

Implementasi *Projection Profile* dan *K-Nearest Neighbors* dalam Pengenalan Tulisan Karakter Aksara Lampung

¹Akmal Junaidi, ²Rehta Yunani, dan ³Ardiansyah

^{1,2,3}Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung
Jalan Soemantri Brojonegoro No.1 Gedong Meneng, Bandar Lampung, Provinsi Lampung, Indonesia
e-mail: ¹akmal.junaidi@fmipa.unila.ac.id, ²rehta.yunani1106@students.unila.ac.id, ³ardiansyah@fmipa.unila.ac.id

Abstract — As one of the countries with a rich culture, Indonesia faces considerable challenges to preserve the diversity of its languages and writings. Lampung Province is one of the regions that has its language and script. However, over time, the variety of writings in the form of the Lampung script began to be forgotten because this variety of writings was not often used as a means of daily communication. One effort can be done to digitize the Lampung script, using the character recognition method using the projection profile as the feature extraction method, and K-Nearest Neighbors (KNN) algorithm to predict character shape. This study has two types of data: training data and testing data with 18-character labels. The comparison of the training and testing data used is 75.56% for training data, and 24.43% for testing data. The features used are features obtained from image extraction measuring 20 x 20 pixels using a projection profile. The types of features obtained are Horizontal Projection Profile (HPP), Vertical Projection Profile (VPP), and Combined Projection Profile (CPP) which are obtained by combining HPP and VPP features. KNN classification is carried out with 10-fold cross-validation, using a base model and an optimized model with GridSearchCV. The best classification results are obtained using the Combined Projection Profile (CPP) feature with an optimized model using GridSearchCV. The evaluation results obtained for the best image data classification with an accuracy of 86.23%, precision of 86.23%, the sensitivity of 86.23%, specificity of 99.19%, F-Measure of 86.23%, and Matthew Correlation Coefficient (MCC) of 85.09%.

Keywords: Lampung Character Recognition; Projection Profile; K-Nearest Neighbors; GridSearchCV.

1. PENDAHULUAN

Sebagai salah satu negara yang memiliki kekayaan budaya dalam jumlah yang masif, Indonesia memiliki tantangan untuk dapat melestarikan budaya yang dimiliki dari generasi ke generasi. Salah satu hasil budaya yang perlu dilestarikan adalah ragam bahasa dan tulisan daerah. Provinsi Lampung, merupakan salah satu provinsi di Indonesia yang memiliki ragam bahasa dan tulisan aksara tersendiri, yakni bahasa dan aksara Lampung. Namun, seiring perkembangan dan percampuran budaya, terkhusus aksara Lampung lambat laun mulai ditinggalkan. Hal ini disebabkan karena aksara Lampung sangat jarang digunakan dalam kehidupan sehari-hari. Berdasarkan pengelompokan jenis aksara, aksara Lampung merupakan turunan dari aksara Devanagari, yang berasal dari India Selatan [1]. Aksara Lampung terbagi menjadi tiga komponen penyusun aksara. Pertama, induk huruf yang merupakan karakter utama pembentuk kata-kata dalam bahasa Lampung yang berjumlah 20 karakter. Kedua, anak huruf yang merupakan peubah bunyi induk huruf, yang membantu pembentukan kata-kata dalam bahasa Lampung. Ketiga, tanda baca yang terdiri dari 5 karakter yang berfungsi untuk memberikan makna tambahan yang terbentuk dari kombinasi karakter induk huruf dan anak huruf.

Melihat permasalahan tersebut, salah satu solusi yang dapat diterapkan adalah dengan melakukan digitalisasi untuk aksara Lampung menggunakan pengenalan karakter atau pengenalan pola. Proses pengenalan pola melibatkan komputer untuk dapat melakukan pengelompokan citra berupa simbol secara otomatis [2]. Lebih lanjut, jenis pengenalan pola yang digunakan adalah pengenalan pengenalan karakter citra tulisan tangan yang lebih dikenal dengan *character recognition* [3]. Beberapa contoh pemanfaatan *character recognition* antara lain untuk pengenalan tulisan tangan [2] dan pengenalan citra huruf alfabet [4]. Secara khusus, usaha untuk melakukan digitalisasi aksara Lampung dengan melalui pengenalan karakter telah dilakukan dengan menggunakan metode *water reservoir*, *branch points*, *pixel density*, dan *end points* [1][5]. Penelitian yang

dilakukan secara spesifik untuk pengenalan karakter tulisan tangan aksara Lampung tersebut menggunakan 18 label untuk karakter citra induk huruf aksara Lampung.

Sebagai upaya untuk memperluas penelitian untuk pengenalan karakter aksara Lampung pada penelitian yang telah dilakukan sebelumnya, penelitian ini akan menggunakan secara spesifik metode ekstraksi fitur *projection profile*, dan metode klasifikasi *K-Nearest Neighbors*. *Projection profile* merupakan proses menambahkan sejumlah piksel hitam pada baris dan kolom [6]. Segmentasi baris dalam sebuah dokumen cetak sering dilakukan dengan metode *projection profile*, namun metode ini dapat diadaptasi untuk dokumen tulisan tangan [7]. Berdasarkan jenis proyeksi yang dilakukan, *projection profile* dapat menghasilkan dua jenis fitur yakni *Horizontal Projection Profile* (HPP) dan *Vertical Projection Profile* (VPP) [6]. *Horizontal Projection Profile* (HPP) adalah proyeksi yang dilakukan pada garis horizontal, atau merupakan baris piksel pada sebuah citra. Kemudian, *Vertical Projection Profile* (VPP) adalah proyeksi yang dilakukan sepanjang garis vertikal, atau merupakan kolom piksel pada sebuah data citra. Berdasarkan kedua jenis fitur yang dapat dihasilkan, kemudian dapat dibentuk sebuah fitur baru yang menggabungkan HPP dan VPP, fitur gabungan ini disebut dengan *Combined Projection Profile* (CPP).

Berbeda dengan penelitian yang telah dilakukan sebelumnya mengenai pengenalan karakter tulisan aksara Lampung [1][5], penelitian ini menggunakan *K-Nearest Neighbors* (KNN) untuk mendapatkan klasifikasi mengenai data citra karakter aksara Lampung. Klasifikasi ditujukan untuk menemukan model yang dapat mencitrakan perbedaan antara kelas atau himpunan, sehingga model yang dihasilkan dapat digunakan untuk melakukan prediksi kelas dari objek label yang tidak diketahui kelas nya [8]. *K-Nearest Neighbors* bekerja dengan cara menemukan sejumlah kelompok “k” objek dalam *data training*, yang memiliki jarak nilai paling dekat dengan objek yang ada pada *data testing* [9]. Hal yang dapat mempengaruhi performa klasifikasi menggunakan KNN adalah jumlah “k” yang ditetapkan [10]. Melihat pengaruh nilai “k” yang sangat signifikan dalam performa klasifikasi yang dihasilkan, penelitian ini akan melakukan klasifikasi berdasarkan dua jenis model KNN, yakni *base model*, dan model yang dioptimasi dengan *GridSearchCV*. Berdasarkan permasalahan yang telah dijabarkan dan penelitian yang telah dilakukan sebelumnya, penelitian ini bertujuan untuk melihat performa pengenalan karakter tulisan aksara Lampung yang dilakukan dengan menggunakan *K-Nearest Neighbors* (KNN). Fitur yang digunakan untuk melakukan klasifikasi adalah fitur *Horizontal Projection Profile* (HPP), *Vertical Projection Profile* (VPP), dan fitur *Combined Projection Profile* (CPP).

2. METODOLOGI PENELITIAN

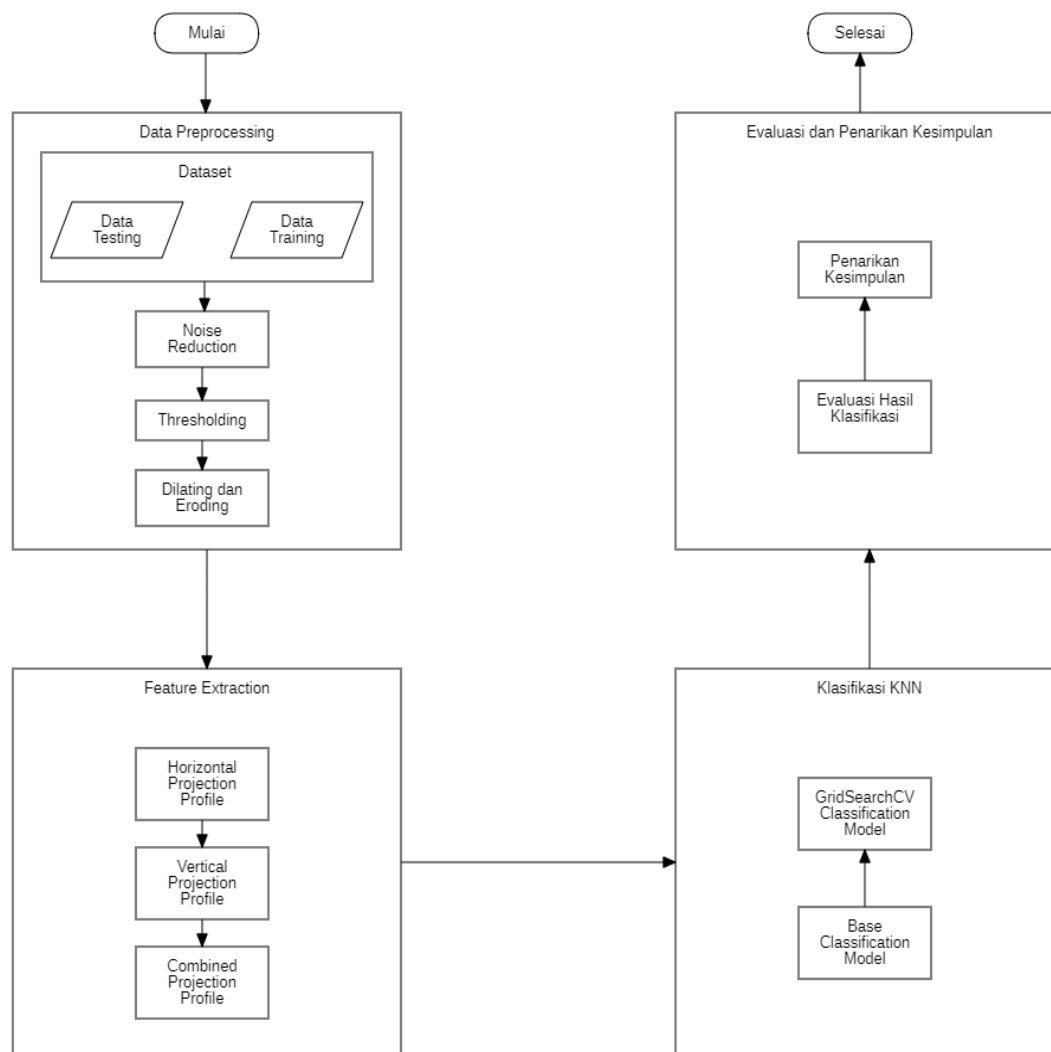
Alur penelitian yang dikerjakan dengan menggunakan metode ekstraksi fitur *projection profile*, dan algoritma klasifikasi *K-Nearest Neighbors* (KNN), dapat dilihat pada Gambar 1.

2.1. Data Preprocessing

Data yang digunakan berupa citra tulisan karakter aksara Lampung yang memiliki jumlah total 32140 data citra. Data yang digunakan kemudian dibagi menjadi *data training* sejumlah 24287 data citra dan *data testing* sejumlah 7853 data citra. Proporsi pembagian data adalah sebesar 75,56% untuk *data training* dan 24,34% untuk *data testing*. Data citra tulisan karakter aksara Lampung yang digunakan memiliki 18 label untuk induk huruf aksara Lampung dan memiliki dimensi 20 x 20 piksel [5]. Data citra yang akan digunakan terlebih dahulu melalui beberapa langkah *preprocessing* sebelum dilanjutkan dalam tahap *feature extraction*. Tahap data *preprocessing* terdiri dari *noise reduction*, *thresholding*, dan gabungan proses *dilating* dengan *eroding*.

Kualitas data citra sangat dipengaruhi oleh jumlah *noise* yang terdapat dalam sebuah citra. Kemunculan *noise* dapat disebabkan oleh beberapa faktor seperti suhu sensor kamera yang terlalu tinggi pada saat pengambilan citra, kurang baiknya kondisi pencahayaan, dan beberapa faktor lingkungan yang dapat menyebabkan kemunculan *noise* [11]. Penelitian ini menggunakan data citra yang diambil menggunakan *scanner*, dan jenis *noise* yang sering ditemukan pada data citra ini adalah *gaussian noise* [11]. Dengan demikian, *noise reduction* pada data citra akan difokuskan untuk menghilangkan *gaussian noise* yang mungkin muncul pada data citra.

Data citra yang telah melalui tahap *noise reduction* kemudian dilanjutkan ke dalam proses *thresholding*. Proses ini akan memberikan batas yang tepat antara pola di dalam citra yang ingin dikenali dengan latar belakang dari sebuah pola citra [12]. Proses *thresholding* memberikan setiap data citra nilai *threshold* yang berbeda, sehingga membuat citra yang akan dikenali memiliki nilai pembeda yang signifikan dengan latar belakang nya [12].



Gambar 1. Alur penelitian pengenalan karakter tulisan aksara Lampung menggunakan *projection profile* dan KNN.

Proses *noise reduction* dan *thresholding* yang telah dilakukan sebelumnya dapat mengakibatkan pola yang terdapat dalam data citra berubah atau hilang, sehingga, proses *dilating* dan *eroding* perlu dilakukan untuk menyempurnakan data citra yang telah melalui beberapa proses dalam tahap *data preprocessing*. *Dilating* dan *eroding* merupakan salah satu jenis *morphological operations*, yang merupakan serangkaian operasi yang dilakukan pada citra dengan cara mengubah bentuk yang mendasari citra biner [13]. *Morphological operations* seperti *dilating* dan *eroding* memerlukan komponen berupa *binary structuring element* [14], yang merupakan pemberi standar bentuk terhadap setiap piksel pada citra biner [14].

Dilating merupakan proses operasi morfologi pada citra biner, yang dilakukan dengan cara menambahkan nilai vektor pada setiap piksel berdasarkan nilai maksimal vektor biner yang ada dalam sebuah citra [13]. Berbanding dengan *dilating* yang bekerja dengan cara menambahkan nilai vektor, *eroding* merupakan operasi morfologi pada citra biner untuk menghilangkan nilai yang merupakan komplemen dari data citra yang dihasilkan proses *dilating* [13]. Penjabaran *data training* dan *testing* yang digunakan dalam penelitian ini, dapat dilihat pada Tabel 1.

Tabel 1. Sebaran label data citra.

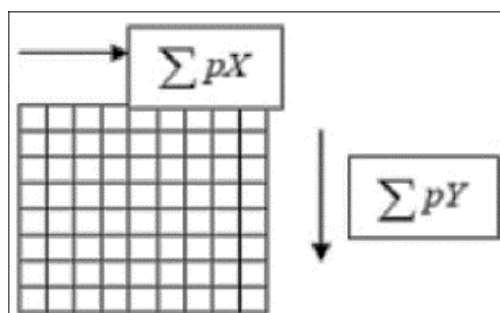
Label Citra	Jumlah Data <i>Training</i>	Jumlah Data <i>Testing</i>
"a"	2243	685
"pa"	1499	458
"ca"	191	47

Label Citra	Jumlah Data <i>Training</i>	Jumlah Data <i>Testing</i>
"da"	1638	526
"ga"	1996	637
"ha"	875	280
"ja"	415	148
"ka"	2374	757
"la"	1316	399
"ma"	2139	735
"na"	904	297
"nga"	524	171
"nya"	594	178
"pa"	2882	920
"sa"	1714	591
"ta"	2315	778
"wa"	181	73
"ya"	487	173
Jumlah	24287	7853

2.2. Feature Extraction

Feature extraction merupakan proses yang ditujukan untuk mengambil fitur atau penciri unik dari sebuah data citra, sehingga, sebuah fitur dapat diartikan sebagai representasi dari sebuah data citra [5]. Teknik *feature extraction* pada data citra memiliki beragam variasi, namun secara garis besar terbagi menjadi dua jenis teknik yakni *structural* dan *statistical feature extraction* [5][15][16][17]. *Structural feature extraction* merupakan ekstraksi fitur berdasarkan bentuk topologi dan geometris dari sebuah data citra [5][15]. Jenis elemen fitur yang termasuk dalam *structural feature extraction* seperti *cross points*, *branch points*, *end points*, dan beragam elemen lainnya. Berbeda dengan *structural feature extraction*, *statistical feature extraction* dihasilkan oleh pengukuran statistik pada sebuah data citra [5].

Salah satu contoh penggunaan *statistical feature extraction* adalah dengan menggunakan metode *projection profile*. Metode ini dilakukan dengan melakukan akumulasi jumlah piksel hitam pada baris atau kolom piksel data citra [6]. Pada penelitian ini, *projection profile* dilakukan pada baris dan kolom piksel data citra. *Projection profile* yang dilakukan pada baris piksel data citra akan menghasilkan fitur *Horizontal Projection Profile* (HPP), sedangkan pada kolom piksel akan menghasilkan fitur *Vertical Projection Profile* (VPP). Setelah didapatkan HPP dan VPP, kemudian dapat dibentuk sebuah fitur yang menggabungkan kedua jenis *projection profile* yang disebut dengan *Combined Projection Profile* (CPP). Ilustrasi *projection profile* pada baris dan kolom piksel data citra ditampilkan pada Gambar 2.



Gambar 2. Ilustrasi *projection profile* pada garis horizontal dan vertikal [18].

2.3. Klasifikasi KNN

Tahap *feature extraction* menghasilkan tiga jenis fitur yakni *Horizontal Projection Profile* (HPP), *Vertical Projection Profile* (VPP), dan *Combined Projection Profile* (CPP). Kemudian, untuk setiap jenis fitur yang

didapatkan dilakukan pemodelan dan klasifikasi menggunakan *K-Nearest Neighbors* (KNN). Algoritma KNN memiliki prinsip kerja dengan cara menemukan sekelompok “k” objek dalam *data training*, kemudian mencari “k” objek pada *data testing* yang memiliki jarak paling dekat [9]. Kemudian, KNN akan menentukan penetapan label berdasarkan dominasi label tertentu berdasarkan jumlah data dengan jarak terdekat (*neighbors*) [9].

Untuk mendapatkan nilai *nearest neighbors*, terdapat beberapa metode yang dapat digunakan dalam KNN, seperti *brute force* dan *K-D Tree*. Prinsip pencarian *nearest neighbors* menggunakan metode *brute force* adalah mencari nilai jarak untuk setiap titik data, sehingga hal ini membutuhkan waktu dan daya komputasi yang lebih besar [19]. Berbeda dengan metode *brute force*, *K-D Tree* memiliki prinsip kerja setiap titik data disimpan dalam bentuk *node* dalam sebuah *tree*, dimana *node* memiliki nilai *null* atau *pointer* ke titik data lain [19]. Penghitungan jarak antar titik data dilakukan dengan menggunakan metrik pengukuran jarak, antara lain *manhattan distance* dan *euclidean distance*. Persamaan metrik pengukuran jarak antar titik data disajikan dalam Persamaan 1 dan 2.

$$d_{s_t} = \sum_{j=1}^n |x_{s_j} - y_{t_j}| \quad (1)$$

$$d_{(s,t)} = \sqrt{\sum_{j=1}^n (x_{s_j} - y_{t_j})^2} \quad (2)$$

Persamaan 1 dan 2 menunjukkan jenis metrik pengukuran jarak yang sering digunakan, namun pada penelitian ini penghitungan jarak antar titik data dilakukan dengan menggunakan *euclidean distance*. Pemodelan dan klasifikasi KNN dilakukan dengan menggunakan dua jenis model yang berbeda yakni *base KNN model*, dan model KNN yang dioptimasi menggunakan *GridSearchCV*. Konfigurasi yang digunakan untuk membentuk *base KNN model* dapat dilihat pada Tabel 2. Kemudian, model KNN yang dioptimasi dengan menggunakan *GridSearchCV* memiliki konfigurasi yang dapat dilihat pada Tabel 3.

Tabel 2. Konfigurasi *Base KNN Model*

Nama Konfigurasi	Nilai Konfigurasi
<i>algorithm</i>	“brute”
<i>metric</i>	“euclidean”
<i>nearest neighbors</i>	5
<i>weights</i>	“uniform”

Tabel 3. Konfigurasi *GridSearchCV KNN Model*

Nama Konfigurasi	Nilai Konfigurasi
<i>algorithm</i>	“kd_tree”
<i>metric</i>	“euclidean”
<i>nearest neighbors</i>	<i>list(range(5,3000,65))</i>
<i>weights</i>	[“uniform”, “distance”]

Pemodelan KNN menggunakan *GridSearchCV* menggunakan *k-fold cross-validation* dengan nilai $k = 10$. Pemilihan nilai $k=10$ karena nilai 10 pada *k-fold cross-validation* merupakan standar umum untuk melakukan *cross-validation* terhadap algoritma klasifikasi [20].

2.4. Evaluasi dan Penarikan Kesimpulan

Evaluasi klasifikasi menggunakan *base KNN model* dan *GridSearchCV KNN model* dilakukan, untuk setiap jenis fitur *projection profile* yang digunakan. Hasil evaluasi, kemudian akan dirangkum dengan menggunakan *confusion matrix*. Bentuk dasar *confusion matrix* adalah sebuah matriks 2x2 [21]. Penjabaran bentuk dasar *confusion matrix* dapat dilihat pada Tabel 4.

Tabel 4. Bentuk dasar *confusion matrix*.

	<i>Predicted Positive</i>	<i>Predicted Negative</i>
<i>Actual Positive</i>	True Positive (TP)	False Negative (FN)
<i>Actual Negative</i>	False Positive (FP)	True Negative (TN)

Nilai *true positive* (TP) merupakan nilai yang didapatkan ketika hasil prediksi menyatakan bahwa sebuah data memiliki label positif, dan kondisi aktual data tersebut merupakan label positif. *False negative* (FN) adalah ketika prediksi menyatakan bahwa sebuah data memiliki label negatif, namun kondisi aktual data tersebut merupakan label positif. *False positive* (FP), adalah ketika prediksi menyatakan bahwa sebuah data memiliki label positif, namun kondisi aktual data tersebut memiliki label negatif. *True negative* (TN), adalah kondisi ketika prediksi menyatakan bahwa sebuah data memiliki label negatif, dan kondisi aktual data tersebut merupakan label negatif.

Berdasarkan hasil perangkuman performa klasifikasi yang didapat menggunakan *confusion matrix*, kemudian hasil tersebut dievaluasi menggunakan beberapa metrik evaluasi kinerja klasifikasi. Metrik evaluasi tersebut adalah *accuracy* (ACC), *sensitivity* (SN), *specificity* (SP), *precision* (PR), dan *matthew correlation coefficient* (MCC). Definisi metrik evaluasi yang telah disebutkan, dapat dilihat dalam Persamaan 3 sampai dengan 7.

Accuracy (ACC)

$$\frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

Sensitivity (SN)

$$\frac{TP}{TP+FN} \quad (4)$$

Specificity (SP)

$$\frac{TN}{TN+FP} \quad (5)$$

Precision (PR)

$$\frac{TP}{TP+FP} \quad (6)$$

Matthew Correlation Coefficient (MCC)

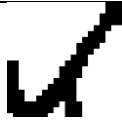
$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (7)$$

3. HASIL DAN PEMBAHASAN

3.1. Data Preprocessing

Pada tahap ini setiap data pada *data training* dan *data testing* melalui beberapa proses yakni *noise reduction*, *thresholding*, dan operasi morfologi berupa *dilating* dan *eroding*. Proses *noise reduction* dilakukan menggunakan *gaussian filter*, karena data citra yang didapatkan melalui *scanner* cenderung memiliki jenis *noise gaussian* [11]. Kemudian, data citra tulisan karakter aksara Lampung akan melalui proses *thresholding* untuk memperjelas pola yang dimiliki data citra dan memberi nilai pembeda yang signifikan dengan latar belakang yang terdapat pada citra [12]. Setelah melalui proses *thresholding*, kemudian pola karakter tulisan aksara Lampung disempurnakan menggunakan operasi morfologi *dilating* dan *eroding*. Sampel hasil tahap data *preprocessing* dapat dilihat dalam Tabel 5.

Tabel 5. Sampel hasil tahap *data preprocessing*.

Label Citra	Hasil Data Preprocessing
"pa"	
"ma"	
"la"	
"da"	
"ka"	

Data yang disajikan pada Tabel 5 menunjukkan sampel hasil data *preprocessing*. Pada Tabel 5, terdapat lima dari delapan belas label citra induk huruf aksara Lampung. Tahap data *preprocessing* dilakukan terhadap *data training* maupun *data testing*. Pemberian perlakuan yang sama pada *data training* maupun *data testing* bertujuan untuk meminimalkan bias yang mungkin muncul dikarenakan perbedaan perlakuan yang dilakukan pada data.

3.2. Feature Extraction

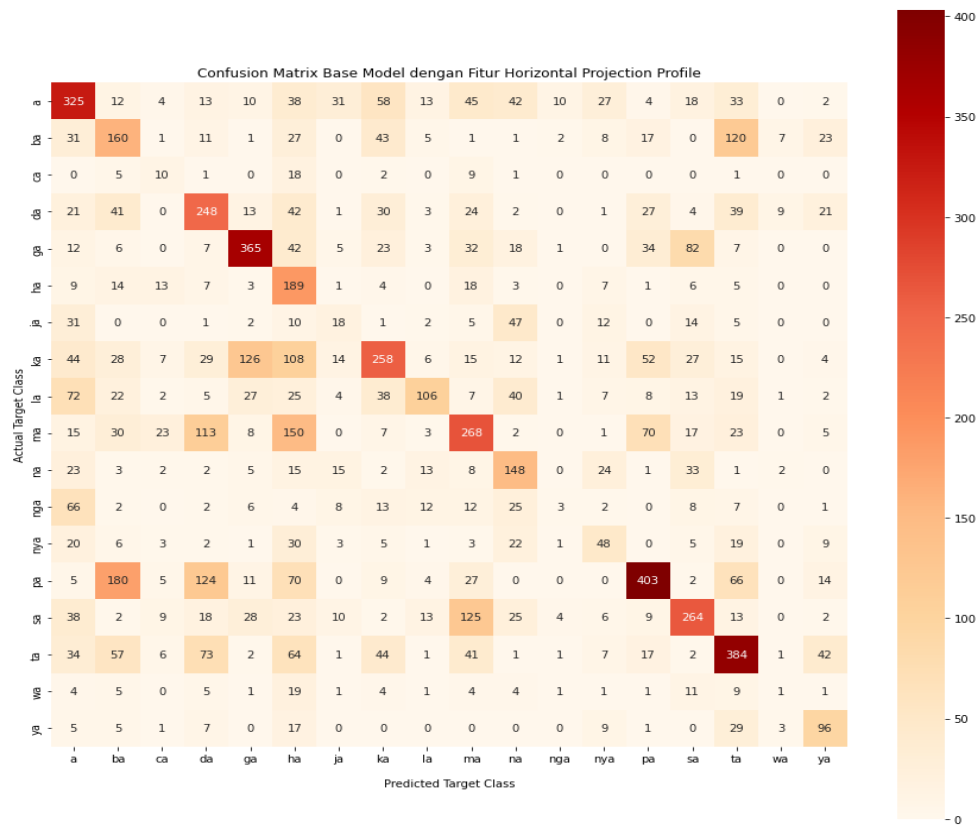
Metode *projection profile* digunakan untuk mendapatkan fitur dalam tahap *feature extraxtion*. Fitur yang didapatkan adalah *Horizontal Projection Profile* (HPP), *Vertical Projection Profile* (VPP), dan fitur baru yang didapatkan dengan cara menggabungkan HPP dan CPP. Data citra yang digunakan memiliki dimensi 20 x 20 piksel, perbandingan jumlah fitur yang didapatkan untuk setiap jenis *projection profile* yang dilakukan disajikan pada Tabel 6. Data yang ditunjukkan pada Tabel 6, menunjukkan perbedaan jumlah fitur yang didapat berdasarkan jenis *projection profile* yang digunakan untuk mendapatkan fitur.

Tabel 6. Jumlah fitur per data citra

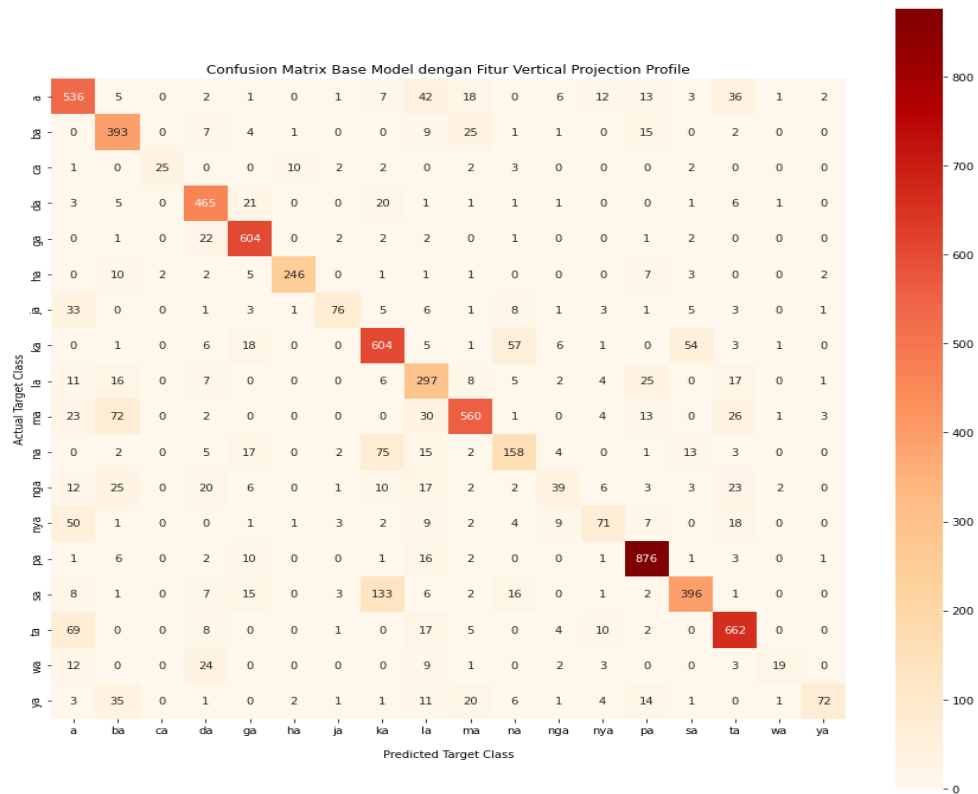
Jenis Fitur	Label Fitur	Jumlah Fitur
<i>Horizontal Projection Profile</i> (HPP)	Label hr_px1 s.d hr_px20	20
<i>Vertical Projection Profile</i> (VPP)	Label vr_px1 s.d vr vr_px20	20
<i>Combined Projection Profile</i> (CPP)	Label hr_px1 s.d hr_px20, dan Label vr_px1 s.d vr_px20	40

3.3. Klasifikasi *K-Nearest Neighbors* (KNN)

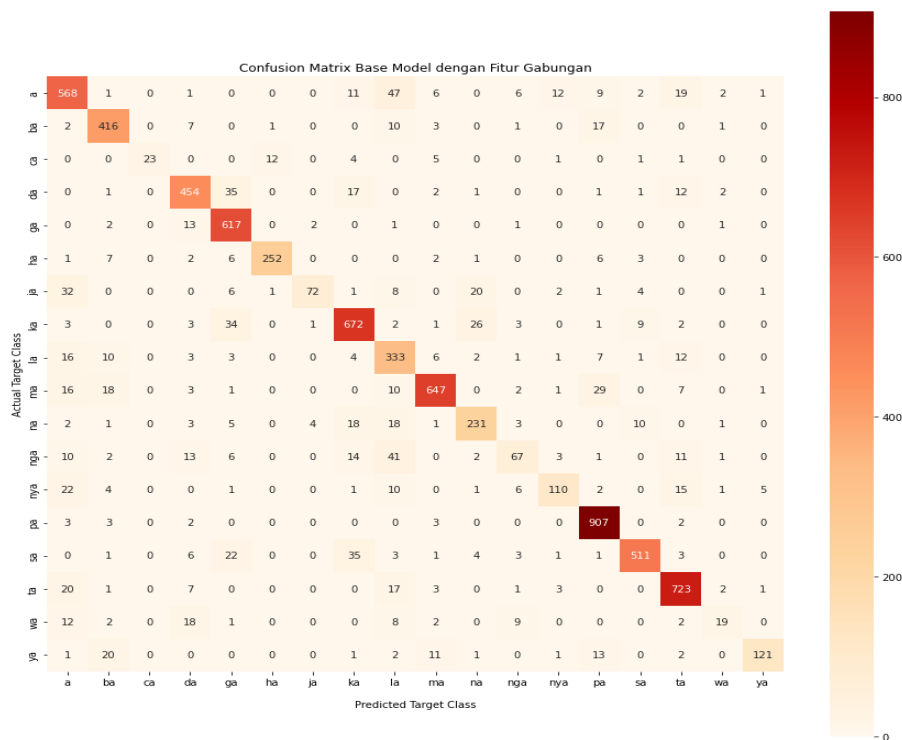
Tahap klasifikasi dilakukan terhadap tiga jenis fitur yang didapat pada tahap *feature extraction*. Penggunaan ketiga jenis fitur untuk klasifikasi ditujukan untuk melihat jenis fitur apa yang menghasilkan performa klasifikasi paling baik. Selain itu, klasifikasi dilakukan dengan menggunakan dua jenis model KNN yakni, *base KNN model*, dan *GridSearchCV KNN model*. *Confusion matrix* klasifikasi KNN dengan menggunakan *base KNN model* dapat dilihat pada Gambar 3, 4, dan 5.



Gambar 3. Confusion matrix Base KNN Model fitur Horizontal Projection Profile (HPP).

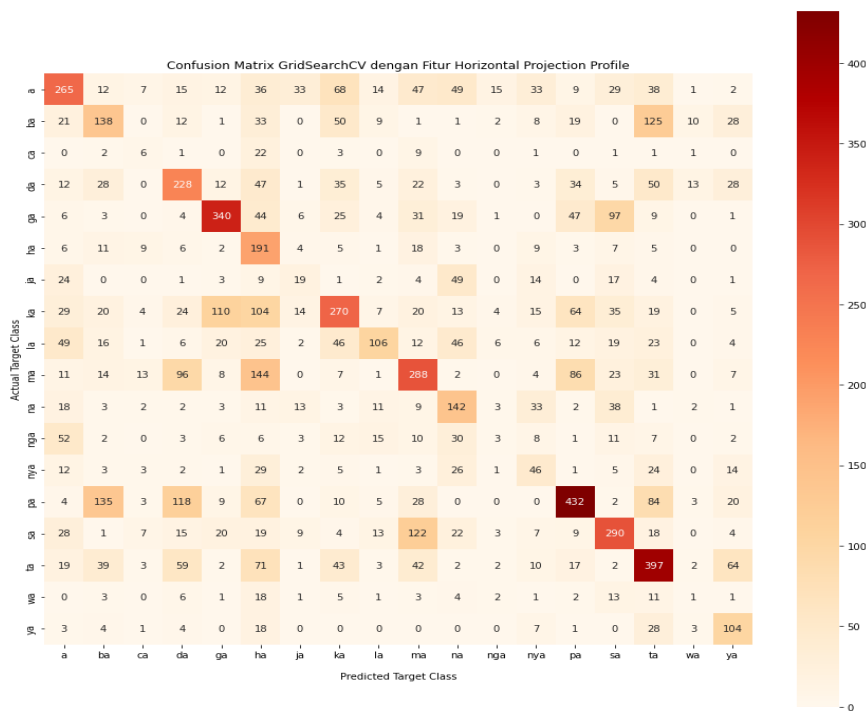


Gambar 4. Confusion matrix Base KNN Model fitur Vertical Projection Profile (VPP).

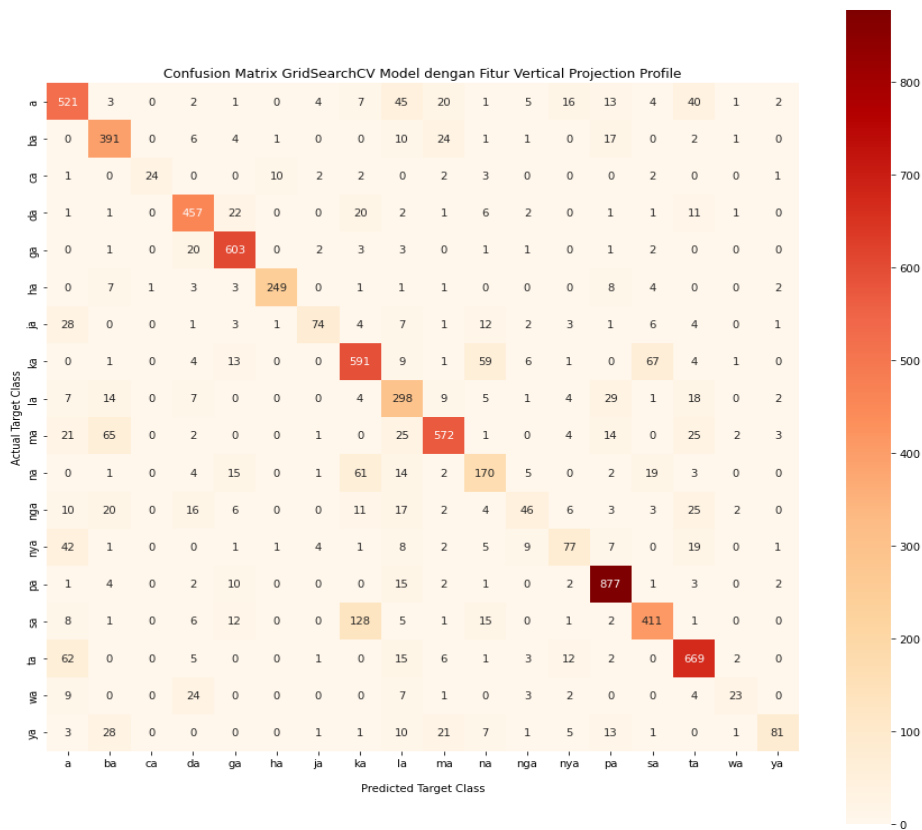


Gambar 5. Confusion matrix Base KNN Model fitur Combined Projection Profile (CPP).

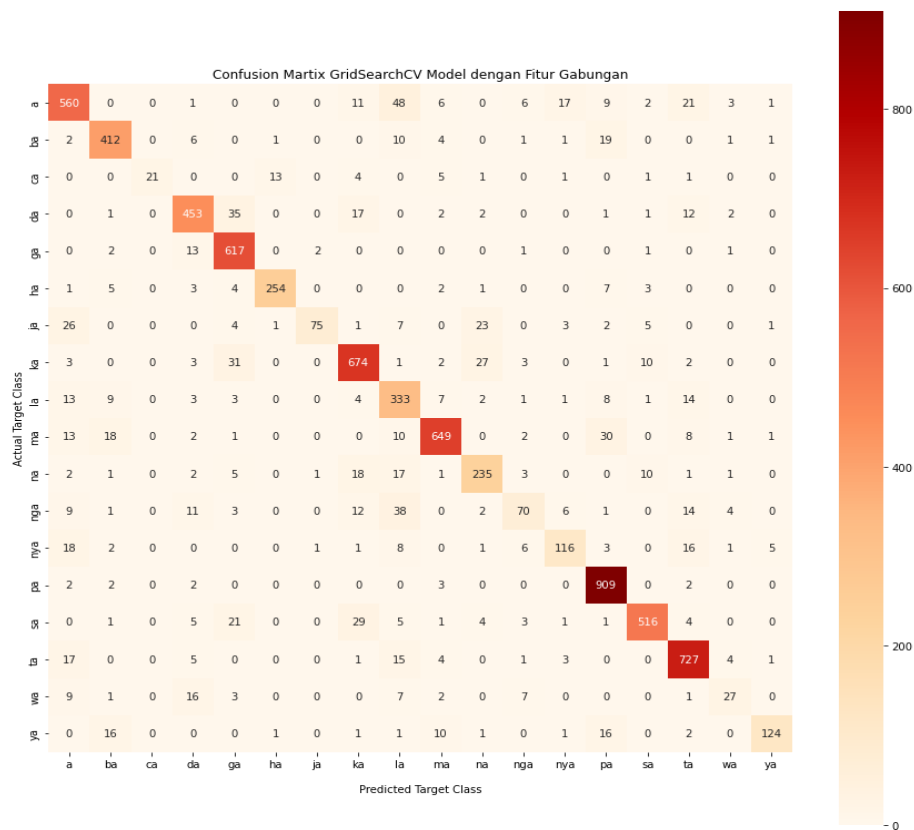
Confusion matrix yang ditunjukkan pada Gambar 3,4, dan 5 merupakan confusion matrix yang didapat dari hasil klasifikasi untuk ketiga jenis fitur dengan menggunakan base KNN model. Selanjutnya, confusion matrix yang didapat dari hasil klasifikasi untuk ketiga jenis fitur dengan menggunakan GridSearchCV KNN model dapat dilihat pada Gambar 6,7, dan 8.



Gambar 6. Confusion matrix GridSearchCV KNN Model fitur Horizontal Projection Profile (HPP).



Gambar 7. Confusion matrix GridSearchCV KNN Model fitur Vertical Projection Profile (VPP).



Gambar 8. Confusion matrix GridSearchCV KNN Model fitur Combined Projection Profile (CPP)

Gambar 6,7, dan 8 menunjukkan *confusion matrix* yang didapatkan menggunakan *GridSearchCV* KNN model terhadap ketiga jenis fitur. *Confusion matrix* yang telah ditunjukkan Gambar 3 sampai dengan 8 digunakan sebagai bahan evaluasi untuk melihat performa klasifikasi KNN pada setiap jenis fitur.

3.4. Evaluasi dan Penarikan Kesimpulan

Evaluasi dilakukan dengan menggunakan beberapa metrik pengukuran performa klasifikasi yakni *accuracy* (ACC), *sensitivity* (SN), *specificity* (SP), *precision* (PR), dan *matthew correlation coefficient* (MCC). Hasil evaluasi performa klasifikasi tulisan karakter aksara Lampung dapat dilihat pada Tabel 7.

Tabel 7. Evaluasi klasifikasi KNN.

Metrik	Base KNN Model			GridSearchCV KNN Model		
	HPP	VPP	CPP	HPP	VPP	CPP
ACC	41,94	77,66	85,86	41,58	78,11	86,23
SN	41,94	77,66	85,86	41,58	78,11	86,23
SP	96,58	98,68	99,16	96,56	98,71	99,19
PR	41,94	77,66	85,86	41,58	78,11	86,23
MCC	37,63	75,80	84,69	37,21	37,21	85,09

Keterangan:

ACC: Accuracy; SN: Sensitivity; SP: Specificity; PR: Precision; MCC: Matthew Correlation Coefficient; HPP: Horizontal Projection Profile; VPP: Vertical Projection Profile; CPP: Combined Projection Profile

Data evaluasi klasifikasi menggunakan dua jenis model KNN terhadap ketiga jenis fitur memiliki kesamaan untuk hasil klasifikasi terbaik diraih dengan fitur *Combined Projection Profile* (CPP). Hasil tersebut diikuti dengan fitur *Vertical Projection Profile* (VPP), dan *Horizontal Projection Profile* (HPP). Selain itu, model KNN yang dioptimasi menggunakan *GridSearchCV* mendapatkan mayoritas hasil klasifikasi yang lebih baik dibandingkan dengan *base KNN model*.

Hasil buruk yang didapatkan fitur *Horizontal Projection Profile* disebabkan kecenderungan metode *Horizontal Projection Profile* untuk digunakan sebagai segmentasi baris dan kurang cocok digunakan untuk mendapatkan fitur dalam sebuah karakter tunggal [6]. Kemudian, hasil baik yang didapatkan fitur *Combined Projection Profile* disebabkan gabungan HPP dan VPP, merepresentasikan pola dalam citra dengan lebih baik [6]. Selain itu, hal tersebut juga meningkatkan data statistik yang berperan dalam peningkatan hasil klasifikasi menggunakan KNN [9][15][17].

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan untuk melakukan pengenalan karakter citra tulisan tangan aksara Lampung, model yang dioptimasi menggunakan *GridSearchCV* mendapat hasil yang lebih baik dibanding dengan *base KNN model*. Hasil yang diraih *Combined Projection Profile* mendapatkan hasil paling baik, dikarenakan penggunaan fitur gabungan menjadikan representasi statistik sebuah citra menjadi lebih baik. Di sisi lain, walaupun fitur *Horizontal Projection Profile* (HPP) menghasilkan klasifikasi terburuk, fitur VPP memberikan dampak yang cukup signifikan dalam peningkatan performa hasil klasifikasi apabila digabungkan dengan fitur HPP.

DAFTAR PUSTAKA

- [1] A. Junaidi, S. Vajda, & G. A. Fink, "Lampung - A new handwritten character benchmark: Database, labeling and recognition," *ACM International Conference Proceeding Series*, 2011.

- [2] A. Andana, R. Widyati, & M. Irzal, "Pengenalannya Citra Tulisan Tangan Dengan Metode Backpropagation," *Jurnal Matematika Terapan*, vol. 2, no. 1, pp. 36–44, 2018.
- [3] M. K. Mahto, K. Bhatia, & R. K. Sharma, "Combined horizontal and vertical projection *feature extraction* technique for Gurmukhi handwritten character recognition," *Conference Proceeding - 2015 International Conference on Advances in Computer Engineering and Applications, ICACEA 2015*, pp. 59–65, 2015.
- [4] B. D. Puspasari, "Aplikasi Pengenalannya Huruf Alfabet Bagi Anak Usia Dini Menggunakan Metode Pengolahan Citra Berbasis Data Suara," *Jurnal ELTEK*, vol. 11, no. 01, pp. 1693–4024, 2013.
- [5] A. Junaidi, "Lampung Handwritten Character Recognition," Dissertation, Technical University Dortmund, Dortmund, 2016.
- [6] A. M. M. O. Chacko & P. M. Dhanya, "A Comparative Study of Different Feature Extraction Techniques for Offline Malayalam Character Recognition," *Smart Innovation, Systems and Technologies*, vol. 32, 2015, pp. 9–18.
- [7] L. Likforman-Sulem, A. Zahour, & B. Taconet, "Text line segmentation of historical documents: A survey," *International Journal on Document Analysis and Recognition*, vol. 9, no. 2–4, pp. 123–138, 2007.
- [8] J. Han, M. Kamber, & J. Pei, *Data Mining: Concepts and Techniques*, Amsterdam: Elsevier, 2012.
- [9] X. Wu & V. Kumar, *The Top Ten Algorithms in Data Mining-Data Mining and Knowledge Discovery*, Florida: CRC Press, 2009.
- [10] W. Wahyono & A. Nugroho, "Penerapan Algoritma C4.5 Untuk Prediksi Tingkat Kompetensi Karyawan Pt Multistrada Arah Sarana," *Joutica*, vol. 3, no. 1, p. 145, 2018.
- [11] P. Cattin, "Image Restoration Introduction to Signal and Image Processing," *MIAC*, vol. 9, no. 19, pp. 1–70, 2016.
- [12] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Trans Syst Man Cybern*, vol. 9, no. 1, pp. 62–66, Jan. 1979.
- [13] G. X. Ritter & J. N. Wilson, *Handbook of Computer Vision Algorithms in Image Algebra*, Florida: CRC Press, 2000.
- [14] R. Szeliski, *Computer Vision: Algorithms and Applications 2nd Edition*, New York: Springer, 2021.
- [15] J. Memon, M. Sami, R. A. Khan, & M. Uddin, "Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR)," *IEEE Access*, vol. 8, pp. 142642–142668, 2020, doi: 10.1109/ACCESS.2020.3012542.
- [16] A. Kacem, N. Aouiti, & A. Belaïd, "Structural features extraction for handwritten Arabic personal names recognition," in *Proceedings - International Workshop on Frontiers in Handwriting Recognition, IWFHR*, 2012, pp. 268–273. doi: 10.1109/ICFHR.2012.276.
- [17] B. M. Vinjit, M. K. Bhojak, S. Kumar and G. Chalak, "A Review on Handwritten Character Recognition Methods and Techniques," *2020 International Conference on Communication and Signal Processing (ICCSP)*, Chennai, India, 2020, pp. 1224–1228, doi: 10.1109/ICCSP48568.2020.9182129.
- [18] B. Amaliah, A. Yuniarti, A. S. Nugroho, & A. Z. Arifin, "Pemisahan Gigi Pada Dental Panoramic Radiograph Dengan Menggunakan Integral Projection yang di Modifikasi," *Jurnal ilmiah KURSOR*, vol. 6, no. 2, pp. 121–128, 2011.
- [19] J. L. Bentley, "Multidimensional Binary Search Trees Used for Associative Searching," *Commun ACM*, vol. 19, 1975.
- [20] P. Refaeilzadeh, L. Tang, & H. Liu, "Cross-Validation," in *Encyclopedia of Database Systems*, Boston, MA: Springer US, 2009, pp. 532–538.

- [21] M. Bekkar, H. K. Djemaa, & T. A. Alitouche, "Evaluation Measures for Models Assessment over Imbalanced Data Sets," *Journal of Information Engineering and Applications*, vol. 3, no. 10, pp. 27–38, 2013.