

PENGENALAN CITRA AKSARA JAWA PADA PLANG JALAN

Nurdana Ahmad Fadil
Program Studi Informatika
Universitas Islam Indonesia
Yogyakarta
16523172@students.uii.ac.id

Izzati Muhimmah
Program Studi Informatika
Universitas Islam Indonesia
Yogyakarta
Izzati@uui.ac.id

Abstract—Aksara Jawa yang mengalami penurunan minat oleh sebagian anak muda menjadi fokus pada penelitian pengenalan citra aksara Jawa. Hasil penelitian terdahulu mengenai segmentasi huruf non-latin selain aksara Jawa memanglah sudah menunjukkan tingkat akurasi yang tinggi namun pada terkhusus segmentasi aksara Jawa masih ada beberapa kekurangan yang akan diperbaiki dalam penelitian kali ini. Penelitian ini menggunakan citra yang ditangkap oleh kamera sehingga gaya penulisan huruf dari aksara Jawa bisa berbeda-beda bukan dari inputan manual sistem yang menjadikan gaya penulisan aksara Jawa sama seperti yang dilakukan penelitian sebelumnya. Metode yang dilakukanpun terbilang umum digunakan seperti segmentasi, morfologi dan *thining*, namun perlu digaribawahi di sini adalah dalam penyusunan metode tersebut ambang batas yang menjadi tolak ukur dalam metode tersebut sangat diperhatikan sedemikian rupa sehingga mendapatkan objek yang ingin dideteksi. Setelah objek telah ditemukan otomatis oleh sistem maka akan dilakukan ekstraksi ciri pada masing-masing objek dengan cara memotong 9 bagian sama rata dan dihitung jumlah piksel perbagian tersebut. Setelah ekstraksi ciri didapatkan, pencocokan dengan data latih dilakukan, kemudian diperoleh tingkat hasil akhir dengan rata2 akurasi sebesar 75%.

Keywords—aksara, Jawa, citra, pengenalan, objek

I. PENDAHULUAN

Perkembangan teknologi yang ada pada saat ini mengakibatkan semuanya akan ketergantungan dengan teknologi, tanpa dipungkiri seiring perkembangan zaman, muncul stigma “Lambat laun budaya akan ditinggalkan dan digantikannya oleh teknologi”, dengan dasar tersebut, penelitian ini ingin mengubah stigma atau cara berpikir kebanyakan orang, yaitu dengan teknologi, budaya yang telah beratus-ratus tahun diwariskan dari nenek moyang juga bisa kita lestarikan. Penelitian ini akan membuktikan bahwa teknologi bukan penghalang untuk berbudaya dan diharapkan penelitian ini akan memotivasi banyak orang agar tetap bisa melestarikan budaya.

Aksara Jawa adalah salah satu aksara tradisional Nusantara yang digunakan untuk menulis bahasa Jawa. Dalam keseharian penggunaan aksara Jawa mulai digantikan dengan huruf Latin, yang mulai pertama kali dikenalkan Belanda. Seiring berjalannya waktu huruf Latin menjadi huruf nasional atau yang dikenal sebagai bahasa Indonesia. Walaupun penggunaan aksara Jawa mulai menurun drastis akan tetapi di beberapa sekolah, aksara Jawa masih menjadi mata pelajaran muatan lokal yang bertujuan mengenalkan betapa pentingnya melestarikan budaya, namun seiring berkembangnya teknologi mata pelajaran muatan lokal yang mempelajari aksara Jawa mulai ditinggalkan dan dialihkan ke

pelajaran yang berbau teknologi. Dalam kasus ini aksara Jawa sudah mulai hilang karena hilangnya penerus di generasi muda kita. Apabila budaya yang telah diwariskan nenek moyang kita tersebut dapat dikonversikan dalam format digital, maka akan banyak sekali manfaat yang bisa diperoleh. Selain manfaat tersebut, penelitian ini bertujuan untuk membantu orang awam dengan aksara Jawa dan menjadikannya bisa membaca dengan mudah [1]. Akara Jawa dalam penelitian ini berfokus pada plang jalan yang biasanya terdapat di sekitar jalan Yogyakarta.

Aksara Jawa sendiri memiliki huruf yang sangat unik yang masuk dalam kategori huruf non-latin, seperti halnya huruf pada bahasa Jepang, Cina dan Thailand. Haruechaiyasak beserta temannya melakukan penelitian menggunakan teknik segmentasi pada bahasa Thailand dengan berbagai metode dengan capaian tingkat akurasi terbaik 95,79% [2]. Segmentasi pada bahasa Dzongkha dilakukan oleh Norbu beserta kawannya dengan akurasi terbaik 91,5% [3]. Pada kasus aksara Jawa sendiri telah dilakukan oleh Prabantoro yang menggunakan kamus dan karakter statistik seperti jumlah karakter pada kata terpanjang, jumlah karakter pada kata terpendek, dan rata-rata panjang kata [4]. Pada penelitian Prabantoro sendiri memiliki kelemahan yaitu karakter aksara Jawa yang haruslah karakter yang diinputkan secara manual menggunakan sistem. Gaya penulisan huruf pada kasus penelitian Prabantoro ini menjadikan semua karakter aksara Jawa sama persis padahal, pada umumnya karakter aksara Jawa memiliki jenis gaya penulisan yang sangat beragam. Penelitian pengenalan aksara Jawa ini difokuskan untuk mendeteksi citra aksara Jawa, selain itu memang dimaksudkan untuk mengatasi kekurangan yang ada pada penelitian sebelumnya [4].

Penelitian pengenalan citra aksara Jawa ini menjadikan metode-metode sederhana sebagai acuan pendeteksian awal sebuah objek aksara Jawa. Hasil akhirnya akan terlihat berapa tingkat akurasi sistem dalam mengenali objek citra aksara Jawa. Dalam proses penelitian ini selain bertujuan mencari hasil akhir juga bertujuan membantu orang awam dengan aksara Jawa dan menjadikannya bisa membaca dengan mudah.

II. KAJIAN PUSTAKA

A. Aksara Jawa

Bahasa Jawa memiliki jutaan penutur, menjadikannya salah satu bahasa yang paling banyak digunakan di Indonesia dan di dunia pada umumnya [5]. Banyak aplikasi dikembangkan untuk meningkatkan kesadaran dan mendidik masyarakat dalam aksara Jawa, beberapa di antaranya menggunakan deteksi stroke atau pengenalan gambar untuk

menerjemahkan aksara Jawa bagi pengguna. Aksara Jawa di sini memiliki kompleksitas dalam gaya penulisan yang menjadikan basis data atau data *training* di sini lebih beraneka ragam dan sangat bervariasi.

B. Citra

Citra adalah sekumpulan informasi yang diciptakan dalam bentuk 2 dimensi yang merepresentasikan suatu objek dimana isi citra merupakan susunan array dari bilangan real atau bilangan yang sangat kompleks dan diwakilkan dalam bilangan-bilangan bit terhingga. Menurut Munir, citra adalah gambar pada bidang dwimarta (dua dimensi) [6]. Citra sendiri mempunyai dua jenis yaitu, citra kontinu dan citra diskrit. Agar sebuah citra dapat diolah dengan komputer digital, maka suatu citra harus direpresentasikan secara numerik dengan nilai-nilai diskrit. Representasi citra dari fungsi kontinu menjadi nilai-nilai diskrit disebut digitalisasi citra dan citra yang dihasilkan disebut citra digital. Secara umum, citra digital mempunyai bentuk empat persegi panjang dan dimensi ukurannya dinyatakan sebagai tinggi x lebar [6].

C. Pengolahan Citra

Pengolahan citra di sini bertujuan untuk mengambil informasi yang dirasa penting atau yang bisa mewakili dari sebuah objek yang ingin dikenali. Hasil dari pengambilan informasi ini akan sangat bermanfaat ketika nilai yang merepresentasikan objek tersebut akurat dan nantinya hasil akurasi dalam sebuah pencocokan dalam data uji pun akan semakin tinggi.

D. Grayscale

Grayscale merupakan sebuah metode yang mengubah citra warna (RGB) menjadi citra *grayscale* (keabuan). Citra *grayscale* merupakan citra yang merepresentasikan setiap piksel yang hanya satu jenis atau satu nilai warna dalam skala keabuan. Mendapatkan nilai intensitas keabuan bisa dengan cara menjumlahkan 30% dari nilai R (Red dari singkatan RGB), 59% dari G (Green dari singkatan RGB), dan 11% dari B (Blue dari singkatan RGB). Hasil dari penjumlahan tersebut, kemudian akan direpresentasikan dalam setiap *channel Red, Green* dan *Blue* dengan nilai intensitas yang sama.

E. Thresholding

Thresholding yang dimaksud di sini merupakan suatu nilai batasan dalam suatu metode yang mengubah citra *grayscale* ke dalam citra biner. *Thresholding* biasa juga disebut *gray level segmentation* atau pemisahan warna dalam skala keabuan. Nilai batasan dalam *thresholding* dapat kita sesuaikan keinginan kita, dalam kasus ini berarti nilai batasan dalam skala keabuan yang melebihi keinginan kita akan berubah menjadi citra hitam (bernilai 1) dan yang tidak memenuhi batasan yang kita inginkan akan menjadi citra putih (bernilai 0) atau juga bisa sebaliknya.

$$\begin{aligned}
 F(x,y) &= 255, \text{ jika } f(x,y) \geq T \\
 F(x,y) &= 0, \text{ jika } f(x,y) < T \\
 \text{Nilai threshold } T &\text{ adalah} \\
 T &= \frac{f_{\max} + f_{\min}}{2} \\
 &= (240 + 50)/2 = 145
 \end{aligned}$$

F. Segmentasi

Segmentasi adalah langkah pertama untuk memproses bahasa non-latin seperti aksara Jawa. Dalam penelitian [7] tersebut segmentasi yang dilakukan berdasarkan pendekatan huruf per huruf. Pada huruf pertama, akan dicocokkan dengan semua huruf aksara Jawa utama yang mungkin tersegmentasi menggunakan basis data yang telah dibuat. Setelah melakukan pendeteksian huruf per huruf nantinya akan membagi huruf per huruf menjadi beberapa bagian untuk dideteksi citra di setiap bagian.

G. Morfologi

Operasi morfologi dalam kasus ini yaitu suatu proses atau metode yang bertujuan untuk mengubah bentuk objek pada citra asli dan biasanya mengubah menjadi citra *grayscale* ataupun citra biner. Haralick mengenalkan metode dalam proses pengurangan *noise* pada citra biner agar objek dari citra biner tersebut dengan mudah memilah objek yang menjadi pusat perhatian dan menghilangkan objek tidak penting [8]. Pada kasus ini menggunakan metode *filling* dengan rumus.

$$A \circ B = (A \ominus B) \oplus B$$

$$\text{Menjadi } A \circ B = \bigcup_{B_x \in A} B_x$$

H. Thining

Thining adalah sebuah metode atau algoritma yang bertujuan untuk memipihkan suatu objek atau menjadikannya *single piksel*. Proses *thining* ini pada dasarnya mencari sebuah piksel yang dapat mewakili secara garis besar di objek tersebut sehingga memunculkan informasi yang penting dan membuang informasi yang kurang penting. Arcelli dan Sanniti pernah memberikan contoh metode *fast thining algorithm* untuk mengurangi komponen yang dirasa kurang membawa informasi dari objek tersebut [9]. Algoritma ini kemudian menjadi populer dan kemudian banyak metode yang kita pakai sekarang adalah hasil metode yang dikembangkan dari metode tersebut.

$$X_H(p)=1$$

Di mana

$$X_H(p) = \sum_{i=1}^4 b_i$$

$$b_i = \begin{cases} 1, & \text{if } x_{2i-1} = 0 \text{ and } (x_{2i} = 1 \text{ or } x_{2i+1} = 1) \\ 0, & \text{otherwise} \end{cases}$$

x_1, x_2, \dots, x_8 adalah nilai-nilai dari sekitar p , dimulai dengan daerah sekitar timur dan diberi nomor dalam urutan berlawanan arah jarum jam.

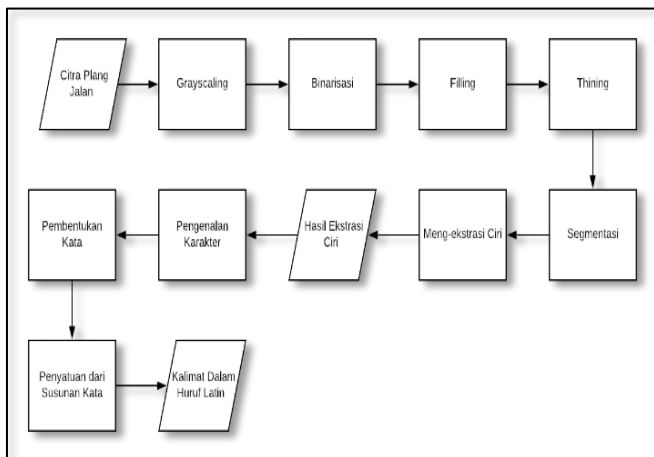
I. Ekstraksi Ciri

Ekstraksi ciri adalah sebuah metode pengambilan informasi dari hasil akhir semua metode, yang nantinya akan disimpan dan akan menjadi beberapa tabel yang tiap tabel berisi

informasi yang dapat mengambil informasi dari objek pada sebuah citra. Tujuan dari ekstraksi ciri sendiri adalah mendefinisikan sifat dari setiap objek yang kita tangkap dalam citra. Ekstraksi citra ini sangat beraneka ragam, berbeda objek dalam citra juga beda metode ekstraksi ciri yang kita pakai. Contoh secara umumnya adalah, jumlah lekukan pada sisi luar, jumlah tonjolan, jumlah lubang pada objek dan masih banyak lagi. Setelah ekstraksi ciri diperoleh kemudian akan dibandingkan dengan data latih yang telah diperoleh sebelumnya dicari tingkat kemiripannya dengan ambang batas. Ekstraksi ciri yang memenuhi ambang batas akan dianggap sebagai kecocokan pada data latih dan kemudian akan menjadikan objek tersebut dikenali oleh sistem dan menerjemahkannya.

III. METODE PENELITIAN

Metode yang diimplementasikan dalam penelitian kali ini akan menggunakan metode segmentasi tiap satu huruf yang nantinya akan dibagi menjadi beberapa bagian. Bagian ini nanti akan dideteksi jumlah citranya dan akan ditampung dalam basis data yang nantinya akan diolah dan dicocokkan dengan data uji.



Gambar 1. Siklus Metode Penelitian

A. Citra Plang Jalan

Citra plang jalan di sini diambil dengan kamera posisi pengambilan gambarnya horizontal antara objek dan lensa kamera. Untuk ukuran pada citra RGB ini diperlukan ukuran yang sama yaitu 500x180 piksel.



Gambar 2. Citra Plang Jalan RGB

B. Grayscale

Proses mengubah citra plang jalan yang sebelumnya masih RGB menjadi *grayscale*. *Grayscale* di sini

menggunakan fungsi 'rgb2gray' pada aplikasi matlab. Fungsi 'rgb2gray' membuat citra objek yang tadinya berwarna akan menjadi citra keabuan atau *grayscale*.



Gambar 3. Citra Plang Jalan Grayscale

C. Binerisasi

Binerisasi dilakukan menggunakan fungsi 'im2bw' dengan nilai *threshold* menggunakan fungsi 'graythresh'. Fungsi 'im2bw' adalah membuat objek citra *grayscale* menjadi citra biner, saat sistem mengkonversikan diperlukan nilai ambang batas yaitu dengan fungsi 'graythresh'. Objek yang akan diambil informasinya adalah objek aksara Jawa jadi banyak informasi yang tidak penting untuk dibuang nantinya.



Gambar 4. Citra Plang jalan Biner

D. Filling

Filling dilakukan untuk membuang bagian dari objek yang tidak penting. Metode *filling* di sini akan dilakukan 2 kali yaitu pertama menggunakan fungsi 'bwareopen'. Fungsi 'bwareopen' bertujuan untuk menyeleksi piksel tidak diinginkan dengan ambang batas 800, objek yang nilai pikselnya lebih dari 800 akan dihilangkan dan kedua menggunakan ambang batas 20, sebaliknya objek yang lebih kecil dari ambang batas 20 akan dihilangkan. Dari sini objek ukuran aksara Jawa bisa diketahui mempunyai ukuran piksel antara 20-800 dengan catatan citra RGB harus sesuai kriteria seperti pada penjelasan poin A. Penyeleksian terhadap karakter aksara Jawa terbilang lumayan berhasil dengan menggunakan metode ini walaupun masih ada beberapa syarat yang harus terpenuhi.

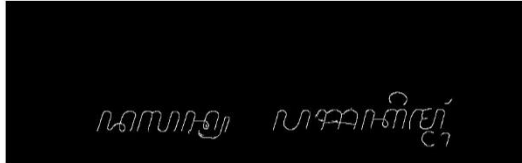


Gambar 5. Citra Plang Jalan Filling

E. Thining

Thining menjadikan objek yang akan diambil informasinya sesimple mungkin atau mengambil bagian yang

penting saja. *Thining* di sini menggunakan fungsi 'bwmorph' yang dipadukan dengan 'thicken'. Fungsi 'bwmorph' bertujuan menerapkan operasi morfologis tertentu ke gambar biner BW dan fungsi 'thicken' ialah $n = \text{Inf}$, mengental objek dengan menambahkan piksel ke bagian luar objek sampai melakukan hal itu akan menghasilkan objek yang sebelumnya tidak terhubung menjadi terhubung. Opsi ini mempertahankan nomor Euler.



Gambar 6. Citra Plang Jalan Thining

F. Segmentasi

Proses segmentasi di sini membagi kalimat aksara Jawa tersebut ke dalam satuan huruf. Proses ini menggunakan fungsi 'bwlabel' dengan ambang batas 8 dan nantinya sistem akan memilah objek dengan ambang batas tersebut untuk dipilih, setelah itu mengambil informasi dengan 'regionprops' yang berisi 'boundingbox' dan 'area' dalam masing-masing objek. Ambang batas 8 diketahui setelah melakukan percobaan berulang kali sehingga memunculkan hasil yang mendekati sempurna. Pada fungsi 'bwlabel' menggunakan jenis :

$$[L,n] = \text{bwlabel}(_)$$

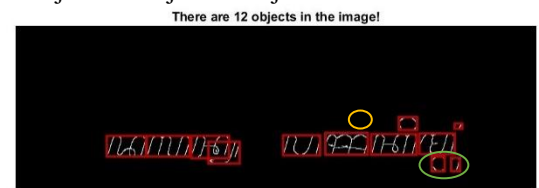
mengembalikan nilai n, ke dalam jumlah objek terhubung yang ditemukan di citra biner. Fungsi 'regionprops' menggunakan jenis :

$$\text{stats} = \text{regionprops}(L, 'BoundingBox', 'Area')$$

Bertujuan untuk mengukur serangkaian properti untuk setiap wilayah berlabel di label L. Fungsi 'BoundingBox' adalah Posisi dan ukuran kotak terkecil yang berisi wilayah, dikembalikan sebagai vektor 1-oleh- (2 * Q). Elemen Q pertama adalah koordinat sudut minimum kotak. Elemen Q kedua adalah ukuran kotak di sepanjang setiap dimensi. Misalnya, kotak pembatas 2-D dengan nilai [5.5 8.5 11 14] menunjukkan bahwa (x, y) koordinat sudut kiri atas kotak adalah (5.5, 8.5), lebar horizontal kotak adalah 11 piksel, dan ketinggian vertikal kotak adalah 14 piksel. Fungsi 'Area' adalah Jumlah piksel aktual di wilayah tersebut, dikembalikan sebagai skalar. (Nilai ini mungkin sedikit berbeda dari nilai yang dikembalikan oleh bwarea, yang menimbang pola piksel yang berbeda secara berbeda). Untuk menemukan yang setara dengan area volume 3-D, gunakan properti 'Volume' dari regionprops3.

Setelah terdeteksi, kita akan melakukan perulangan untuk membatasi masing-masing objek dengan warna merah dan kemudian memotongnya dan menjadikan sebuah citra baru. Pada kasus ini sitem mendeteksi adanya 12 objek berbeda yang nantinya akan kita peroleh hasil ekstraksi dengan cara membagi tiap objek dengan 9 potongan yang sama. Pada kasus ini terdapat satu objek yang tak bisa dikenali yaitu objek yang berada di atas karakter 'ga' yaitu karakter 'layar'. Tanda lingkaran kuning menunjukkan hilangnya objek. Ada

juga yang seharusnya menjadi satu objek namun sistem mengenalinya menjadi 2 objek yaitu karakter pasangan 'ba'. Tanda lingkaran hijau menunjukkan kejadian tersebut.



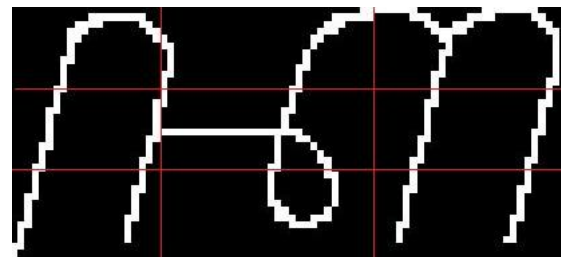
Gambar 7 Citra Plang Jalan Segmentasi

IV. HASIL DAN PEMBAHASAN

Hasil dan pembahasan akan merujuk pada hasil dari ekstraksi ciri dan juga pengenalan pola yang dilakukan oleh sistem. Data dari ekstraksi ciri akan disimpan pada tabel untuk menentukan huruf dalam aksara jawa dan akan dilakukan dengan hasil uji.

A. Ekstraksi Ciri

Proses ekstraksi ciri pada penelitian kali ini yaitu menghitung jumlah piksel pada masing-masing bagian dari objek yang telah dipotong-potong ke dalam satu huruf, dimana satu huruf tersebut akan dibagi menjadi 9 bagian yang sama rata. Pemotongan 9 bagian ini dirasa cukup untuk mewakili setiap piksel yang berada pada objek aksara Jawa tersebut, karena dirasa hasil yang diperoleh sudah lumayan bagus. Terlebih lagi pemotongan melebihi 9 bagian akan terlalu banyak nilai dan menjadikan proses ekstraksi cukup memakan waktu.



Gambar 8. Citra dibagi ke dalam 9 bagian

Ekstraksi ciri nantinya akan disimpan, pada tabel 1 terdapat tabel yang memuat ekstraksi ciri tersebut. Ada 12 kolom dan 9 baris yang berarti ada 12 objek yang masing-masing mempunyai 9 ekstraksi ciri. Dalam penelitian ini masing-masing nilai tersebut mempunyai jumlah piksel pada bagian dari objek. Ekstraksi ciri di sini menjadikan sebuah objek baru yang akan dikenali oleh sistem atau biasanya disebut data uji.

	1	2	3	4	5	6	7	8	9	10	11	12
1	29	33	32	38	29	50	41	19	27	15	9	0
2	22	44	23	17	12	38	26	14	34	8	5	0
3	41	39	40	40	41	48	56	14	37	6	13	4
4	30	30	34	39	29	49	35	11	21	12	0	0
5	39	16	43	13	1	53	37	0	22	0	0	4
6	16	30	26	35	31	21	30	10	30	0	11	2
7	28	27	29	42	20	16	25	6	22	13	0	5
8	38	31	26	25	35	13	22	0	35	9	6	7
9	27	27	17	5	11	30	23	13	17	10	3	0

Tabel 1. Hasil Ekstrasi Ciri

B. Pengenalan Karakter

Pengenalan karakter akan dilakukan pencocokan dari hasil ekstraksi ciri dengan data latih yang telah dibuat sebelumnya. Data latih adalah data yang berisi patokan atau pondasi bagi sistem untuk mengenali data baru, jadi pada kasus ini data latih sebagai landasan utama dan data baru yang diinputkan akan dikenali oleh sistem atau tidak, tergantung dengan kemiripan data latih. Data latih telah disiapkan dan berisi 12 objek yang telah terekstraksi cirinya.

	1	2	3	4	5	6	7	8	9	10	11	12
1	33	35	33	39	34	48	43	19	32	19	11	0
2	29	49	23	18	12	44	30	15	36	11	9	0
3	49	42	47	44	43	51	59	13	40	8	12	5
4	37	32	37	38	32	52	38	10	25	11	5	0
5	54	17	48	15	2	51	39	0	19	0	0	4
6	28	33	27	37	31	20	32	9	30	14	11	1
7	43	28	32	43	23	16	31	6	24	19	3	5
8	32	33	24	23	39	17	21	0	33	17	11	5
9	17	27	17	9	25	37	25	12	17	12	1	0

Tabel 2. Data Latih

Dalam kasus ini telah disamakan posisi urutan huruf dalam data latih sehingga pencocokan di sini hanya terbatas pada huruf yang ada dalam objek yang ingin kita cari informasinya. Ambang batas dalam kasus ini adalah 3 jika dalam tabel hasil ekstraksi ciri selisihnya tidak melebihi 3 maka akan dianggap cocok.

	1	2	3	4	5	6	7	8	9	10	11	12
1	29	33	32	38	29	50	41	19	27	15	9	0
2	22	44	23	17	12	38	26	14	34	8	5	0
3	41	39	40	40	41	48	56	14	37	6	13	4
4	30	30	34	39	29	49	35	11	21	12	0	0
5	39	16	43	13	1	53	37	0	22	0	0	4
6	16	30	26	35	31	21	30	10	30	0	11	2
7	28	27	29	42	20	16	25	6	22	13	0	5
8	38	31	26	25	35	13	22	0	35	9	6	7
9	27	27	17	5	11	30	23	13	17	10	3	0

Tabel 3. Tabel Kecocokan Hasil Ekstraksi Dengan Data Latih

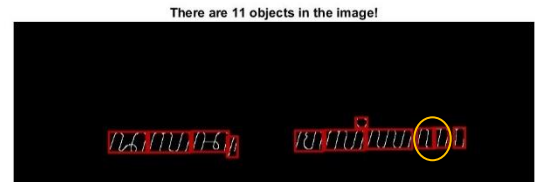
Dalam Tabel 3 terdapat tabel yang ada 77 nilai yang berbeda warnanya yang menandakan kecocokan hasil ekstraksi dengan data latih. Sementara itu ada 31 nilai yang berwarna dasar menandakan tidak adanya kecocokan antara hasil ekstraksi dengan data latih. Rumus mendapatkan nilai akurasi adalah jumlah kecocokan data uji atau dalam kasus ini ekstrasi ciri dengan data latih, lalu dibagi jumlah data dan dikali 100%. Dari 108 nilai ada 77 yang cocok maka dapat disimpulkan dalam kasus perhitungan kali ini sistem bisa mendeteksi kecocokan dengan akurasi 71%, pengenalan pola di sini cukup sederhana jadi dalam kasus ini sangat mudah mendapatkan nilai akurasi tinggi.

Kasus kedua dilakukan dengan mengubah objek yang akan kita kenali dan juga data latih ikut diubah. Di sini sistem berhasil mengenali karakter aksara Jawa yaitu ada 11 objek yang bisa dideteksi untuk mengekstraksinya. Walau dalam kasus ini satu karakter huruf aksara Jawa dideteksi

menjadi 2 objek karena dalam aksara jawa memang ada 2 objek terpisah namun masih dalam satu karakter. Kejadian ini bisa dilihat dalam gambar 10 pada bagian lingkaran kuning.



Gambar 9. Citra RGB kasus 2



Gambar 10. Hasil segmentasi oleh sistem

Dalam tabel di bawah ada 79 nilai yang berbeda warnanya yang menandakan kecocokan hasil ekstraksi dengan data latih. Sementara itu ada 20 nilai yang berwarna dasar menandakan tidak adanya kecocokan antara hasil ekstraksi dengan data latih. Dari 99 nilai ada 79 yang cocok maka dapat disimpulkan dalam kasus perhitungan kali ini sistem bisa mendeteksi kecocokan dengan akurasi 79%, Pengenalan pola di sini cukup sederhana jadi dalam kasus ini sangat mudah mendapatkan nilai akurasi tinggi.

	1	2	3	4	5	6	7	8	9	10	11
1	28	34	30	8	22	43	12	38	19	11	20
2	19	40	17	15	26	34	7	36	21	23	10
3	40	38	42	22	39	39	11	38	6	20	17
4	28	30	31	5	24	30	8	28	14	2	14
5	39	15	37	10	20	15	0	27	14	13	0
6	17	31	22	15	18	31	8	28	0	14	13
7	29	27	25	14	17	23	8	32	12	17	11
8	40	28	14	0	28	25	10	32	17	0	1
9	28	23	20	1	9	21	7	32	11	10	11

Tabel 4. Tabel Ekstraksi Ciri

	1	2	3	4	5	6	7	8	9	10	11
1	31	30	29	33	23	33	11	41	17	14	19
2	21	41	17	17	20	41	8	38	20	23	10
3	38	38	42	37	36	37	11	36	6	18	16
4	31	27	25	19	22	28	8	27	13	2	13
5	38	16	38	23	18	14	0	28	14	11	0
6	19	28	21	29	16	27	8	28	0	13	12
7	26	26	27	28	21	25	8	32	11	17	13
8	32	21	15	21	32	24	7	33	16	0	0
9	25	30	23	9	12	25	8	25	11	10	11

Tabel 5. Tabel Data Latih

	1	2	3	4	5	6	7	8	9	10	11
1	28	34	30	8	22	43	12	38	19	11	20
2	19	40	17	15	26	34	7	36	21	23	10
3	40	38	42	22	39	39	11	38	6	20	17
4	28	30	31	5	24	30	8	28	14	2	14
5	39	15	37	10	20	15	0	27	14	13	0
6	17	31	22	15	18	31	8	28	0	14	13
7	29	27	25	14	17	23	8	32	12	17	11
8	40	28	14	0	28	25	10	32	17	0	1
9	28	23	20	1	9	21	7	32	11	10	11

Tabel 6. Pencocokan Hasil Ekstraksi Dengan Data Latih

Proses pengenalan aksara Jawa dilakukan dengan data latih paling minimal yaitu ada 60 objek yang berisi 20 huruf aksara Jawa, 20 pasangan huruf aksara Jawa dan 20 sandhangan aksara Jawa. Dari 60 objek tersebut akan dibagi lagi menjadi 9 nilai dari masing-masing objek berarti ada 540 nilai yang menjadi data latih. Pada proses penelitian ini tidak memuat data latih sebanyak itu, karena ini kasus simpel jadi saya sederhanakan dalam lingkup data latih yang sesuai dengan objek yang awal kita cari informasinya.

Dalam penelitian ini selain mencapai tujuan pengenalan pola karakter juga menghasilkan banyak sekali capaian dalam mendeteksi karakter aksara Jawa. Segmentasi yang begitu rumit dalam sistem ini telah mencapai akurasi yang hampir sempurna, dapat dilihat sistem bisa langsung menemukan objek-objek aksara Jawa secara keseluruhan dan membuang objek yang tidak ada kaitannya dengan aksara Jawa. Segmentasi pada sistem ini juga bisa langsung memotong antara objek-objek yang telah ditentukan sehingga sistem langsung memotongnya dan menjadikan sebuah citra baru. Dalam ke-3 aspek aksara Jawa yaitu huruf, pasangan dan sandhangan juga telah tercakup pada penelitian kali ini, sehingga kriteria pendeteksian objek aksara Jawa telah berjalan dengan maksimal. Tabel pada gambar 17 di bawah ini menunjukkan akurasi pada masing-masing aspek dan nilai yang berwarna merah merupakan nilai yang terdeteksi cocok oleh sistem.

Huruf	Kasus 1										Kasus 2										Total	Akurasi
	Da	La	Na	Pa	Ga	Ka	Ma	Da	La	Na	Ma	La	Ta	Ba	Ra							
1	29	31	32	29	50	41	27	28	34	30	32	43	38	19	11	20	109/144	75%				
2	22	44	23	19	38	26	34	19	40	17	26	34	36	21	28	10						
3	41	39	40	41	48	56	37	40	38	43	39	39	38	6	20	17						
4	30	30	32	29	49	35	21	28	30	31	34	30	28	14	2	14						
5	39	16	43	1	53	37	22	39	15	37	20	15	27	14	13	0						
6	16	20	26	21	21	30	30	17	11	21	18	31	28	0	14	13						
7	28	27	29	20	16	25	22	25	27	25	17	23	32	12	17	11						
8	38	31	28	35	13	22	35	40	28	14	28	25	32	17	0	1						
9	27	27	12	11	30	19	17	28	23	20	9	21	32	11	10	11						
Jumlah	0	8	7	6	6	7	7	7	6	8	7	5	8	9	9	9						

Tabel 7. Hasil Akurasi Huruf Aksara Jawa

Sandhangan	Kasus 1			Kasus 2		Total	Akurasi
	pangkon	pepet e	cecac (ng)	pangkon	wulu(i)		
1	38	19	0	8	12	35/45	77%
2	17	14	0	15	7		
3	40	14	4	22	11		
4	39	11	0	5	8		
5	13	0	4	10	0		
6	35	10	2	15	8		
7	42	6	5	14	8		
8	25	0	7	0	10		
9	5	13	0	1	7		
Jumlah	7	9	9	1	9		

Tabel 8. Hasil Akurasi Sandhangan

Pasangan	Kasus 1		Kasus 2	Total	Akurasi
	Pasangan Ba		-		
1	15	9	-	11/18	61%
2	8	5	-		
3	6	13	-		
4	12	0	-		
5	0	0	-		
6	0	11	-		
7	13	0	-		
8	9	6	-		
9	10	3	-		
Jumlah	5	6	-		

Tabel 9. Hasil Akurasi Pasangan

V. KESIMPULAN

Dalam penelitian pengenalan citra aksara Jawa ini, penerapan beberapa metode di atas telah menyeleksi citra yang tidak diinginkan dan metode segmentasi yang telah terbukti menghasilkan identifikasi aksara Jawa secara otomatis dengan akurasi yang cukup memuaskan. Pada kasus 1 terdapat 2 kesalahan pada metode segmentasi, dengan tingkat akurasi hasil akhir 71% dan kasus kedua hanya terdapat 1 kesalahan pada metode segmentasi, dengan tingkat akurasi hasil akhir 79%. Tingkat akurasi dalam ke-3 aspek aksara Jawa juga cukup memuaskan. Pada huruf aksara Jawa memiliki tingkat akurasi 75%, sandhangan aksara Jawa memiliki tingkat akurasi 77% dan pasangan aksara Jawa memiliki tingkat akurasi 61%. Dalam kedepannya semoga penelitian ini bisa dikembangkan dengan cara citra yang diidentifikasi tidak memiliki kriteria tertentu dan metode-metode yang memiliki ambang batas tidak lagi ditentukan dengan cara manual.

VI. REFERENSI

- [1] A. R. Widiarti, *Pengenalan citra Dokumen Sastra Jawa Konsep Dan Implementasinya*, pp. 1-2, 2006.
- [2] C. Haruechaiyasak, S. Kongyoung and M. Dailey, "A comparative study on Thai word segmentation approaches," *Computer Science and Information Management Asian Institute of Technology*, 2008.
- [3] S. Norbu, P. Choejey, T. Dendup, S. Hussain and A. Mauz, "Dzongkha Word Segmentation," *8th Workshop on Asian Language Resources*, 2010.
- [4] A. Prabantoror, "Segmentasi Aksara Jawa," 2014.
- [5] L. L. Zhangrila, "Accuracy Level of \$P Algorithm for Javanese Script Detection on," *3rd International Conference on Computer Science and Computational Intelligence 2018*, p. 9, 2018.
- [6] M. R., "Pengolahan citra digital dengan pendekatan algoritmik," 2004.
- [7] D. Tanaya and M. Adriani, "Dictionary-based Word Segmentation for Javanese," *5th Workshop on Spoken Language Technology for Under-resourced Languages, SLTU 2016*, p. 6, 2016.
- [8] R. M. Haralick, "Digital Step Edges from Zero Crossing of Second Directional Derivatives," *Readings in Computer Vision*, pp. 216-226, 1987.
- [9] C. Arcelli and G. S. Di Baja, "A Width-Independent Fast Thinning Algorithm," *IEEE*, 1985.