

KLASIFIKASI PENYAKIT PADA TANAMAN KOPI ARABIKA MENGGUNAKAN METODE K-NEAREST NEIGHBOR (KNN) BERBASIS CITRA**Sandri¹, Gidion A.N. Pongdatu², Juprianus Rusman³**sandrimatarru@gmail.com, dionpongdatu@ukitoraja.ac.id, rusman.jr@ukitoraja.ac.id

Program Studi Teknik Informatika Universitas Kristen Indonesia Toraja

Informasi Artikel

Diterima : 9 Mar 2023

Direview : 30 Mar 2023

Disetujui : 20 Apr 2023

Kata KunciPenyakit Kopi,
Klasifikasi, GLCM, KNN,
Confusion Matrix**Abstrak**

Salah satu tanaman yang tumbuh di daerah tropis dan memiliki harga jual yang cukup tinggi serta menjadi sumber devisa bagi Indonesia ialah kopi. Kualitas dan produksi kopi akan menurun jika kopi mudah terserang penyakit yang disebabkan oleh perubahan iklim seperti cuaca, suhu, kelembapan udara, tanah, perawatan tanaman yang kurang maksimal dan ketinggian lahan. Semakin berkembangnya teknologi saat ini diperlukan suatu metode kecerdasan buatan untuk membantu petani dalam mengenali jenis penyakit kopi. Metode yang digunakan untuk melakukan klasifikasi penyakit kopi arabika ialah K-Nearest Neighbor. Adapun Dataset yang diproses dibagi menjadi 2 yakni data training sebesar 320 citra dan data testing sebanyak 80 citra selanjutnya dilakukan tahapan image preprocessing. Setelah itu dilakukan ekstraksi glcm yakni fitur yang digunakan contrast, homogeneity, correlation, energy. Pada penelitian ini hasil percobaan dari beberapa nilai K pada K-Nearest Neighbor menunjukkan akurasi tertinggi terdapat pada K=11 dengan hasil pengujian model menggunakan confusion matrix memperoleh tingkat akurasi sebesar 94 %.

KeywordsCoffee Disease, Classification,
GLCM, KNN, Confusion
Matrix**Abstract**

One of the plants that grows in the tropics and has a fairly high selling price and is a source of foreign exchange for Indonesia is coffee. The quality and production of coffee will decrease if the coffee is susceptible to diseases caused by climate change such as weather, temperature, humidity, soil, less than optimal plant maintenance and land elevation. The current development of technology requires an artificial intelligence method to help farmers recognize coffee disease types. The method used to classify Arabica coffee diseases is K-Nearest Neighbor. The processed dataset is divided into 2, namely training data of 320 images and testing data of 80 images, then the image preprocessing stage is carried out. After that, glcm extraction is performed, namely the features used are contrast, homogeneity, correlation, energy. In this study the experimental results of several K values in the K-Nearest Neighbor showed that the highest accuracy was found at K=11 with the results of model testing using the confusion matrix obtaining an accuracy rate of 94%.

A. Pendahuluan

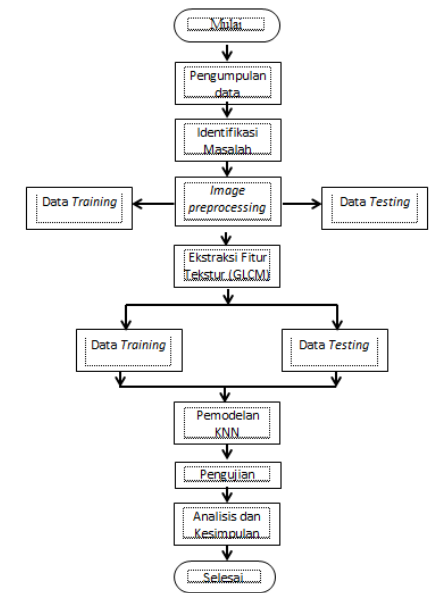
Kopi merupakan salah satu tanaman perkebunan di Indonesia yang memiliki nilai ekonomis yang cukup tinggi dan sudah lama menjadi tanaman yang dibudidayakan[1]. Salah satu daerah produksi kopi dengan kualitas terbaik di Indonesia yang dikenal hingga manca negara adalah Toraja. Toraja merupakan daerah dataran tinggi serta daerah pegunungan yang memiliki ketinggian dengan kisaran dari 300m sampai 2500m dari permukaan laut yang berada di Sulawesi Selatan. Secara umum ada dua jenis kopi yaitu kopi arabika dan kopi robusta, pertumbuhan dan produksi kopi sangat tergantung pada keadaan iklim, tanah, perawatan yang kurang maksimal dan ketinggian lahan. Dari dua jenis kopi tersebut kopi arabika (*arabica coffee*) atau biasa di sebut *Queen Of Coffee* yang diprioritaskan di Toraja karena memiliki ciri aroma yang khas seperti aroma tanah, rasa pahitnya menonjol namun dengan kadar asam yang rendah dan memiliki harga nilai jual yang tinggi. Kopi arabika dapat hidup juga didataran rendah sampai dataran yang lebih tinggi, tetapi apabila ditanam didataran yang lebih rendah atau lebih tinggi kurang produktif karena jika berada ditinggian kisaran 1000 sampai 1800 mdpl akan mudah terserang penyakit karat daun maupun bercak daun sebaliknya kopi dapat bertumbuh secara *vegetative* jika ditanam di dataran tinggi yang lebih tinggi dari 1850 mdpl dan udara akan terlalu dingin maka mengakibatkan kualitas kopi arabika menurun dan kurangnya produksi sehingga kerugian besar bagi petani kopi dan negara Indonesia. Dengan demikian kopi arabika lebih rentan terserang penyakit akibatnya petani kopi mengalami kesulitan dalam mengidentifikasi serta mengelompokan berbagai macam penyakit kopi.[1][2]

Salah satu cara untuk mengidentifikasi ialah menggunakan teknologi computer vision yakni pengolahan citra. Adapun konsep pengolahan citra yakni pengenalan pola. Pengenalan pola adalah proses pengenalan pola gambar melalui komputer yang dapat dideteksi atau dibaca. Salah satu ekstraksi fitur yang digunakan untuk membedakan gambar sebelum diklasifikasikan dengan cara mengambil ciri pada suatu citra atau dapat dikatakan teknik mendapatkan ciri pada citra dan diolah secara lebih lanjut ialah fitur tekstur *Gray Level Co-ocurance Matrix* (GLCM). [3]

Dengan semakin berkembangnya teknologi, proses klasifikasi penyakit kopi dapat menggunakan metode kecerdasan buatan pada salah satu cabang yakni *machine learning*. Metode *machine learning* yang sudah banyak diterapkan dalam melakukan klasifikasi ialah *K-Nearest Neighbor*. *K-Nearest Neighbor* merupakan sebuah algoritma yang menggunakan metode *supervise learning* untuk dapat digunakan dalam klasifikasi sebuah objek berdasarkan kelas yang sama dengan kelas mayoritas tetangga terdekat. Jarak dihitung dengan menggunakan rumus *Euclidean distance*[4][5]. Pada tahun 2020 Annida dkk melakukan penelitian tentang deteksi penyakit daun pada tanaman padi menggunakan algoritma *decision tree random forest*, *naïve bayes*, *SVM* dan *KNN*. Dari kelima metode yang digunakan metode memberikan nilai akurasi terbaik ialah KNN dengan nilai akurasi 87%[6]. Tujuan penelitian ini yakni untuk memodelkan serta mengetahui kinerja model pada metode KNN dalam mengklasifikasikan penyakit pada tanaman kopi arabika.

B. Metode Penelitian

Tahapan penelitian ini menggambarkan proses penelitian yang akan ditempuh sekaligus menggambarkan penelitian secara keseluruhan. Tahapan penelitian pada klasifikasi penyakit pada tanaman kopi arabika menggunakan metode KNN



Gambar 1. Flowchart Penelitian

1. Pengumpulan Data

Pengumpulan data yang terbagi menjadi 2 tahapan yaitu observasi dan wawancara ke lokasi perkebunan kopi arabika yang berada di Batutumonga, Sesean, Kabupaten Toraja Utara kemudian pengambilan citra penyakit kopi dengan spesifikasi kamera 3008 x 2000 dilakukan dari berbagai sudut dengan jarak 50 cm. Citra yang diproses sebanyak 400 citra dengan masing-masing 200 citra yang terdiri dari bercak daun dan karat daun.



Gambar 2. Citra Daun Kopi (a) Bercak Daun (b) Karat Daun

Setelah melakukan pengambilan data citra yakni melakukan pembagian dataset dibagi menjadi 2 macam data yakni data *Training* dan data *Testing* dengan menggunakan perbandingan 80:20.

Tabel 1. Pembagian Dataset

Kelas	Training	Testing
Bercak Daun	160	40
Karat Daun	160	40
Total	320	80

2. Image Preprocessing

Beberapa tahap sebelum melakukan ekstraksi ialah Proses *image preprocessing* yakni melakukan *cropping* dengan tujuan untuk memperkecil ukuran citra dengan mengambil bagian citra yang dibutuhkan, *segmentasi* dengan tujuan

untuk memisahkan objek dan background dan *resize* bertujuan untuk menyamakan ukuran citra serta memudahkan untuk proses pengolahan citra dengan ukuran 200 x 200 piksel pada setiap citra dengan menggunakan aplikasi *adobe photoshop* selanjutnya dilakukan konversi citra dari RGB menjadi *grayscale*. Berikut proses *image preprocessing* dibawah ini



Gambar 3. Proses *Image Preprocessing* (a) Cropping (b) Segmentasi (c) Resize (d) Konversi Citra RGB Menjadi Grayscale

3. Ekstraksi *Gray Level Co-Occurance Matrix*

Ekstraksi Fitur Tekstur *Gray Level Co-Occurrence Matrix* (GLCM) Merupakan teknik ekstraksi ciri yang efektif untuk melakukan klasifikasi dan identifikasi karena informasi yang diberikan sangat detail berdasarkan citra dalam hal tekstur[7][8]. Adapun tahapan yang akan dilakukan pada ekstraksi fitur GLCM sebagai berikut:

Langkah 1 : Melakukan konversi citra RGB menjadi Grayscale

Langkah 2 : Menentukan orientasi sudut dan *distance* (jarak)

Langkah 3 : Menghitung kemunculan nilai yang sama pada keseluruhan matriks

Langkah 4 : Melakukan normalisasi pada matriks GLCM

Langkah 5 : Menghitung nilai fitur pada GLCM yakni

$$- \text{contrast}, \quad \sum_{i=1}^K \sum_{j=1}^K (i,j)^2 * P(i,j) \quad (1)$$

$$- \text{homogeneity}, \quad \sum_{i=1}^K \sum_{j=1}^K \frac{P_{ij}}{1 + |i - j|} \quad (2)$$

$$- \text{energy}, \quad \sum_{i=1}^K \sum_{j=1}^K P(i,j)^2 \quad (3)$$

$$- \text{corelation} \quad \bullet \quad \mu = \sum_i^n \sum_j^n I_p(I,j) \quad (4)$$

$$\bullet \quad \sigma = \sqrt{\sum_i \sum_j (i - \mu_i)^2 P(i,j)} \quad (5)$$

$$\bullet \quad \sum_{i=1}^K \sum_{j=1}^K \frac{(i-\mu)(j-\mu)P(i,j)}{\sigma_i \sigma_j} \quad (6)$$

4. Klasifikasi *K-Nearest Neighbor*

KNN adalah metode klasifikasi berdasarkan data terdekat dengan objek yang akan diklasifikasi. Data pembelajaran diproyeksikan kedalam sebuah ruang multidimensi yang mana masing-masing dimensi merepresentasikan fitur dari data. Ruang multidimensi tersebut dibagi kedalam bagian-bagian berdasarkan

klasifikasi data pembelajaran. Dekat atau jauhnya tetangga dapat dihitung berdasarkan jarak *Euclidean*[9]. Berikut Rumus KNN

$$Dist(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (7)$$

5. Confusion Matrix

Confusion matrix merupakan model dalam mengklasifikasikan sebuah tabel yang terdapat banyaknya baris data uji untuk diprediksi benar dan tidak benar yang diketahui pada pengukuran tingkat akurasi[10]. Berikut rumus *confusion matrix* yang digunakan untuk menghitung *akurasi*, *precision*, *recall* dan *F1 Score*

$$akurasi = \frac{TP + TN}{TP + FN + FP + TN} \quad (8)$$

$$Precision = \frac{TP}{TP + FP} \quad (9)$$

$$Recall = \frac{TP}{TP + FN} \quad (10)$$

$$F1\ Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (11)$$

C. Hasil dan Pembahasan

1) Ekstraksi Fitur GLCM

Adapun dataset yang digunakan dalam penelitian ini ialah data *training* 320 citra dan data *testing* 80 citra, selanjutnya data tersebut dilakukan proses ekstraksi fitur GLCM untuk mendapatkan ciri pada tekstur pada citra penyakit kopi menggunakan bahasa pemrograman Python. Berikut hasil ekstraksi GLCM menggunakan orientasi sudut $0^\circ, 45^\circ, 90^\circ, 135^\circ$ dengan jarak = 1

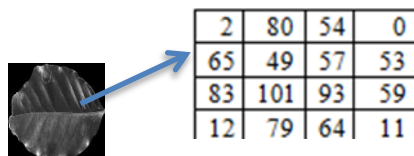
Tabel 2. Data *Training* Yang Telah Di Ekstraksi

File	Con 0	...	Con 135	Hom 0	...	Hom 135	Ene 0	...	Ene 135	Cor 0	...	Cor 135	Class
BD1. Jpg	202, 95	...	340, 98	0,372	...	0,342	0,205	...	0,200	0,942	...	0,902	BD
BD2. Jpg	275, 40	...	596, 72	0,398	...	0,337	0,197	...	0,190	0,960	...	0,914	BD
BD3. Jpg	262, 71	...	490, 11	0,421	...	0,386	0,254	...	0,252	0,944	...	0,896	BD
...	...	...	...	...	...	...	...	...	...	...	...	...	...
BD 160. Jpg	204, 85	...	363, 12	0,378	...	0,358	0,242	...	0,236	0,977	...	0,960	BD
KD1. Jpg	227, 06	...	316, 29	0,311	...	0,290	0,164	...	0,159	0,979	...	0,971	KD
KD2. Jpg	143, 14	...	188, 70	0,377	...	0,359	0,243	...	0,233	0,978	...	0,970	KD
KD3. Jpg	119, 86	...	283, 95	0,442	...	0,377	0,292	...	0,284	0,981	...	0,955	KD
...	...	...	...	...	...	...	...	...	...	...	...	...	...
KD 160. Jpg	352, 17	...	390, 51	0,390	...	0,370	0,284	...	0,279	0,976	...	0,973	KD

Tabel 3. Data *Testing* Yang Telah Di Ekstraksi

File	Con 0	...	Con 135	Hom 0	...	Hom 135	Ene 0	...	Ene 135	Cor 0	...	Cor 135	Class
BD1. Jpg	192, 99	...	464, 92	0,394	...	0,364	0,256	...	0,251	0,982	...	0,958	BD
BD2. Jpg	155, 19	...	327, 20	0,441	...	0,377	0,261	...	0,253	0,983	...	0,965	BD
...	...	...	...	...	...	...	...	...	...	...	...	...	...
BD 40. Jpg	180, 98	...	302, 56	0,408	...	0,380	0,262	...	0,256	0,984	...	0,974	BD
KD1. Jpg	305, 55	...	670, 25	0,457	...	0,425	0,312	...	0,307	0,979	...	0,955	KD
KD2. Jpg	243, 43	...	645, 56	0,507	...	0,468	0,332	...	0,327	0,982	...	0,954	KD
...	...	...	...	...	...	...	...	...	...	...	...	...	...
KD 40. Jpg	341, 09	...	583, 33	0,323	...	0,307	0,206	...	0,208	0,976	...	0,959	KD

Pada kasus dibawah ini merupakan cara untuk menghitung GLCM yang diilustrasikan skala intensitas dari 0 sampai 101. Berikut tahapan untuk mendapatkan nilai dari ekstraksi *gray level co-ocurance matrix*

**Gambar 4.** Hasil Konversi Citra RGB Menjadi *Grayscale*

- a. Menghitung keseluruhan matriks dengan kemunculan nilai kemunculan yang sama, misalnya nilai intensitas 2 dan 80 maka nilai kemunculannya 1

MENGHITUNG MATRIKS YANG BERPASANGAN DENGAN 0 DERAJAT													
	0	2	11	12	49	53	54	57	59	64	65	79	80
0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0
49	0	0	0	0	0	0	0	0	0	0	0	0	0
53	0	0	0	0	0	0	0	0	0	0	0	0	0
54	1	0	0	0	0	0	0	0	0	0	0	0	0
57	0	0	0	0	0	0	0	0	0	0	0	0	0
59	0	0	0	0	0	0	0	0	0	0	0	0	0
64	0	0	0	0	0	0	0	0	0	0	0	0	0
65	0	0	0	0	0	0	0	0	0	0	0	0	0
79	0	0	0	0	0	0	0	0	0	0	0	0	0
80	0	0	0	0	0	0	0	0	0	0	0	0	0
83	0	0	0	0	0	0	0	0	0	0	0	0	0
93	0	0	0	0	0	0	0	0	0	0	0	0	0
101	0	0	0	0	0	0	0	0	0	0	0	0	0

Gambar 5. Hasil Perhitungan Kemunculan Nilai Yang Sama

- b. Selanjutnya melakukan normalisasi pada matriks glcm pada gambar 7 dengan cara nilai setiap intensitas pada baris dibagi jumlah keseluruhan kemunculan nilai yang sama yakni 12.

MELAKUKAN NORMALISASI DATA													
	0	2	11	12	49	53	54	57	59	64	65	79	80
0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0
49	0	0	0	0	0	0	0	0	0	0	0	0	0
53	0	0	0	0	0	0	0	0	0	0	0	0	0
54	0.08333333	0	0	0	0	0	0	0	0	0	0	0	0
57	0	0	0	0	0	0	0	0	0	0	0	0	0
59	0	0	0	0	0	0	0	0	0	0	0	0	0
64	0	0	0	0	0	0	0	0	0	0	0	0	0
65	0	0	0	0	0	0	0	0	0	0	0	0	0
79	0	0	0	0	0	0	0	0	0	0	0	0	0
80	0	0	0	0	0	0	0	0	0	0	0	0	0
83	0	0	0	0	0	0	0	0	0	0	0	0	0
93	0	0	0	0	0	0	0	0	0	0	0	0	0
101	0	0	0	0	0	0	0	0	0	0	0	0	0

Gambar 6. Normalisasi Pada Matriks GLCM

c. Langkah selanjutnya menghitung nilai fitur pada glcm yakni *contrast*, *homogeneity*, *energy*, *correlation*. Berikut cara menghitung nilai fitur

- Contrast

$$\text{Contrast} = (|2-80|)^2 * (0.083333) + (|12-79|)^2 * (0.083333) + (|49-57|)^2 * (0.083333) + (|57-0|)^2 * (0.083333) + (|64-11|)^2 * (0.083333) + (|65-49|)^2 * (0.083333) + (|79-65|)^2 * (0.083333) + (|80-54|)^2 * (0.083333) + (|83-101|)^2 * (0.083333) + (|93-59|)^2 * (0.083333) + (|101-93|)^2 * (0.083333)$$

$$\text{Contrast} = 1587,5$$

- Homogeneity

$$\text{Hom} = (0,8333/1+|2-80|) + (0,8333/1+|12-79|) + (0,8333/1+|49-57|) + (0,8333/1+|54-0|) + (0,8333/1+|57-53|) + (0,8333/1+|64-11|) + (0,8333/1+|65-49|) + (0,8333/1+|79-64|) + (0,8333/1+|80-54|) + (0,8333/1+|83-101|) + (0,8333/1+|93-59|) + (0,8333/1+|101-93|)$$

$$\text{Hom} = 0.06048752$$

- Energy

$$\text{Energy} = \sqrt{\frac{(0,8333)^2 + (0,8333)^2 + (0,8333)^2 + (0,8333)^2 + (0,8333)^2 + (0,8333)^2 + (0,8333)^2 + (0,8333)^2 + (0,8333)^2 + (0,8333)^2 + (0,8333)^2 + (0,8333)^2}{(0,8333)^2 + (0,8333)^2}}$$

$$\text{Energy} = \sqrt{0.08333} = 0,28867$$

- Correlation

$$\mu_i = (2*0,8333) + (12*0,8333) + (49*0,8333) + (54*0,8333) + (57*0,8333) + (64*0,8333) + (65*0,8333) + (79*0,8333) + (80*0,8333) + (83*0,8333) + (93*0,8333) + (101*0,8333)$$

$$\mu_i = 61,583$$

$$\mu_j = (0*0,8333) + (11*0,8333) + (49*0,8333) + (53*0,8333) + (54*0,8333) + (57*0,8333) + (59*0,8333) + (64*0,8333) + (79*0,8333) + (80*0,8333) + (93*0,8333) + (101*0,8333)$$

$$\mu_j = 58,333$$

$$\sigma_i = \sqrt{\frac{(0,8333 * (2 - 61,583))^2 + (0,8333 * (11 - 61,583))^2 + (0,8333 * (49 - 61,583))^2 + (0,8333 * (54 - 61,583))^2 + (0,8333 * (57 - 61,583))^2 + (0,8333 * (64 - 61,583))^2 + (0,8333 * (65 - 61,583))^2 + (0,8333 * (79 - 61,583))^2 + (0,8333 * (80 - 61,583))^2 + (0,8333 * (83 - 61,583))^2 + (0,8333 * (93 - 61,583))^2 + (0,8333 * (101 - 61,583))^2}{12}}$$

$$\sigma_i = \sqrt{833,7569} = 28,875$$

$$\sigma_j = \sqrt{\frac{(0,8333 * (54 - 61,583))^2 + (0,8333 * (11 - 61,583))^2 + (0,8333 * (49 - 61,583))^2 + (0,8333 * (53 - 61,583))^2 + (0,8333 * (54 - 61,583))^2 + (0,8333 * (57 - 61,583))^2 + (0,8333 * (59 - 61,583))^2 + (0,8333 * (64 - 61,583))^2 + (0,8333 * (79 - 61,583))^2 + (0,8333 * (80 - 61,583))^2 + (0,8333 * (93 - 61,583))^2 + (0,8333 * (101 - 61,583))^2}{12}}$$

$$\sigma_j = \sqrt{810,89} = \sigma_j = 28,476$$

$$\text{Corelation} = \frac{(54-61,583)*(0-58,33)*(0,8333)/(28,875*28,476) + (64-61,583)*(11-58,33)*(0,8333)/(28,875*28,476) + (65-61,583)*(49-58,33)*(0,8333)/(28,875*28,476) + (57-61,583)*(53-58,33)*(0,8333)/(28,875*28,476) + (80-61,583)*(54-58,33)*(0,8333)/(28,875*28,476) + (49-61,583)*(57-58,33)*(0,8333)/(28,875*28,476) + (93-61,583)*(59-58,33)*(0,8333)/(28,875*28,476) + (79-61,583)*(64-58,33)*(0,8333)/(28,875*28,476) + (12-61,583)*(79-58,33)*(0,8333)/(28,875*28,476) + (2-61,583)*(80-58,33)*(0,8333)/(28,875*28,476) + (101-61,583)*(93-58,33)*(0,8333)/(28,875*28,476) + (83-61,583)*(101-58,33)*(0,8333)/(28,875*28,476)}{12}$$

Correlation = 0,0325

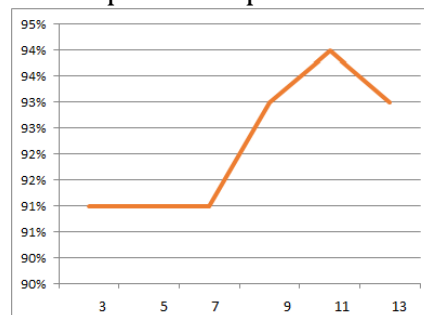
```
In [1]: runfile('D:/New folder/glcmlperfile.py', wdir='D:/New folder')
[1589.9166666666665, 0.00870105560404706, 0.2886751345948129, 0.03480213600126424]

In [2]:
```

Gambar 7. Hasil Ekstraksi Fitur Matriks 4x4 Menggunakan *Python*

2) Klasifikasi *K-Nearest Neighbor*

Data yang telah diekstraksi kemudian terlebih dahulu normalisasi menggunakan *min max scaling* dengan tujuan untuk agar data yang akan digunakan tidak memiliki rentan jarak yang berbeda jauh selanjutnya melakukan pembuatan model yang digunakan ialah berdasarkan jarak ketetanggaan terdekat pada data *training* dengan menggunakan salah satu jarak pada KNN yakni jarak *euclidean*. Proses pembuatan model ialah dengan memanggil fungsi *kneighborsclassifier* setelah itu menentukan nilai *k*. Nilai *k* yang digunakan ialah nilai yang ganjil dan nilai *k* yang berbeda yakni *K=3,5,7,9,11,13* dengan tujuan dapat mengetahui kinerja model pada setiap nilai *k*.



Gambar 8. Proses Percobaan Pada Beberapa Nilai *K*

Dari hasil grafik akurasi yang ditampilkan bahwa semakin tinggi nilai *K* maka semakin bagus proses klasifikasi namun pada *k=13* mengalami penurunan maka nilai akurasi tertinggi terdapat pada *k=11*. Berikut dibawah ini hasil confusion matrix pada *k=11*

Tabel 4. Hasil Confusion Matrix

Class Aktual	Class Prediksi	
	Bercak Daun	Karat Daun
Bercak Daun	39	1
Karat Daun	4	36

Dari tabel 4 *confusion matriks* hasil pengujian selanjutnya di lakukan evaluasi terhadap kinerja pada model dengan mengukur akurasi, *precision*, *recall* dan *f1 score*. Berikut dibawah ini hasil pengujian model untuk *k=11*

```
Test set accuracy: 0.9375
confusion matrix
[[39  1]
 [ 4 36]]
precision recall f1-score support
    BD      0.91    0.97    0.94      40
    KD      0.97    0.90    0.94      40
 accuracy          0.94          0.94      80
 macro avg          0.94          0.94      80
```

Gambar 9. Hasil Kinerja Model Untuk *K=11*

D. Simpulan

Berdasarkan hasil yang telah dilakukan dapat disimpulkan dari beberapa hal dari penelitian ini yakni metode *k-nearest neighbor* dalam melakukan klasifikasi

pada penyakit tanaman kopi arabika dengan nilai k yang terbaik ialah $k = 11$ menggunakan jarak euclidean dengan tingkat akurasi dalam penelitian ini sebesar 94 %.

E. Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Program Studi Teknik Informatika Universitas Kristen Indonesia Toraja yang telah memberi dukungan serta fasilitas untuk kelancaran penelitian ini.

F. Referensi

- [1] B. Prastowo, E. Karmawati, Rubijo, Siswanto, C. Indrawanto Dan S. J. Munarso "Budidaya Dan Pasca Panen Kopi".
- [2] R. Harni, Samsudin, G. Indriati, F. Soesanthi, Khaerati, E. Taufiq, A. M. Hasibuan, Dan A. D. Hapsari, "Teknologi Pengendalian Hama Dan Penyakit Tanaman Kopi".
- [3] F. Shofrotun, T. Sutojo, D. R. Ignatius, And M. Setiadi, "Identifikasi Tumbuhan Obat Herbal Berdasarkan Citra Daun Menggunakan Algoritma Gray Level Co-Occurrence Matrix Dan K-Nearest Neighbor," Vol. 6, No. November 2017, Pp. 51–56, 2018, Doi: 10.14710/Jtsiskom.6.2.2018.51-56.
- [4] M. S. Mustafa And I. W. Simpen, "Implementasi Algoritma K-Nearest Neighbor (KNN) Untuk Memprediksi Pasien Terkena Penyakit Diabetes Pada Puskesmas Manyampa Kabupaten Bulukumba," Vol. Viii, No. 1, Pp. 1–10.
- [5] N. N. Dzikrulloh And B. D. Setiawan, "Penerapan Metode K – Nearest Neighbor (KNN) Dan Metode Weighted Product (WP) Dalam Penerimaan Calon Guru Dan Karyawan Tata Usaha Baru Berwawasan Teknologi (Studi Kasus : Sekolah Menengah Kejuruan Muhammadiyah 2 Kediri)," Vol. 1, No. 5, Pp. 378–385, 2017.
- [6] A. Purnamawati, W. Nugroho, D. Putri, And W. F. Hidayat, "Deteksi Penyakit Daun Pada Tanaman Padi Menggunakan Algoritma Decision Tree, Random Forest, Naïve Bayes, Svmdan Knn," *Infotekjar J. Nas. Inform. Dan Teknol. Jar.*, Vol. 5, No. 1, Pp. 212–215, 2020, [Online].Available: <https://doi.org/10.30743/infotekjar.V5i1.2934>
- [7] J. Kusanti And N. A. Haris, "Klasifikasi Penyakit Daun Padi Berdasarkan Hasil Ekstraksi Fitur Glcm Interval 4 Sudut," Vol. 03, No. 01, Pp. 1–6, 2018.
- [8] Y. F. Achmad, A. Yulfitri, "Peran Enterprise Architecture & Big Data Dalam Perkembangan Industri Digital," 2021.
- [9] H. Wijayanto, "Klasifikasi Batik Menggunakan Metode K-Nearest Neighbour Berdasarkan Gray Level Co-Occurrence Matrices (GLCM)".
- [10] H. Shim *Et Al.*, "Implementasi Deep Learning Untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Wayang Golek," *Adv. Opt. Mater.*, Vol. 10, No. 1, Pp. 1–9, 2018, [Online].Available: <https://doi.org/10.1103/Physrevb.101.089902%0a>