

## Analisis Unjuk Kerja K-Nearest Neighbour untuk Klasifikasi Citra Aksara Bali Tulis Tangan

Anastasia Rita Widiarti<sup>\*1</sup>, Hari Suparwito<sup>2</sup>

<sup>1,2</sup>Fakultas Sains dan Teknologi, Universitas Sanata Dharma, Yogyakarta, Indonesia

e-mail: <sup>\*1</sup>[rita\\_widiarti@usd.ac.id](mailto:rita_widiarti@usd.ac.id), <sup>2</sup>[shirsj@jesuits.net](mailto:shirsj@jesuits.net)

### Abstrak

Keterbatasan tenaga ahli filolog, dan rentannya material daun lontar yang menjadi aset warisan leluhur jaman dulu, menjadi pemicu untuk dilakukannya otomatisasi alih aksara atau transliterasi citra aksara Bali di daun lontar berbantuan komputer. Algoritma klasifikasi k-nearest neighbour atau kNN, bisa menjadi alat yang dapat digunakan untuk transliterasi tersebut. Prinsip kerja kNN yang sederhana, yaitu dengan mencocokkan kemiripan data baru ke data-data uji terdekat, mampu digunakan untuk tranlisterasi citra aksara Bali.

Pendekatan yang dilakukan pada penelitian ini, selain menitik beratkan pada tahap klasifikasi, juga memperhitungkan dua tahap proses sebelum dilakukan klasifikasi. Perlu proses penyiapan citra yang terdiri dari binerisasi, pemotongan bagian kosong, penyamaan ukuran, dan penipisan, dan proses ekstraksi ciri yang menggunakan algoritma intensity of pixels.

Dengan mempergunakan 18 kelas yang mewakili 18 aksara Bali, dan jumlah data 1001 citra, diperoleh rerata prosentase akurasi 84.746%. Akurasi tersebut diperoleh dengan menerapkan prinsip uji silang 3-fold. Dari penelitian ini pula dapat disimpulkan, meskipun data citra yang digunakan adalah hasil tulisan tangan, dengan mempergunakan data latih yang cukup besar, kNN mampu digunakan untuk klasifikasi. Hal ini menunjukkan bahwa kNN dapat diterapkan sebagai metode klasifikasi citra aksara Bali di daun lontar, sehingga dapat dikembangkan lebih lanjut sebagai mesin utama untuk transliterasi citra daun lontar.

**Kata kunci**— citra aksara, intensity of pixels, klasifikasi, k-nearest neighbour

### Abstract

A lack of philologists and the vulnerability of palm leaf material have become triggers for the scripting automation or transliteration of Balinese script images on computer-assisted palm leaves. One possibility to solve this problem is to create a transliteration machine. We proposed a machine learning technique using the k-NN algorithm to create a transliteration of Balinese script images. The benefit of the kNN algorithm is simply working by matching the similarity of new data to the nearest test data.

Instead of focusing on the classification technique, the study approaches also analyze the two previous processes: the first process is an image preparation process consisting of binarization, cutting the blanks, equalizing size, and thinning. The second is a feature extraction process using the character intensity algorithm. Our experiment employed 18 classes representing 18 Balinese characters.

The optimal accuracy using a 3-fold cross-validation method to 1001 image data yields an average of accuracy is 84.746%. Although the image data used is handwritten, however, kNN algorithm performed classification well using an extensive training dataset. For that reason, the kNN algorithm could be potential for Balinese script images transliteration.

**Keywords**— character image, intensity of character, classification, k-nearest neighbour

## 1. PENDAHULUAN

Keterbatasan tenaga ahli filolog yang mampu membaca dokumen beraksara Bali pada daun lontar, serta tidak familiernya lagi generasi sekarang dengan aksara Bali tersebut, menyebabkan koleksi lontar seperti yang tersimpan di Pustaka Artati Universitas Sanata Dharma hanya teronggok saja setelah dilakukan katalogisasi. Selain itu ketakutan akan rusaknya lontar semakin tidak memungkinkan adanya upaya untuk menyebarluaskan gagasan dari isi lontar, yang mungkin sangat bermanfaat dalam kehidupan sekarang. Jika hal ini dibiarkan, maka ditakutkan bahwa lambat laun lontar menjadi rusak dan kemudian punah.

Perlu adanya upaya untuk ikut melestarikan warisan salah satu budaya tulis yang dimiliki tersebut. Salah satunya adalah dengan mengembangkan sistem pembacaan lontar secara otomatis berbantuan komputer. Perkembangan teknologi pembelajaran mesin telah membuka peluang tersebut, misalnya dengan memanfaatkan algoritma klasifikasi k-nearest neighbour (kNN) untuk mengembangkan sistem transliterasi aksara Bali ke aksara Latin atau Roman.

Terdapat persoalan ketika akan mengembangkan aplikasi berbasis klasifikasi, utamanya untuk klasifikasi aksara kuno, misalnya ketersediaan data, tingginya variabilitas data terutama karena aksara ditulis dengan tangan, dan ketepatan pemilihan model atau algoritma klasifikasi yang akan memberikan unjuk kerja optimal. Persoalan pertama dan kedua merujuk pada data, yang sudah tidak mungkin ditemukan atau ditambah lagi karena para penulisnya sudah meninggal semua, kecuali meminta ahli aksara tersebut untuk menuliskan kembali persis aksaranya. Namun persoalan ini akan menimbulkan kegagalan proses manakala data yang digunakan mempunyai banyak derau, atau kualitas gambarnya kurang bagus.

Di bidang pelestarian budaya, ternyata algoritma klasifikasi kNN telah digunakan untuk pengenalan berbagai citra aksara, termasuk aksara Bali. Sudarma dan Darma (2014) yang mengambil fitur semantik untuk mengenali 54 citra aksara Bali tulisan tangan, mendapatkan akurasi 88.89% [1]. Di tahun 2017, Darma dan Ariasih berhasil meningkatkan akurasi menjadi 97.53% untuk 81 citra, dengan menambahkan ciri arah garis selain ciri semantik [2]. Masih dilakukan oleh Darma untuk melihat unjuk kerja kNN, ternyata dengan menerapkan ciri *zoning*, kNN mampu menghasilkan akurasi yang sama yaitu di 97.5% [3].

Peneliti lain yang menggunakan ciri Local Binary Pattern atau LBP, yaitu Sari, Hidayat, dan Sunarya (2015) memperoleh akurasi pengenalan menggunakan kNN sebesar 74.6% [4]. Akurasi yang diperoleh diklaim lebih tinggi dibandingkan jika digunakan algoritma klasifikasi Support Vector Machine (SVM). Klaim tersebut ternyata berbeda dari penelitian Sthevanie, Aristya, dan Ramadhan yang menggunakan ciri PHOG dalam risetnya untuk pengenalan aksara Bali. Mereka memperoleh akurasi pada data validasi sebesar 68.245% dengan SVM, di mana nilai akurasi tersebut lebih besar nilainya jika dibandingkan menggunakan kNN [5].

Perbedaan klaim terkait algoritma klasifikasi yang terbaik antara kNN dan SVM untuk pengenalan aksara Bali ini memunculkan keraguan untuk menerapkan algoritma kNN dalam studi ini. Namun dari nilai prosentase akurasi pengenalan citra aksara Bali di penelitian-penelitian di atas, ada dugaan bahwa metode kNN dapat digunakan untuk klasifikasi kNN. Dari kajian yang dilakukan oleh Rajput, dan Ummature (2017), lebih menegaskan kembali bahwa kNN dapat digunakan untuk klasifikasi aksara-aksara kuno [6]. Selain penelitian-penelitian di atas, masih ada penelitian lain terkait dengan studi pengenalan citra aksara Bali, namun dengan metode ekstraksi ciri dan klasifikasi berbeda-beda [7].

Paper ini menyajikan hasil penelitian yang memanfaatkan kNN untuk klasifikasi citra aksara Bali, dengan tujuan untuk mengetahui unjuk kerja dari metode kNN tersebut. Kekhususan dari studi ini adalah pada penggunaan dataset yang diperoleh dari hasil digitalisasi lontar yang diberi nama Adiparwa. Selain itu juga pada kombinasi dari pemanfaatan teknologi penyiapan citra dan ekstraksi cirinya sebelum diklasifikasi dengan kNN.

Diharapkan dengan selesainya studi ini, akan diperoleh alat bantu bagi orang awam atau filolog dalam mengefisienkan pekerjaannya misal untuk transliterasi atau alihaksara lontar tersebut dengan hasil dari klasifikasi. Di sisi lain, pemanfaatan atas hasil studi ini diharapkan dapat diperluas untuk pelestarian warisan leluhur dalam bentuk budaya tulis yang tertuang dalam media daun lontar.

## 2. METODE PENELITIAN

### 2.1 Dataset Citra Aksara Bali

Dataset yang menjadi data sampel adalah citra-citra aksara Bali hasil dari proses segmentasi pada citra daun lontar, yang sebelumnya telah didigitalisasi dengan cara difoto mempergunakan kamera, dan telah dilaporkan dalam penelitian mengenai segmentasi citra daun lontar [8]. Untuk bisa digunakan dalam proses klasifikasi, hasil segmentasi tersebut kemudian dikelompok-kelompokkan terlebih dahulu dalam penelitian mengenai klastering citra aksara Bali, sehingga diperoleh kelas-kelas dari citra-citranya [9].

Dari dua penelitian sebelumnya tersebut, selanjutnya dilakukan pengecekan kesamaan data citra dalam setiap kelompok, lalu dilanjutkan dengan pemberian label di bawah supervisi seorang praktisi pembaca lontar beraksara Bali. Hasil dari proses awal pada data input tersebut, telah menghasilkan data sampel sebanyak 1001, yang digunakan sebagai dataset dalam penelitian ini, seperti tersaji pada Tabel 1.

Tabel 1 Data Penelitian

Data ke	Citra	Label	Jumlah Aksara
1		Ha	30
2		Ta	52
3		Ba	45
4		Sa	64
5		Ma	76
6		Ka	36
7		Nga	28
8		Koma	81
9		Ulu	121
10		Tedong	49
11		Na	108
12		Da	20
13		Wa	42
14		Taleng	92
15		Ra	69
16		Ya	35
17		Suku	33
18		ngka	20

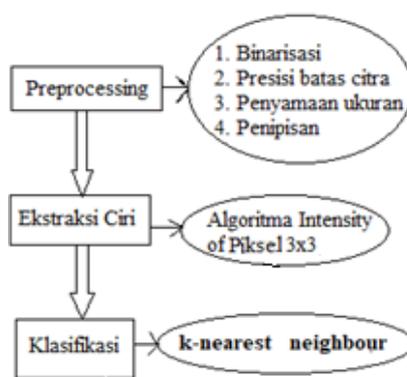
Alasan utama yang menjadi pertimbangan dalam memilih data citra-citra aksara seperti tersaji pada Tabel 1, adalah pada ketersediaan data yang diperoleh dari sumber primer daun

lontar yang dipakai. Hanya 18 aksara Bali yang digunakan, di mana hanya 11 aksara Wianjana dari 18 yang ada di kelompok aksara tersebut, dan sisanya adalah aksara-aksara yang menjadi bagian lainnya.

Dari data-data citra aksara yang ditemukan seperti tersaji di Tabel 1, menurut ahli aksara Bali, banyak penulisan aksara yang dinilai tidak sesuai aturan, dan bentuk aksaranya tidak pas. Kesan ahli tersebut tentu akan menimbulkan persoalan tersendiri untuk proses-proses dalam tahap otomatisasi pembacaan. Diperlukan preprosesing yang tepat, dan ekstraksi ciri yang tepat atau cocok, sehingga akan menghasilkan akurasi maksimal.

## 2.2 Desain Alur Proses Pengolahan Data

Secara garis besar, terdapat 3 proses utama yang dirancang untuk mewujudkan alat bantu pengujian untuk melihat unjuk kerja dari klasifikasi kNN, dengan memperhatikan dataset yang dipakai sebagai sampel, seperti ditunjukkan pada Gambar 1. Keseluruhan dataset pertama kali harus disiapkan di proses *Preprocessing*, setelah itu data masuk ke proses *Ekstraksi Ciri*, dan terakhir masuk proses *Klasifikasi*.



Gambar 1 Desain alur proses

### 2.2.1 Proses Preprocessing

Dari hasil penelusuran pada kualitas citra dataset, maka didapatkan 4 subproses pada tahap *Preprocessing* untuk menyiapkan agar citra siap diproses lebih lanjut, seperti ditunjukkan pada Gambar 1., yaitu subproses *Binarisasi*, *Presisi batas citra*, *Penyamaan ukuran*, dan *Penipisan* citra. Keempat subproses tersebut dilakukan secara terurut.

Subproses *Binarisasi* diimplementasikan menggunakan fungsi *im2bw()* di Matlab, untuk mengubah representasi warna dari non biner ke biner. Dalam subproses *Presisi batas citra*, bagian atau ruang kosong persis di atas, di bawah, di kiri, dan kanan citra dibuang. Proses ini dilakukan secara otomatis dengan menerapkan profil proyeksi pada histogram citra biner. Subproses *Penyamaan ukuran* untuk mengubah ukuran citra dataset, ke satuan ukuran piksel tertentu, dalam hal ini ke ukuran 30x30, 45x45, dan 60x60 piksel. Dasar dari pemilihan ukuran tersebut adalah bahwa ukuran-ukuran yang ditetapkan masih berada dalam range ukuran-ukuran citra data yang digunakan sebagai dataset. Subproses *Penipisan* dengan algoritma *Rosenfeld*, digunakan untuk menyamakan ketebalan citra, karena karakteristik data citra tulisan tangan biasanya tidak konsisten dalam ketebalannya mengingat media yang digunakan untuk menulis.

### 2.2.2 Proses ekstraksi ciri dengan “intensity of pixel”

Proses *Ekstraksi ciri* seperti ditunjukkan pada proses kedua di Gambar 1, diimplementasikan dengan menggunakan konsep *intensity of pixel 3x3* atau disingkat *IoP*, yaitu dengan menghitung jumlahan semua piksel yang berwarna hitam pada suatu luasan yang ditentukan. Luasan yang diambil dalam penelitian ini berukuran 3x3 piksel. Sehingga, jika terdapat suatu luasan citra berukuran 3x3 piksel yang menjadi bagian dari citra biner  $M$ , maka rumus yang digunakan untuk mendapatkan nilai intensitas di luasan tersebut ditunjukkan pada

persamaan (1), jika 1 mewakili intensitas warna hitam.

$$i = \sum_{j=1}^3 \sum_{k=1}^3 M_{j,k} \quad (1)$$

Gambar 2 memperlihatkan contoh representasi data citra biner  $M$  berukuran 9x9 piksel, yang telah dibagi menjadi 3x3 bagian atau luasan. Maka vektor ciri dari citra biner  $M$  jika setiap luasan dikenakan persamaan (1) adalah [2, 3, 4, 4, 3, 5, 3, 4, 1], di mana data ke 1 mewakili ciri di luasan di kiri atas, dan data ke 9 mewakili ciri di luasan kanan bawah.

1	1	0	0	1	1	0	1	0
1	0	1	1	0	0	1	1	1
1	1	1	1	1	1	1	0	0
1	0	1	1	0	1	0	1	1
0	0	0	0	1	1	0	0	0
1	1	1	0	1	1	0	1	1
1	1	1	0	1	0	1	1	1
0	0	1	1	0	1	1	0	1
0	1	1	0	1	1	1	1	1

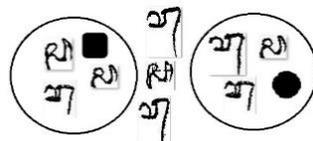
Gambar 2 Representasi citra biner  $M$

### 2.2.3 Proses klasifikasi dengan kNN

Pada penerapannya untuk pembelajaran mesin yang terawasi, algoritma kNN menggunakan pendekatan statistik non-parametrik terutama untuk proses klasifikasi dan regresi. Algoritma ini juga populer digunakan karena kesederhaan logika alur prosesnya namun efisien, dan tidak membutuhkan pembelajaran [10].

Proses klasifikasi dengan algoritma kNN, prinsip kerjanya sangat sederhana sekali. Untuk memasukkan sebuah data baru ke dalam suatu label kelas sesuai yang ada di data latih, hanya perlu diukur jarak kedekatan data baru tersebut dengan data-data yang ada di data latih. Jika mengambil nilai  $k=1$ , maka data baru akan diasumsikan sama dengan label dari 1 data di data latih yang paling dekat jaraknya.

Untuk mengantisipasi kejadian seri, maka nilai  $k$  biasa diambil nilai ganjil. Jika nilai  $k$  lebih dari 1 misalnya  $k=3$  ditulis 3NN, seperti tersaji di Gambar 3, maka label kelas data yang baru akan diambil dari kelas label mayoritas yang berjarak 3 paling dekat. Sebagai contoh, untuk data baru yang diwakili dengan tanda ■ di lingkaran kiri, maka data tersebut akan diberi label di kelas yang dimiliki citra ୧. Contoh lain citra ● di lingkaran kanan akan diberi label yang dimiliki citra ୩. Proses klasifikasi yang akan diimplementasikan dalam alat uji menggunakan fungsi *fitcknn()* di Matlab 2014 dengan paramater jarak euclidean.



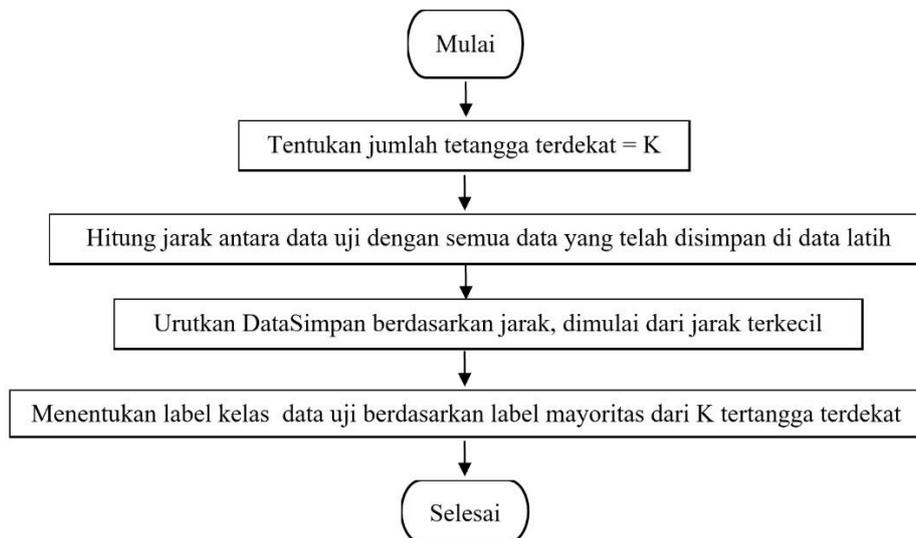
Gambar 3 Ilustrasi klasifikasi dengan 2 label kelas menggunakan 3NN

Alur pemrosesan data menggunakan kNN dapat dilihat pada Gambar 4. Pertama kali ditentukan nilai  $k$  tetangga terdekat yang akan menjadi panduan klasifikasi. Data yang tersimpan di dalam data uji, yaitu data yang belum ditentukan kelasnya kemudian diproses untuk dihitung jaraknya dari data uji ke data yang dikelompokkan sebagai data latih menggunakan persamaan (2). Jika sebuah data di data uji mempunyai ciri sebanyak 9 elemen dari  $c1$  sampai dengan  $c9$ ,

maka jarak euclidean  $d$  dari data uji tersebut ke data latih  $tI$  yang juga mempunyai 9 elemen ciri ditunjukkan pada persamaan (2).

$$d = \sqrt{(c_1 - t_{11})^2 + (c_2 - t_{12})^2 + (c_3 - t_{13})^2 + \dots + (c_9 - t_{19})^2} \quad (2)$$

Jika terdapat  $n$  data latih, maka akan diperoleh data jarak sebanyak  $n$  pula. Data tersebut kemudian diurutkan dari jarak terkecil untuk memudahkan pemilihan  $k$  data terkecil yang akan dilihat mayoritas label di  $k$  data yang terpilih tersebut. Label yang terpilih akan menjadi luaran dari proses klasifikasi.



Gambar 4 Representasi citra biner  $M$

### 2.3 Skenario Pengujian

Pengujian dilakukan untuk mengetahui unjuk kerja kNN dengan mengubah variabel jumlah  $k$  data latih terdekat dengan data uji, yaitu dengan nilai  $k = [1, 3, 5, 7, 9, 11, 13, 15, 19]$ . Nilai-nilai  $k$  yang diambil selalu ganjil, agar tidak terjadi adanya kesamaan jarak yang dapat muncul pada proses kNN saat dijalankan. Selain itu, nilai  $k$  yang dipilih telah mempertimbangkan banyaknya data yang ada dan ukuran dimensi yang dibentuk oleh data.

Pengujian juga akan melihat dampak dari perbedaan pada saat digunakan jumlah data yang jumlahnya sama di setiap kelas alias seimbang, dan jumlah data yang tidak seimbang. Keseimbangan data yang dimaksudkan adalah bahwa jumlah citra aksara perkelas dibuat sama, yaitu disamakan dengan jumlah citra aksara terkecil yang ditemukan dari hasil segmentasi citra daun lontar, yakni 20 citra per kelas.

Skenario pengujian lain juga dibuat untuk melihat dampak dari preprocessing data, yaitu terkait dengan proses perubahan ukuran citra saat dilakukan proses ekstraksi ciri. Dalam hal ini digunakan variasi ukuran 30x30 piksel, 45x45 piksel, dan 60x60 piksel. Ukuran pembagian bidang citra tetap yaitu 3x3 bagian.

Untuk mendapatkan kinerja dari kNN, digunakan uji silang 3-fold. Dua pertiga data akan digunakan sebagai data latih, dan sisanya untuk data uji. Pada setiap fold, setiap data pada satu label kelas ada sampelnya, dengan menggunakan konsep urutan data yang muncul.

## 3. HASIL DAN PEMBAHASAN

Pada bagian ini, kinerja kNN untuk transliterasi citra aksara Bali tulisan tangan dianalisis. Eksperimen dilakukan untuk dataset yang berisi 18 kelas, di mana masing-masing kelas terdiri dari beragam jumlah sampel seperti tersaji pada Tabel 1. Untuk analisis pada jumlah data yang seimbang, digunakan 20 sampel citra per kelas. Total gambar terdapat 1001

yang digunakan untuk tujuan analisis. Dataset citra aksara Bali dibagi secara acak, yaitu 2/3 bagian untuk data latih, dan sisanya untuk data uji. Desain yang dibuat, diimplementasikan dalam platform Matlab, dan kinerja dianalisis dengan pemanfaatan pengukuran statistik akurasi.

Data input berupa dataset citra aksara Bali tulisan tangan, pertama-tama dibinerkan, dipotong bagian atas-bawah-kiri-kanan yang kosong, disamakan ukurannya sesuai skenario ukuran piksel yang ditentukan, kemudian ditipiskan. Citra baru yang dihasilkan kemudian diekstraksi fitur menggunakan ciri *IoP* dengan ukuran pembagian jendela 3x3. Hasil ekstraksinya tersebut kemudian diproses dengan algoritma klasifikasi *kNN* yang bekerja dengan dua mode proses di data latih dan data uji. Untuk mengukur kedekatan data uji dengan data latih pada saat klasifikasi, di penelitian ini digunakan rumus jarak Euclidean. Di akhir proses, hasil transliterasinya diukur dengan metode matrik konfusi dan secara keseluruhan dianalisis akurasinya dengan validasi uji silang 3-fold.

Hasil-hasil skenario pertama dengan mempergunakan nilai  $k=1$  selanjutnya disebut 1NN, pada ukuran citra 30x30 piksel, 45x45 piksel, dan 60x60 piksel dengan jumlah data di setiap kelasnya tidak seimbang tersaji pada Tabel 2. Total jumlah data yang digunakan baik untuk data latih maupun data uji sebanyak 1001 data. Pada setiap fold diukur akurasinya untuk melihat set data latih manakah yang memberikan akurasi paling baik, dan melihat ukuran piksel manakah yang paling sesuai.

Tabel 2 Rerata prosentase akurasi data latih dengan nilai  $k=1$  pada data tidak seimbang

Fold	Rerata Prosentase Akurasi pada Ukuran Citra		
	30x30 piksel	45x45 piksel	60x60 piksel
1	84.6847	81.0811	82.5826
2	84.985	83.4835	82.5826
3	85.0746	84.4776	85.6716

Dari data yang tersaji pada Tabel 2, menunjukkan bahwa semua rata-rata akurasi di setiap foldnya di atas 81%. Sebuah nilai akurasi yang bisa diasumsikan bahwa klasifikasi 1NN dan mempergunakan algoritma ekstraksi ciri yang diterapkan, telah berhasil untuk digunakan sebagai transliterasi citra aksara Bali tulisan tangan. Dataset latih di fold 3 memberikan kinerja terbaik, yaitu dapat diketahui dari tingginya nilai rerata akurasi di semua skenario ukuran citra berbeda-beda.

Dari tabel tersebut juga dapat diketahui bahwa skenario terbaik saat 1NN dapat dicapai pada ukuran citra 60x60 piksel. Namun, jika dilihat dari rerata akurasi keseluruhan pada semua fold diperoleh nilai akurasi berturut-turut 84.915% untuk ukuran 30x30 piksel, 83.014% untuk ukuran 45x45 piksel, dan 83.612% untuk 60x60 piksel. Kesimpulan lain dapat diperoleh yaitu bahwa ukuran citra 30x30 piksel memberikan kinerja terbaik. Oleh karena itu, pada skenario pengujian selanjutnya dengan nilai  $k$  berbeda, akan tetap dilihat kinerja algoritma jika digunakan 3 ukuran citra yang telah ditetapkan dalam penelitian ini.

Pengujian seperti pada skenario pertama yang menghasilkan data rerata akurasi pada Tabel 2, kemudian dilakukan untuk skenario  $k=[3, 5, 7, 9, 11, 13, 15, 17, 19]$  untuk semua ukuran citra. Dari keseluruhan skenario pertama, diperoleh rerata prosentase pada setiap  $k$  mulai dari  $k=1$  sampai dengan  $k=19$ , seperti tersaji pada Tabel 3.

Rerata prosentase yang tersaji pada setiap nilai  $k$  di Tabel 3 adalah nilai rata-rata akurasi pada semua fold dari uji silang 3-fold. Dari data yang tersaji pada Tabel 3, dapat diasumsikan bahwa jika melihat 11 jarak terdekat data uji ke data latih, akan diperoleh hasil rerata akurasi yang maksimal, yaitu sebesar 85.215%. Ada kenaikan akurasi dari  $k=1$  menuju ke  $k=11$ , dan selanjutnya menunjukkan adanya tren penurunan akurasi setelah  $k=11$ . Dari Tabel 3 tersebut, dapat disimpulkan bahwa ada pengaruh atas penggunaan jumlah tetangga terdekat pada saat akan mengklasifikasikan data uji ke dalam sebuah label kelas tertentu.

Tabel 3 Rerata prosentase akurasi data latih dengan semua nilai  $k$  pada data tidak seimbang

Nilai $k$	Rerata Prosentase Akurasi pada Ukuran Citra			Rerata prosentase Akurasi
	30x30 piksel	45x45 piksel	60x60 piksel	
1	84.9148	83.0141	83.6123	83.847
3	85.1132	84.5162	83.9191	84.516
5	85.0161	86.4133	84.2177	85.216
7	84.4179	85.6125	84.2135	84.748
9	84.6169	86.3144	84.5132	85.148
11	84.4167	86.8143	84.4149	85.215
13	84.1181	86.4157	84.5156	85.017
15	84.2182	85.5172	84.515	84.750
17	84.0175	85.9176	84.117	84.684
19	84.3183	84.8188	83.819	84.319

Dari Tabel 3, dapat dihitung nilai rerata akurasi untuk keseluruhan skenario pengujian yaitu sebesar 84.746%, berdasarkan pada data di kolom ke-4 yaitu di kolom *Rerata prosentase Akurasi*.

Jika dilihat dari tren rerata akurasi berdasarkan pada ukuran citra aksara yang digunakan, dari Tabel 3 diperoleh informasi bahwa rerata akurasi pada ukuran 30x30 piksel adalah sebesar 85.1132%, pada ukuran 45x45 sebesar 86.8143%, dan di ukuran 60x60 piksel sebesar 84.5156%. Hasil dari rerata akurasi berdasar ukuran piksel ini menunjukkan bahwa ukuran yang paling sesuai dari citra aksara Bali adalah di 45x45 piksel.

Skenario pertama di atas, kemudian diterapkan untuk melihat apakah ada perbedaan yang mencolok jika digunakan jumlah data yang seimbang untuk setiap kelas label, yang dalam penelitian ini dibuat setiap kelas berjumlah 20 data. Jumlahan tersebut ditetapkan, karena jumlah minimal data di dalam kelas label di penelitian ini adalah 20. Dari pengujian dengan mengubah nilai  $k$ , dan mengubah ukuran citra aksara, diperoleh rerata prosentase akurasi pada semua fold akurasi di semua nilai  $k$ , seperti tersaji pada Tabel 4.

Tabel 4 Rerata prosentase akurasi pada data latih untuk semua  $k$  dengan data seimbang

Nilai $k$ pada $kNN$	Rerata Prosentase Akurasi pada Ukuran Citra			Rerata prosentase Akurasi
	30x30 piksel	45x45 piksel	60x60 piksel	
1	82.8924	81.746	81.3933	82.0106
3	80.9524	81.4815	81.8342	81.4227
5	79.7619	80.9965	82.5397	81.0994
7	80.4233	81.3051	82.8483	81.5256
9	82.1869	79.1446	81.0847	80.8054
11	80.6437	80.8201	82.5397	81.3345
13	80.8642	80.5115	82.1869	81.1875
15	81.4374	78.6596	80.776	80.2910
17	79.7178	78.351	80.291	79.4533
19	79.4533	78.0864	80.5996	79.3798

Terdapat perbedaan pada nilai rerata prosentase akurasi jika mempergunakan data di setiap kelas yang seimbang, yaitu mengalami penurunan akurasi saat digunakan jumlah data yang seimbang di setiap kelas label jika dibandingkan dengan hasil untuk jumlah data yang tidak seimbang. Asumsi yang dapat diambil dari pengujian dengan jumlah data yang tidak seimbang tidak lagi berlaku, yaitu semula  $k$  terbaik di nilai 11, menjadi ke 1 tetangga saja untuk rerata prosentase terbaik. Demikian pula untuk ukuran citra sebelum diekstraksi ciri, semula berukuran 45x45 piksel menjadi ukuran 60x60 piksel yang terbaik. Dari sini dapat disimpulkan bahwa keseimbangan jumlah data di tiap kelas label mempunyai pengaruh pada nilai akurasi yang dihasilkan. Adanya penurunan nilai akurasi bisa jadi disebabkan karena jumlah data yang digunakan di tiap kelas label terlalu sedikit, yakni 20. Bandingkan, jika digunakan data yang

tidak seimbang, hanya 2 kelas label saja yang jumlah data di kelasnya sebanyak 20, lainnya di atas 20. Dapat disimpulkan bahwa semakin banyak data latih yang digunakan tentu akan mempengaruhi kinerja dari algoritma kNN ini.

Masih pada pengujian dengan jumlah data seimbang, jika menilik hasil evaluasi kinerja kNN dengan matrik *confusi* untuk sebuah fold di skenario ukuran citra 60x60 piksel, kesalahan klasifikasi yang besar terjadi pada saat memprediksi aksara *Ma*. Sensitivitas aksara tersebut hanya 0.2, jika menilik data pada matrik konfusinya di Tabel 5.

Tabel 5 Matriks konfusi pengujian pada skenario data seimbang ukuran 60x60 piksel

Citra	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	4	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
2	4	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	1	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	4	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	5	0	0	0	0	0	0	0	2	0	0	0
8	0	0	0	0	0	0	0	5	0	2	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	4	0	0	0	0	0	3	0	0	0
10	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	5	2	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	5

Dari evaluasi pada keseluruhan fold di skenario di atas, kesalahan prediksi yang besar yaitu karena rerata sensitivitasnya kurang dari 0,5 terjadi untuk aksara *ha*, *ta*, dan *ma*. Sedangkan untuk aksara yang sensitivitasnya tinggi terjadi pada aksara selain *ba*, *tedong*, *na*, *da*, *wa*, *ra*, dan *suku* yaitu di kisaran 0.9. Kesalahan prediksi terjadi karena terdapat kemiripan bentuk aksara yang relatif besar di antara aksara-aksara di atas, sehingga diperlukan algoritma ekstraksi ciri yang lebih dari sekedar menghitung luasan piksel hitam pada citranya.

#### 4. KESIMPULAN

Dari penelitian yang sudah dilakukan, dapat disimpulkan bahwa algoritma klasifikasi kNN dapat diterapkan untuk transliterasi atau alih aksara citra aksara Bali tulisan tangan yang diperoleh dari citra lontar. Terdapat rangkaian proses yang utama untuk menghasilkan akurasi terbaik di data latih yaitu sebesar 86.8143% sebelum dilakukan klasifikasi, yaitu proses preprossesing citra aksara Bali untuk membenerkan citra, memotong bagian kosong dari data citra input, menyamakan ukuran, dan menipiskannya. Algoritma *IoP* yang prinsip kerjanya menghitung luasan piksel hitam pada suatu area tertentu kemudian diterapkan pada citra yang telah mengalami preprossesing.

Dari penelitian ini pula, dapat disimpulkan bahwa untuk mendapatkan nilai akurasi yang maksimal, diperlukan data latih yang banyak, sehingga akan memberikan kesempatan mendapat hasil prediksi label kelas terdekat yang tepat. Penggunaan nilai *k* berbeda, mempengaruhi nilai akurasi, di mana dalam penelitian ini tidak dapat diduga nilai *k* terbaik untuk mendapatkan akurasi yang maksimal.

Tingginya variabilitas data yang digunakan, dalam hal ini adalah citra tulisan tangan, memerlukan lebih banyak data latih di setiap kelas label, sehingga variasi tulisan tangan menjadi terwakili di kelasnya.

## 5. SARAN

Klasifikasi data citra dengan mempergunakan *kNN* akan sangat dipengaruhi oleh ketepatan proses pra-prosesing, serta algoritma ekstraksi ciri yang digunakan. Semakin bervariasi data input memerlukan proses yang lebih tepat lagi, maka masih terbuka banyak kesempatan penelitian untuk menerapkan algoritma klasifikasi *kNN* pada penelitian ini dengan mempergunakan metode pra-prosesing dan ekstraksi ciri lainnya. Penelitian ini juga masih harus dikembangkan lagi dengan menambahkan data-data untuk aksara Bali lainnya yang belum digunakan sebelum dapat dibuat sebuah sistem transliterasi yang utuh untuk transliterasi citra daun lontar beraksara Bali.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Universitas Sanata Dharma yang telah memberi “dukungan financial” terhadap penelitian ini.

## DAFTAR PUSTAKA

- [1] M. Sudarma and I. W. A. S. Darma, "The Identification of Balinese Scripts' Characters based on Semantic Feature and K Nearest Neighbor," *International Journal of Computer Applications*, pp. 14-18, April 2014.
- [2] I. W. A. S. Darma and N. K. Ariasih, "Handwritten Balinese Character Recognition using K-Nearest Neighbor," in *The 2nd ICCT International Conference on Culture Technology*, Japan, 2017.
- [3] I. W. A. S. Darma, "Implementation of Zoning and K-Nearest Neighbors in Character Recognition of Wrésastra Script," *Lontar Komputer*, vol. 10, no. 1, pp. 9-18, April 2019.
- [4] I. A. D. P. Sari, B. Hidayat and U. Sunarya, "Pengenalan Aksara Bali Dengan Metode Local Binary Pattern," in *Engineering*, 2015.
- [5] F. Sthevanie, I. P. I. Aristya and K. N. Ramadhani, "Pengenalan Aksara Bali Menggunakan Metode Pyramid Histogram of Oriented Gradients," *Ind. Journal on Computing*, vol. 5, no. 1, pp. 73-84, 2020.
- [6] G. G. Rajput and S. B. Ummapure, "Script identification from handwritten documents using SIFT method," in *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, Chennai, India, 2017.
- [7] S. B. Ahmed, S. Naz, M. I. Razzak and T. M. Breuel, "Balinese Character Recognition Using Bidirectional LSTM Classifier," in *International Conference on Machine learning and Signal Processing (MALSIP)*, Ho Chi Minh, 2015.
- [8] A. R. Widiarti and K. Pinaryanto, "Segmentasi Citra Huruf Daun Lontar. Laporan Penelitian.," Lembaga Penelitian dan Pengabdian pada Masyarakat, Yogyakarta, 2019.
- [9] A. R. Widiarti and C. K. Adi, "Clustering Citra Aksara Bali Hasil Segmentasi Citra Daun Lontar. Laporan Penelitian Hibah Penelitian Internal Reguler.," Lembaga Penelitian dan Pengabdian pada Masyarakat, Yogyakarta, 2019.
- [10] X. Wu, S. Wang and L. Zhang, "The theory and application of k-nearest neighbor algorithm," *Computer engineering and Application*, vol. 53, no. 21, 2017.