu persoalan di bidang komputer yang banyak menjadi perhatian adalah analisis citra (*image analysis*). Analisis citra merupakan ilmu yang membahas tentang algoritma-algoritma dan teknik-teknik yang diterapkan pada citra untuk menghasilkan deskripsi yang dikenali komputer.

Pengenalan aksara yang dijadikan objek dalam penelitian ini adalah Aksara Bali. Aksara Bali memiliki karakteristik yang menarik dimana beberapa karakter memiliki ciri yang hampir sama dan hanya dibedakan oleh beberapa coretan atau lekukan garis. Oleh karena itu dibuatlah suatu sistem yang dapat mengenali kalimat Aksara Bali dengan nama aksaranya, dimana pembelajaran Aksara Bali selalu ada dalam kurikulum disekolah.

## **2. Landasan Teori**

### **2.1 Pengolahan Citra Digital**

---

<sup>1</sup> Program Studi Teknik Informatika, UKDW Yogyakarta

<sup>2</sup> Program Studi Teknik Informatika, UKDW Yogyakarta

<sup>3</sup> Program Studi Teknik Informatika, UKDW Yogyakarta

Pengolahan citra digital adalah sebuah disiplin ilmu yang mempelajari hal-hal berkaitan dengan perbaikan kualitas gambar (peningkatan kontras, transformasi warna, restorasi citra), transformasi gambar (rotasi, translasi, skala, transformasi geometrik), melakukan proses penarikan informasi atau deskripsi objek atau pengenalan objek yang terkandung pada citra, melakukan kompresi atau reduksi data untuk tujuan penyimpanan data, transmisi data, dan waktu proses data. Input dari pengolahan citra adalah citra, sedangkan output-nya adalah citra hasil pengolahan. (Sutoyo.dkk, 2009, hal.5)

## 2.2 Peningkatan Kualitas Citra

### A. Transformasi Citra Warna Menjadi Citra *Grayscale*

Citra *grayscale* merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap pikselnya, dengan kala lain nilai dari RED = BLUE = GREEN. Nilai tersebut digunakan untuk menunjukkan tingkat intensitas. Warna yang dimiliki adalah hitam, keabuan, dan putih. Tingkat keabuan disini merupakan warna abu dengan berbagai tingkat dari hitam hingga mendekati putih. (Putra,2010, hal.40)

Citra warna dapat diubah menjadi citra *grayscale* dengan cara menghitung rata-rata elemen warna *Red*, *Green*, dan *Blue*. Secara matematis perhitungannya adalah sebagai berikut. (Sutoyo.dkk, 2009, hal.32)

$$fo(x, y) = \frac{f_i^R(x,y) + f_i^G(x,y) + f_i^B(x,y)}{3} \quad [1]$$

### B. Operasi Ambang Batas (*Thresholding*)

*Thresholding* atau pengambangan akan menghasilkan citra biner, yaitu citra yang memiliki dua nilai tingkat keabuan yaitu hitam dan putih. Secara umum memiliki proses pengembangan citra *grayscale* untuk menghasilkan citra biner adalah sebagai berikut.

$$g(x, y) = \begin{cases} 1 & \text{if } f(x, y) \geq T \\ 0 & \text{if } f(x, y) < T \end{cases} \quad [2]$$

Dengan  $g(x,y)$  adalah citra biner dari citra *grayscale*  $f(x,y)$ , dan T menyatakan nilai ambang (*threshold*). Nilai T memegang peranan sangat penting dalam proses pengembangan. Kualitas hasil citra biner sangat tergantung dari nilai T yang digunakan.

## 2.3 Pengurangan *Noise*

Pengurangan *noise* merupakan suatu proses untuk mereduksi atau mengurangi *noise* pada sebuah citra digital. Filter median merupakan *order-statistic filter* yang paling banyak dikenal. Filter median menghitung nilai dari setiap piksel baru, yaitu nilai pada piksel pusat koordinat *sliding window*. Untuk ukuran window  $m$  baris dan  $n$  kolom maka banyaknya piksel dalam *window* adalah  $m \times n$  baris. Cara kerja *filter* ini dirumuskan sebagai berikut.

$$f(x, y) = \text{median}_{(s,t) \in S_{x,y}} \quad [3]$$

## 2.4 Operasi *Opening* dan *Closing*

Operasi *opening* merupakan operasi erosi yang dilanjutkan dengan operasi dilasi secara berturut-turut dengan elemen penstruktur yang sama. Operasi ini cenderung membuka atau menutup bentuk – bentuk tipis pada objek. Gunanya adalah untuk menghapus *noise* dengan bentuk yang sesuai dengan elemen penstruktur yang digunakan. Selain itu, operasi ini juga digunakan untuk memisahkan objek-objek yang menempel satu sama lain. Secara matematis operasi *opening* dapat dituliskan sebagai berikut.

$$A \circ B = (A \ominus B) \oplus B \quad [4]$$

Operasi *closing* adalah kebalikan dari operasi *opening*, yaitu operasi dilasi yang dilanjutkan dengan operasi erosi secara berturut-turut dengan elemen penstruktur yang sama. Akibat operasi ini adalah menutupnya celah-celah atau lubang lebih kecil dari elemen

penstruktur. Secara matematis operasi *closing* dapat dituliskan sebagai berikut. (Sutoyo.dkk, 2009, hal.212)

$$A \bullet B = (A \oplus B) \ominus B \quad [5]$$

Hasil operasi *closing* hampir mirip dengan operasi dilasi yakni memperbesarr batas luar dari objek *foreground* dan juga menutup lubang kecil yang terletak di tengah objek, namun hasil operasi *closing* tidak sebesar hasil dilasi. Hasil dilasi akan menyebabkan pembengkakan bentuk keseluruhan objek. Efek ini dapat dikurangi dengan menerapkan proses erosi setelah proses dilasi tersebut. (Putra,2010, hal.295)

### 2.5 Perangkaan (*Thinning*)

Perangkaan atau *thinning* adalah suatu proses pengurangan komponen-komponen citra dengan tujuan untuk mendapatkan informasi yang paling mendasar.

Metode *thinning* yang digunakan dalam proses ini adalah *Zhan-Suen method*. Algoritma dari metode ini menggunakan metode iterasi, dimana berarti nilai yang baru didapat dari proses sebelumnya. *Contour point* adalah setiap piksel dengan nilai 1 dan memiliki setidaknya satu dari 8-tetangganya memiliki nilai 0. Gambar 1 mengilustrasikan *countour point P1* dan 8-tetangganya.

P9	P2	P3
P8	<b>P1</b>	P4
P7	P6	P5

Gambar 1

Contoh piksel p1 dengan tetangganya

Dengan informasi ini, langkah pertama adalah menandai *countour point p* untuk dihapus jika semua kondisi ini dipenuhi :

1.  $2 \leq N(P1) \leq 6$
2.  $S(P1) = 1$
3.  $P2 \text{ OR } P4 \text{ OR } P6 = 0$
4.  $P4 \text{ OR } P6 \text{ OR } P8 = 0$

dimana  $N(P1)$  adalah jumlah tetangga dari  $P1$  yang tidak 0, yaitu

$$N(P1) = P2 + P3 + P4 + \dots + P9 \quad [6]$$

dan  $S(P1)$  adalah jumlah dari transisi 0 – 1 pada urutan  $P2, P3, P4, \dots, P8, P9$ . Pada langkah kedua, kondisi (1) dan (2) sama dengan langkah pertama, sedangkan kondisi (3) dan (4) diubah menjadi :

3.  $P2 \text{ OR } P4 \text{ OR } P8 = 0$
4.  $P2 \text{ OR } P6 \text{ OR } P8 = 0$

Langkah pertama dilakukan terhadap semua *border pisel* di citra. Jika salah satu dari keempat kondisi di atas dipenuhi atau dilanggar maka nilai piksel yang bersangkutan tidak diubah, sebaliknya jika semua kondisi tersebut dipenuhi maka piksel tersebut ditandai untuk penghapusan.

Piksel yang telah ditandai tidak akan dihapus sebelum semua *border points* selesai diproses. Hal ini berguna untuk mencegah perubahan struktur data. Setelah langkah 1 selesai dilakukan untuk semua *border points* maka dilakukan penghapusan untuk titik yang telah ditandai (diubah menjadi 0). Setelah itu dilakukan langkah 2 pada data hasil dari langkah 1 dengan cara yang sama dengan langkah 1 sehingga, dalam satu kali iterasi urutan algoritmanya terdiri dari:

1. Menjalankan langkah 1 untuk menandai *border points* yang akan dihapus,
2. Hapus titik-titik yang ditandai dengan menggantinya menjadi angka 0,

3. Menjalankan langkah 2 pada sisa *border points* yang pada langkah 1 belum dihapus lalu yang sesuai dengan semua kondisi yang seharusnya dipenuhi pada langkah 2 kemudian ditandai untuk dihapus,
4. Hapus titik- titik yang ditandai dengan menggantinya menjadi angka 0. (Zurnawita dan Suar, 2009)

## 2.6 Segmentasi

Segmentasi adalah proses pemecahan citra ke dalam obyek-obyek yang terkandung di dalamnya (Kasturi *et al.*, 2002). Proses segmentasi pada citra dokumen dapat dilakukan dengan mempergunakan histogram citra serta profil proyeksi dari citra tersebut.

### Profil Proyeksi

Widiarti (2007) apabila terdapat sebuah citra teks biner  $S(N,M)$ , di mana  $N$  menyatakan banyaknya baris citra dan  $M$  adalah banyaknya kolom citra seperti ditunjukkan pada Gambar 2, maka dapat ditentukan profil vertikal dan profil horisontal dari citra teks tersebut. Profil vertikal adalah banyaknya piksel hitam yang tegak lurus sumbu  $y$ , sedangkan profil horisontal adalah banyaknya piksel hitam yang tegak lurus sumbu  $x$ . Profil vertikal direpresentasikan dengan suatu vektor ( $P_v$ ) berukuran  $N$ . Profil vertikal pada baris ke- $i$ , yaitu ( $P_v [i]$ ), didefinisikan sebagai berikut:

[7]

Sedangkan profil horisontal direpresentasikan dengan suatu vektor ( $P_h$ ) berukuran  $M$ . Profil horisontal pada kolom ke- $j$ , yaitu ( ), didefinisikan sebagai berikut:

[8]



Gambar 2  
Profil proyeksi horisontal dan vertikal

## 2.7 Pengenalan Pola

Proses pengenalan pola terdiri dari tiga fase utama yaitu segmentasi citra, ekstraksi ciri dan klasifikasi. Fase segmentasi citra adalah suatu fase yang bertujuan memisahkan citra menjadi pusat perhatian dari bagian citra lainnya. Fase ekstraksi ciri adalah fase dilakukannya pengukuran terhadap citra. Fase ekstraksi ciri akan menghasilkan beberapa ciri yang diwujudkan dalam bentuk suatu vektor ciri. Vektor ciri hasil ekstraksi ciri ini digunakan oleh fase klasifikasi. Output dari fase klasifikasi adalah suatu keputusan termasuk kelas apakah suatu obyek itu. (Widiarti,2009)

### 2.7.1 Pendekatan Direction Feature

Metode pertama adalah memakai ciri-ciri khusus yang dimiliki Aksara Bali. Dimana Aksara Bali banyak memiliki lekukan-lekukan serta garis-garis khusus sebagai penanda karakteristik suku kata tertentu. Dan akan digunakan metode ekstraksi *Direction Feature* untuk menemukan ciri dari setiap obyek, dengan cara penentuan arah garis-garis dari piksel-piksel *foreground* yang ada di dalam citra karakter. Kemudian mengubahnya menjadi nilai-nilai vektor yang diperlukan untuk proses selanjutnya. Hal tersebut yang menyebabkan arah dari elemen-elemen garis di dalam karakter dapat dijadikan ciri khusus dari citra karakter yang dikenali, lihat tabel 1. Di dalam penelitian ini digunakan empat garis yang berbeda sebagai ciri yaitu : horisontal, diagonal kanan, vertikal, dan diagonal kiri.

Tabel 1.  
Orientasi arah garis dalam direction feature

<b>Bentuk</b>		/	—	\
<b>Arah</b>	vertikal	diagonal kanan	horisontal	diagonal kiri
<b>Nilai</b>	2	3	4	5

Untuk melakukan pelabelan arah pada masing masing piksel dapat dilakukan dengan cara sebagai berikut:

1. Lakukan pengecekan secara raster dari kiri ke kanan
2. Apabila menemukan sebuah piksel *foreground* maka lakukan pengecekan dengan melihat tetangga dari piksel tersebut.
3. O adalah piksel yang akan dicek, kemudian pengecekan dilakukan dari x1 – x8. Apabila pada posisi tetangga dari x1 sampai x8 ditemukan piksel foreground, maka ubahlah nilai O menjadi nilai arah berdasarkan aturan dibawah ini :
  - Jika pada posisi x1 atau x5 maka nilai arah adalah 5
  - Jika pada posisi x2 atau x6 maka nilai arah adalah 2
  - Jika pada posisi x3 atau x7 maka nilai arah adalah 3
  - Jika pada posisi x4 atau x8 maka nilai arah adalah 4

Tabel 2.  
Matrik ketetanggaan dalam penentuan nilai label

<b>X1</b>	<b>X2</b>	<b>X3</b>
<b>X8</b>	<b>O</b>	<b>X4</b>
<b>X7</b>	<b>X6</b>	<b>X5</b>

<b>5</b>	<b>2</b>	<b>3</b>
<b>4</b>	<b>X</b>	<b>4</b>
<b>3</b>	<b>2</b>	<b>5</b>

Tabel 2 menunjukkan matrik ketetanggaan dalam penentuan nilai label pada 8 titik tetangga. Setelah proses ekstraksi dan pelabelan dilakukan, maka akan dilakukan pengecekan terhadap data yang terdapat pada basis data. Dimana dilakukan perbandingan dan dicari nilai *Euclide* terkecil dari dua metode yang dipakai.

**2.7.2 Pendekatan Area Binary Object Feature**

Terdapat banyak cara untuk memperoleh sifat dari suatu karakter. Salah satu cara yang dapat dipakai adalah dengan mencari sifat dari sekelompok bagian karakter (unit). Dimana sifat ditandai dengan sebuah angka. Secara garis besar, salah satu cara yang dipakai untuk ekstraksi fitur adalah *Area Binary Object Feature* dimana :

- a. Sebuah citra karakter dibagi menjadi  $N \times N$  bagian sehingga menjadi  $N*N$  bagian yang disebut dengan unit.
- b. Setiap unit dalam keseluruhan citra akan dicari banyaknya piksel obyek.

**2.7.3 Langkah Klasifikasi**

Cara untuk melakukan klasifikasi dengan pendekatan tersebut adalah sebagai berikut :

- a. Nilai setiap unit pada lokasi yang bersesuaian dari karakter yang akan dikenali dibandingkan dengan nilai setiap unit karakter yang terdapat pada basis data. Perbandingan ini dilakukan dengan menggunakan modifikasi jarak dari jarak *Euclide*.

Jarak *Euclidean*  $De(p,q)$  dari piksel  $p,q$  dengan koordinat masing-masing titik  $(x,y)$  dan  $(s,t)$ , didefinisikan sebagai berikut :

$$d_{ij} = \sqrt{\sum_{k=1}^n (x_{ik} - x_{jk})^2} \quad [9]$$

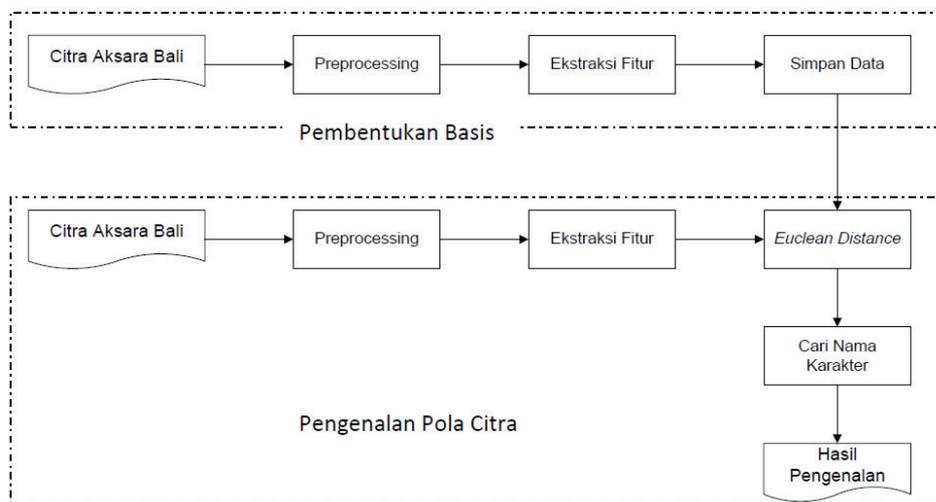
*Euclidean Distance* adalah metrik yang paling sering digunakan untuk menghitung kesamaan 2 vektor. *Euclidean Distance* menghitung akar dari kuadrat perbedaan 2 vektor (*root of square distance between 2 vectors*). (Putra, 2010, hal.311)

- b. Dicari jarak yang paling minimum dari nilai-nilai hasil perbandingan pada langkah satu(a). Jarak yang paling minimum dan masih dibawah batas atas nilai jarak yang diperoleh akan dipilih sebagai karakter yang paling mendekati karakter yang akan dikenali.

### 3. Implementasi Sistem

#### 3.1. Blok Diagram Sistem

Gambar 3 menggambarkan struktur hubungan antara proses-proses utama dalam sistem pengenalan Aksara Bali pada citra.



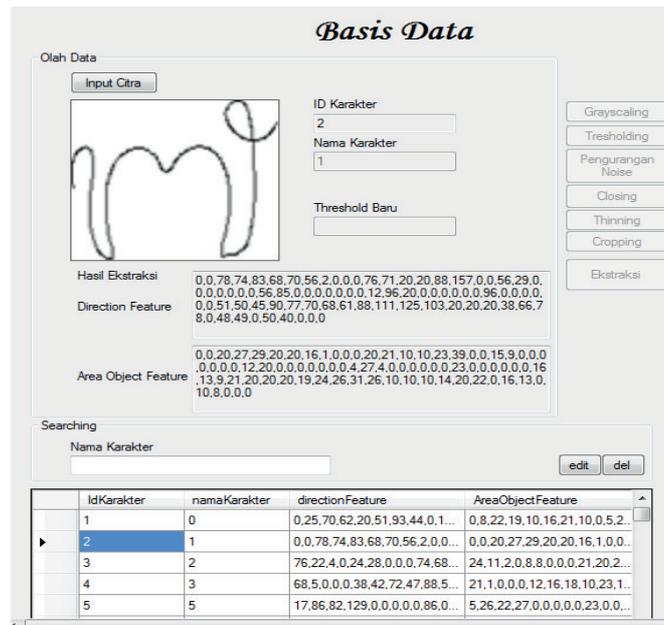
Gambar 3. Hubungan antara bagian

Gambar 3 menggambarkan hubungan antara bagian pembentukan basis data dan pengenalan pola citra. Proses diawali dengan *inputan* citra lalu dilakukan *preprocessing* setelah itu dilakuan ekstraksi untuk mendapatkan fitur dari citra Aksara Bali tersebut. Pada proses pertama fitur yang didapat disimpan dalam basis data dan diberi nama setiap karakternya. Pada proses kedua, fitur yang didapat dibandingkan dengan fitur-fitur yang ada dalam basis data menggunakan perhitungan *Euclidean Distance* kemudian akan menghasilkan kumpulan nama aksara citra Aksara Bali tersebut.

#### 3.2. Implementasi Rancangan Antar muka

##### A. Implementasi Rancangan Antar Muka Bagian Pembentukan Basis Data

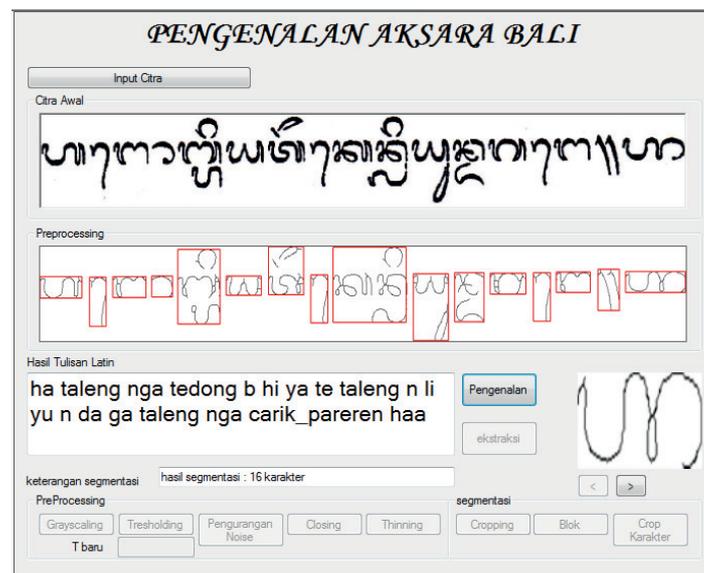
Antar muka proses pembentukan bais data digunakan untuk melakukan pembentukan basis data Aksara Bali. Implementasi antar muka bagian pembentukan basis data ditunjukkan pada Gambar 4.



Gambar 4. Implementasi rancangan antar muka pembentukan basis data

### B. Implementasi Rancangan Antar Muka Bagian Pengenalan

Antar muka bagian pengenalan digunakan untuk melakukan proses pengenalan terhadap citra yang diinputkan. Gambar 5 menunjukkan implementasi rancangan antar muka bagian pengenalan.



Gambar 5. Implementasi rancangan antar muka bagian pengenalan

## 4. Analisis Sistem

Tabel 3 menunjukkan hasil pengenalan karakter penyusun citra input dengan citra karakter yang disimpan dalam basis data. Parameter Nama Citra sebagai perwakilan nama citra yang diuji, Karakter Sebenarnya adalah jumlah karakter dengan perhitungan manual dalam citra yang diinputkan. Katacter Terdeteksi adalah jumlah karakter yang dideteksi oleh sistem. Karakter Dikenali dengan Benar adalah jumlah karakter yang terdeteksi dan dengan benar dikenali nama karakternya. Karakter Dikenali dengan Salah, adalah jumlah karakter sebenarnya dikurangi dengan Karakter Dikenali dengan Benar. Persentasi

Kebenara merupakan persentasi dari Karakter Dikenali dengan Benar dan Karakter Sebenarnya.

Tabel 3.

Hasil pengujian karakter buku Widya Guna Bahasa Bali (hal 52)

No	Nama Citra	Karakter Sebenarnya	Karakter Terdeteksi	Karakter Dikenali dengan Benar	Karakter Dikenali dengan Salah	Persentase Kebenaran
1	Baris1-bag1	18	18	18	0	100,00%
2	Baris1-bag2	18	18	18	0	100,00%
3	Baris2-bag1	16	16	16	0	100,00%
4	Baris2-bag2	15	14	13	2	86,67%
5	Baris3-bag1	17	16	16	1	94,12%
6	Baris3-bag2	19	18	17	2	89,47%
7	Baris4-bag1	16	15	14	2	87,50%
8	Baris4-bag2	16	16	15	1	93,75%
9	Baris5-bag1	19	19	17	2	89,47%
10	Baris5-bag2	18	17	15	3	83,33%
11	Baris6-bag1	18	18	17	1	94,44%
12	Baris6-bag2	16	16	14	2	87,50%
13	Baris7-bag1	19	18	16	3	84,21%
14	Baris7-bag2	16	15	14	2	87,50%
15	Baris8-bag1	19	19	18	1	94,74%
16	Baris8-bag2	22	22	22	0	100,00%
17	Baris9-bag1	21	21	21	0	100,00%
18	Baris9-bag2	14	13	12	2	85,71%
19	Baris10-bag1	16	16	15	1	93,75%
20	Baris10-bag2	20	19	17	3	85,00%
21	Baris11-bag1	15	15	12	3	80,00%
22	Baris11-bag2	17	16	14	3	82,35%
23	Baris12-bag1	19	19	19	0	100,00%
24	Baris12-bag2	19	19	19	0	100,00%
25	Baris13-bag1	22	22	22	0	100,00%
26	Baris13-bag2	14	14	13	1	92,86%
27	Baris14-bag1	21	19	16	5	76,19%
28	Baris14-bag2	15	15	14	1	93,33%
29	Baris15-bag1	21	21	21	0	100,00%
30	Baris15-bag2	19	17	15	4	78,95%
31	Baris16-bag1	24	24	23	1	95,83%
32	Baris16-bag2	14	14	13	1	92,86%
<b>TOTAL</b>		<b>573</b>	<b>559</b>	<b>526</b>	<b>47</b>	<b>91,80%</b>

Berdasarkan Tabel 3, total karakter sebenarnya citra adalah 573 karakter. Karakter aksara yang benar dikenali dengan benar adalah 526 karakter, dan karakter yang dikenali dengan salah sebanyak 47 karakter, dengan persentase keberhasilan 91,80%. Perbandingan jumlah karakter sebenarnya dengan karakter terdeteksi dipengaruhi oleh jarak antar karakter.

Tabel 4.  
Pengaruh tidak dilakukannya salah satu preprocessing terhadap pengenalan citra

NO	Tanpa Preprocessing	Karakter Sebenarnya	Karakter Terdeteksi	Karakter Dikenali dengan Benar	Karakter Dikenali dengan Salah	Persentase Kebenaran
<b>Baris1-bag1</b>						
1	Grayscale	18	18	18	0	100,00%
2	Thresholding	18	1	0	18	0,00%
3	Pengurangan Noise	18	18	17	1	94,44%
4	Closing	18	18	18	0	100,00%
5	Thinning	18	17	16	2	88,89%
<b>Baris2-bag2</b>						
1	Grayscale	15	14	13	2	86,67%
2	Thresholding	15	1	0	15	0,00%
3	Pengurangan Noise	15	14	11	4	73,33%
4	Closing	15	14	13	2	86,67%
5	Thinning	15	13	10	5	66,67%
<b>Baris4-bag2</b>						
1	Grayscale	16	16	15	1	93,75%
2	Thresholding	16	1	0	16	0,00%
3	Pengurangan Noise	16	16	13	3	81,25%
4	Closing	16	16	14	2	87,50%
5	Thinning	16	14	10	6	62,50%
<b>Baris14-bag1</b>						
1	Grayscale	21	19	17	4	80,95%
2	Thresholding	21	0	0	21	0,00%
3	Pengurangan Noise	21	23	17	6	80,95%
4	Closing	21	19	17	4	80,95%
5	Thinning	21	19	17	4	80,95%

Tabel 4 menunjukkan pengaruh tidak dilakukannya salah satu *preprocessing* terhadap pengenalan citra. Beberapa Sampel yang digunakan adalah Baris1-bag1, Baris2-bag2, Baris4-bag2 serta Baris14-bag1 pada Buku Widya Guna Bahasa Bali (hal 52). Hasil diatas adalah analisa terhadap tidak digunakannya salah satu proses *preprocessing* yang ada pada kolom 2 dan dengan ketentuan proses *preprocessing* lain tetap dilakukan. Perbandingan hasil terhadap dilakukannya semua parameter dapat dilihat pada tabel 5. Terdapat hasil paling buruk jika tidak dilakukan *thresholding* yaitu tidak adanya binerisasi citra. Karena citra diatas merupakan citra yang warna dasar karakternya adalah hitam, maka saat tidak dilakukan *grayscale* persentase hasil sama dengan citra yang menggunakan semua proses *preprocessing*. Tanpa adanya pengurangan *noise* atau *closing* cukup mampu melakukan pengenalan, walaupun hasilnya tidak sama dengan pada tabel 5. Langkah-langkah *preprocessing* dalam pengenalan diperlukan untuk membantu proses pengenalan.

Tabel 5.  
Hasil pengenalan terhadap dilakukannya seluruh preprocessing

No	Nama Citra	Karakter Sebenarnya	Karakter Terdeteksi	Karakter Dikenali dengan Benar	Karakter Dikenali dengan Salah	Persentase Kebenaran
1	Baris1-bag1	18	18	18	0	100,00%
2	Baris2-bag2	15	14	13	2	86,67%
3	Baris4-bag2	16	16	15	1	93,75%
4	Baris14-bag1	21	19	16	5	76,19%

Tabel 6.  
Pengaruh nilai ambang (t) terhadap pengenalan citra

Nilai Ambang (t)	Karakter Sebenarnya	Karakter Terdeteksi	Karakter Dikenali Dengan Benar	Karakter Dikenali Dengan Salah	Persentase Kebenaran
<b>Baris1-bag1</b>					
64	18	18	18	0	100,00%
100	18	18	18	0	100,00%
<b>128</b>	<b>18</b>	<b>18</b>	<b>18</b>	<b>0</b>	<b>100,00%</b>
150	18	18	18	0	100,00%
192	18	18	18	0	100,00%
<b>Baris2-bag2</b>					
64	15	15	13	2	86,67%
100	15	14	13	2	86,67%
<b>128</b>	<b>15</b>	<b>14</b>	<b>13</b>	<b>2</b>	<b>86,67%</b>
150	15	<b>14</b>	<b>13</b>	2	86,67%
192	15	14	12	3	80,00%
<b>Baris4-bag2</b>					
64	16	16	15	1	93,75%
100	16	16	15	1	93,75%
<b>128</b>	<b>16</b>	<b>16</b>	<b>15</b>	<b>1</b>	<b>93,75%</b>
150	16	16	15	1	93,75%
192	16	15	10	6	62,50%
<b>Baris14-bag1</b>					
64	21	19	16	5	76,19%
120	21	19	16	5	76,19%
<b>128</b>	<b>21</b>	<b>19</b>	<b>16</b>	<b>5</b>	<b>76,19%</b>
150	21	20	16	5	76,19%
192	21	19	14	7	66,67%

Dari hasil Tabel 6 yang menunjukkan pengaruh nilai ambang (tr) terhadap pengenalan citra. Sampel yang digunakan sama dengan sampel pada Tabel 4. Pentingnya pemilihan nilai ambang untuk mendapatkan citra yang baik dalam pemrosesan selanjutnya. Pemilihan nilai ambang tergantung pada kualitas citra input. Pada pemilihan T192 didapat hasil yang paling buruk karena semakin banyak noise yang terjadi dan menyebabkan persentase keberhasilan pengenalan berkurang.

Selain faktor *preprocessing* serta pengaruh nilai ambang (T), terdapat faktor jarak antar karakter yang menentukan keberhasilan dalam pengenalan citra.



- Sunarya, I.M.G. (2012). *Pengenalan Aksara Bali Tercetak Pada Citra Dokumen Menggunakan Multi Layer Perceptron Dengan Pembelajaran Backpropagation* (Tesis S2 Ilmu Komputer, Fakultas MIPA, Universitas Gadjah Mada Yogyakarta, 2012).
- Sutoyo, T., Mulyanto, E., Suhartono, V., Nurhayati, O.D., & Wijanarto. (2009). *Teori Pengolahan Citra Digital*. Yogyakarta : Penerbit Andi.
- Widiarti, A.R. (2006). *Pengenalan Citra Dokumen Sastra Jawa Konsep dan Implementasinya* (Tesis S2 Ilmu Komputer, Fakultas MIPA, Universitas Gadjah Mada Yogyakarta, 2012).
- Widiarti, A.R. (2007). Segmentasi Citra Dokumen Teks Sastra Jawa Modern Mempergunakan Profil Proyeksi. *SIGMA*, 10(2), Juli 2007: 167-176.
- Wirawan, Made. (2011). *Pengenalan Aksara Bali pada Pen Screen Berbasis Jaringan Syaraf Tiruan Backpropagation* ( Skripsi S1, Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana,2011), dari SinTA (Sistem Informasi Tugas Akhir).
- Zurnawita dan Suar, Z,. (2009). Algoritma Image Thinning. *Elektron*, 1(1), Politeknik Negeri Padang, 29-37.