

BAB 3

METODE PENELITIAN

3.1. Subjek dan Obyek Penelitian

Subjek penelitian merupakan data yang diamati. Subjek penelitian ini merupakan dataset jeruk sehat dan jeruk busuk bersumber dari platform kaggle pada tautan berikut <https://s.id/DatasetJeruk>, data tersebut dibuat oleh Sriram Reddy Kalluri, data tersebut merupakan data umum dan bukan data untuk penelitian ilmiah. Objek penelitian adalah permasalahan yang akan diteliti. Objek penelitian ini adalah ALGORITMA *DISTANCE WEIGHTED K-NEAREST NEIGHBOR* PADA KLASIFIKASI KUALITAS JERUK MENGGUNAKAN EKSTRAKSI FITUR *GRAY LEVEL CO OCCURRENCE MATRIX*

3.2. Alat dan Bahan Penelitian

3.2.1. Alat

A. Perangkat Keras

Perangkat keras yang digunakan untuk membuat penelitian ini :

1. *Processor : Intel core i5-8350*
2. *Memory : 8 GB RAM DDR 4*
3. *Graphic Card : Intel UHD graphics 620*
4. *Operating System : Windows 11 64 bit*
5. *SSD : 256 GB*

B. Perangkat Lunak

Perangkat lunak yang digunakan untuk membuat penelitian ini :

1. *Google Chrome* digunakan untuk menjelajah internet.
2. *Google Colab* digunakan untuk membangun model

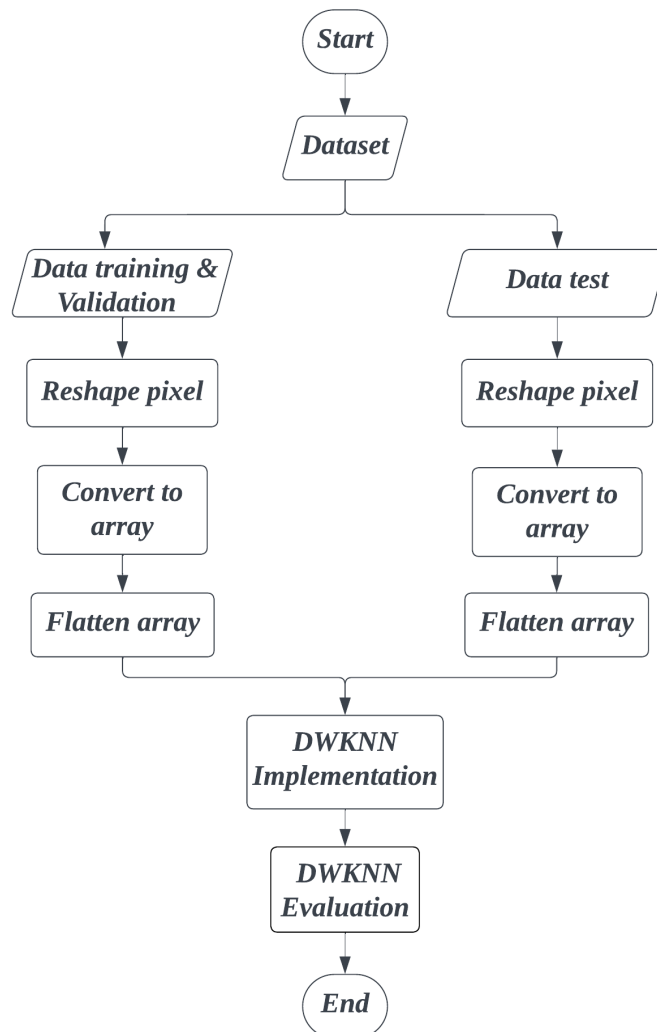
3.2.2. Bahan Penelitian

Dataset dalam penelitian ini bersumber pada platform kaggle pada tautan <https://s.id/DatasetJeruk>. *Dataset* tersebut menghasilkan lebih dari 3800 yang sudah dilabeli menjadi *fresh oranges* dan *rotten oranges*. *Dataset* tersebut akan dilakukan

pengolahan dan analisis sehingga menghasilkan model *machine learning* yang dapat memprediksi *fresh oranges* dan *rotten oranges*.

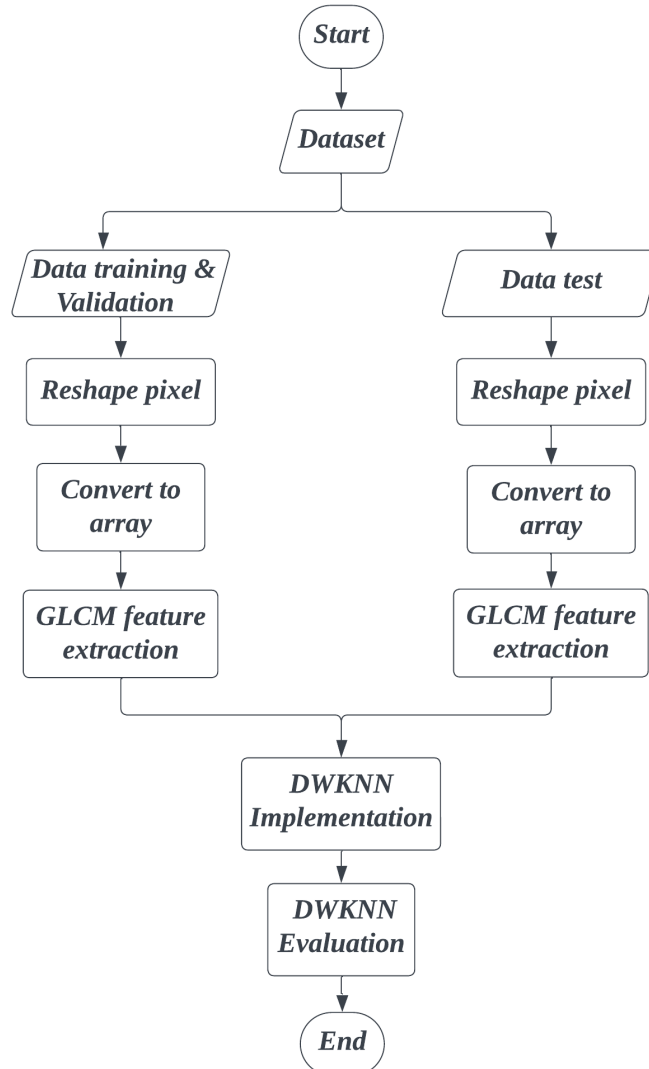
3.3. Diagram Alir Penelitian

Bagian ini menjelaskan mengenai tahapan yang dilakukan pada penelitian PENERAPAN ALGORITMA *DISTANCE WEIGTED K-NEAREST NEIGHBOR* PADA KLASIFIKASI KUALITAS JERUK MENGGUNAKAN EKSTRAKSI FITUR *GRAY LEVEL CO OCCURRENCE MATRIX*. Tahap ini disusun secara sistematis sehingga memudahkan peneliti untuk mencapai tujuan. Pada gambar 3.1. adalah diagram alir penelitian DWKNN tanpa menggunakan GLCM, dimulai dari dataset, konversi ke *array*, konversi ke *flatten array*, implementasi DWKNN dan evaluasi DWKNN.



Gambar 3. 1. Diagram alir penelitian DWKNN tanpa menggunakan GLCM

Lalu pada gambar 3.2. adalah diagram alir penelitian DWKNN menggunakan GLCM, dimulai dari dataset, konversi ke *array*, ekstraksi fitur GLCM, implementasi DWKNN dan evaluasi DWKNN.

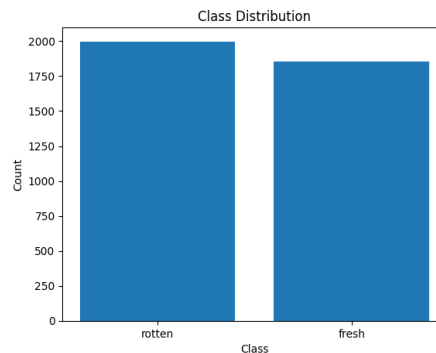


Gambar 3. 2. Diagram alir penelitian DWKNN menggunakan GLCM

3.4. Dataset

Dataset pada penelitian ini terdapat 2 kelas tertera pada gambar 3.3, yaitu *fresh oranges* dan *rotten oranges*, dengan total 3852 data dibagi menjadi 1998 data *rotten oranges* dan 1854 data *fresh oranges*. Berdasarkan data tersebut memiliki keseimbangan data yang baik, dikarenakan selisih data yang cenderung kecil. *Dataset* tersebut dibagi menjadi 3, yaitu data latih 70%, data validasi 20%, dan data tes 10%. Secara keseluruhan data tersebut memiliki selisih kelas dalam tahap wajar,

sehingga ketika melakukan pembagian menjadi data latih, data validasi dan data tes menghasilkan data dengan selisih kelas dalam cenderung kecil.



Gambar 3. 3. Distribusi dataset

3.5. Reshape pixel

Sebelum mengubah gambar kedalam bentuk *array*, penelitian ini mengubah seluruh ukuran data gambar menjadi 224 x 224 piksel. Mengubah besaran gambar bertujuan untuk melakukan standarisasi. Sehingga, seluruh gambar akan diolah dengan format ukuran data yang sama dan menghasilkan nilai *array* dengan format yang sama.

3.6. Konversi ke array

Proses konversi citra dalam bentuk *array* diperlukan dalam proses pembuatan model, karena algoritma *machine learning* tidak dapat langsung memproses data dalam bentuk gambar dan hanya bisa mengolah dalam tipe data angka. Data gambar dalam pengolahan citra dikonversi kedalam *array*, penelitian ini mengkonversi *array* kedalam bentuk 2 dimensi, *array* 2 dimensi berfungsi untuk diolah kembali menggunakan ekstraksi fitur pada GLCM, konversi diimplementasikan pada seluruh data, yaitu data latih, data validasi dan data tes.

3.7. Flatten array dan Ekstraksi Fitur GLCM

Penelitian ini membandingkan performa klasifikasi DWKNN menggunakan *array* 1D dan GLCM. Tujuannya, adalah untuk membandingkan performa DWKNN menggunakan GLCM dan tanpa menggunakan GLCM. Model pertama pada gambar 3.1 menggunakan ekstraksi fitur *array* dan dilakukan *flatten* dan melakukan normalisasi untuk dilakukan klasifikasi secara langsung. Sehingga,

model pertama hanya memanfaatkan *array* untuk melakukan klasifikasi tanpa adanya ekstraksi fitur lain.

Model kedua pada gambar 3.2 menggunakan ekstraksi ciri GLCM. Pemilihan fitur yang ada pada GLCM adalah *contrast*, *correlation*, *dissimilarity*, *homogeneity*, *energy*, dan *ASM*. Ekstraksi fitur ini menggunakan 4 sudut yaitu 0, 45, 90, dan 135, sehingga menghasilkan 4 fitur baru berdasarkan masing masing fitur pada GLCM dengan total 24 fitur yang merepresentasikan masing - masing data gambar.

3.8. Penghitungan GLCM

Penelitian ini menggunakan sampel array pada tabel 3.1. untuk melakukan penghitungan GLCM, tujuannya untuk mengetahui penghitungan setiap fitur yang digunakan pada GLCM. *Array* 3x3 tersebut akan dikonversi menjadi 24 fitur dikarenakan penelitian ini menggunakan 6 fitur GLCM dan 4 derajat.

Tabel 3. 1. Sampel *array*

<i>Array</i>		
3	5	7
2	4	1
6	3	5

Array tersebut akan diubah menjadi *array* GLCM dengan derajat yang sudah ditentukan pada penelitian ini, yaitu 0°, 45°, 90° dan 135°. Sehingga, hasil *array* akan terkonversi menjadi 4 *Array* GLCM, dengan mengambil nilai terbesar pada tabel 3.1. mengidentifikasi tipe bilangan untuk acuan baris dan kolom. Sesuai dengan jumlah *array* yang ada, maka konversi tabel 3.1. akan menghasilkan 4 *array* GLCM, yang nantinya dijadikan sebagai acuan penghitungan masing – masing fitur.

Melalui penghitungan ekstraksi fitur GLCM pada Tabel 3.2 dengan orientasi 0°, hasilnya menggambarkan sebuah *array* yang terdiri dari 6 nilai. Proses ini diacu pada Gambar 2.3 dan melibatkan pencarian satu *array* tetangga terdekat, suatu aspek krusial dalam analisis tekstur. Hasil *array* ini mencerminkan informasi signifikan tentang pola visual pada gambar jeruk, yang selanjutnya dapat diinterpretasikan dalam konteks klasifikasi kualitas buah menggunakan metode *Distance Weighted K-Nearest Neighbors* (DWKNN).

Tabel 3. 2. Hasil *array* GLCM 0°

Array 8 x 8

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	1	0	0	0
3	0	0	0	0	0	2	0	0
4	0	1	0	0	0	0	0	0
5	0	0	0	0	0	0	0	1
6	0	0	0	1	0	0	0	0
7	0	0	0	0	0	0	0	0

Berdasarkan tabel 3.3. Berikut merupakan cara penghitungan ekstraksi fitur GLCM 45°. Mengacu pada gambar 2.4. akan menghasilkan *array* dengan total 4 nilai, dengan mencari 1 *array* tetangga terdekat.

Tabel 3. 3. Hasil tabel *array* GLCM 45°

Array 8 x 8

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	0	0
3	0	1	0	0	0	0	0	0
4	0	0	0	0	0	0	0	1
5	0	0	0	0	0	0	0	0
6	0	0	0	0	1	0	0	0
7	0	0	0	0	0	0	0	0

Berdasarkan tabel 3.4. Berikut merupakan cara penghitungan ekstraksi fitur GLCM 90°. Mengacu pada gambar 2.5. akan menghasilkan *array* dengan total 6 nilai, dengan mencari 1 *array* tetangga terdekat.

Tabel 3. 4. Hasil tabel *array* GLCM 90°

Array 8 x 8

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1
2	0	0	0	1	0	0	0	0
3	0	0	0	0	1	0	0	0
4	0	0	0	0	0	1	0	0
5	0	1	0	0	0	0	0	0
6	0	0	1	0	0	0	0	0
7	0	0	0	0	0	0	0	0

Berdasarkan tabel 3.5. Berikut merupakan cara penghitungan ekstraksi fitur GLCM. Berikut merupakan cara penghitungan ekstraksi fitur GLCM 135°. Mengacu pada gambar 2.6. akan menghasilkan *array* dengan total 4 nilai, dengan mencari 1 *array* tetangga terdekat.

Tabel 3. 5. Hasil tabel *array* GLCM 135°

<i>Array 8 x 8</i>								
	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0
2	0	0	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0
4	0	0	0	1	0	0	0	0
5	0	0	0	0	1	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0

Tabel 3.6 membahas mengenai penghitungan *contrast* pada ekstraksi fitur GLCM, penghitungan ini dilakukan dengan mengambil *array* pada tabel 3.1. lalu akan dikonversi menjadi fitur *contrast* menggunakan formula (2,7).

$$Contrast = \sum_{i,j=0}^{levels-1} P_{i,j} (i - j)^2$$

i = Nilai baris

j = Nilai kolom

P = Probabilitas kemunculan nilai *array* GLCM

Tabel 3. 6. Penghitungan *contrast*

<i>Contrast</i>			
Derajat	Lokasi	Rumus	Hasil
0	(2,4)	$(2-4)^2 * 1/6$	0,6666666667
	(3,5)	$(3-5)^2 * 2/6$	1,3333333333
	(4,1)	$(4-1)^2 * 1/6$	1,5
	(5,7)	$(5-7)^2 * 1/6$	0,6666666667
	(6,3)	$(6-3)^2 * 1/6$	1,5
Total			5,6666666667
45	(2,5)	$(2-5)^2 * 1/4$	2,25
	(3,1)	$(3-1)^2 * 1/4$	1
	(4,7)	$(4-7)^2 * 1/4$	2,25
	(6,4)	$(6-4)^2 * 1/4$	1
Total			6,5
90	(1,7)	$(1-7)^2 * 1/6$	6
	(2,3)	$(2-3)^2 * 1/6$	0,1666666667
	(3,4)	$(3-4)^2 * 1/6$	0,1666666667
	(4,5)	$(4-5)^2 * 1/6$	0,1666666667

<i>Contrast</i>			
Derajat	Lokasi	Rumus	Hasil
	(5,1)	$(5-1)^2 \cdot 1/6$	2,666666667
	(6,2)	$(6-2)^2 \cdot 1/6$	2,666666667
Total			11,83333333
135	(1,5)	$(1-5)^2 \cdot 1/4$	4
	(3,2)	$(3-2)^2 \cdot 1/4$	0,25
	(4,3)	$(4-3)^2 \cdot 1/4$	0,25
	(5,4)	$(5-4)^2 \cdot 1/4$	0,25
Total			4,75

Tabel 3.7. membahas mengenai penghitungan *homogeneity* pada ekstraksi fitur GLCM, penghitungan ini dilakukan dengan mengambil *array* pada tabel 3.1.

$$Homogeneity = \sum_{i,j=0}^{levels-1} \frac{P_{i,j}}{1+(i-j)^2}$$

i = Nilai baris

j = Nilai kolom

P = Probabilitas kemunculan nilai *array* GLCM

Tabel 3. 7. Penghitungan *homogeneity*

<i>Homogeneity</i>			
Derajat	Lokasi	Rumus	Hasil
0	(2,4)	$(1/((1+(2-4)^2)))/6$	0,033333333333
	(3,5)	$(2/(1+(3-5)^2))/6$	0,066666666667
	(4,1)	$(1/(1+(4-1)^2))/6$	0,016666666667
	(5,7)	$(1/(1+(5-7)^2))/6$	0,033333333333
	(6,3)	$(1/(1+(6-3)^2))/6$	0,016666666667
Total			0,1666666667
45	(2,5)	$(1/((1+(2-5)^2)))/4$	0,025
	(3,1)	$(1/((1+(3-1)^2)))/4$	0,05
	(4,7)	$(1/((1+(4-7)^2)))/4$	0,025
	(6,4)	$(1/((1+(6-4)^2)))/4$	0,05
Total			0,15
90	(1,7)	$(1/((1+(1-7)^2)))/6$	0,004504504505
	(2,3)	$(1/(1+(2-3)^2))/6$	0,083333333333
	(3,4)	$(1/(1+(3-4)^2))/6$	0,083333333333
	(4,5)	$(1/(1+(4-5)^2))/6$	0,083333333333
	(5,1)	$(1/(1+(5-1)^2))/6$	0,009803921569
	(6,2)	$(1/(1+(6-2)^2))/6$	0,009803921569
Total			0,2741123476
135	(1,5)	$(1/((1+(1-5)^2)))/4$	0,01470588235
	(3,2)	$(1/((1+(3-2)^2)))/4$	0,125
	(4,3)	$(1/((1+(4-3)^2)))/4$	0,125
	(5,4)	$(1/((1+(5-4)^2)))/4$	0,125
Total			0,3897058824

Tabel 3.8. membahas mengenai penghitungan *dissimilarity* pada ekstraksi fitur GLCM, penghitungan ini dilakukan dengan mengambil *array* pada tabel 3.1. lalu akan dikonversi menjadi fitur *dissimilarity* menggunakan formula (2,9).

$$Dissimilarity = \sum_{i,j=0}^{levels-1} P_{i,j} |i - j|$$

i = Nilai baris

j = Nilai kolom

P = Probabilitas kemunculan nilai *array* GLCM

Tabel 3. 8. Penghitungan *dissimilarity*

<i>Dissimilarity</i>			
Derajat	Lokasi	Rumus	Hasil
0	(2,4)	abs(2-4)*1/6	0,3333333333
	(3,5)	abs(3-5)*2/6	0,6666666667
	(4,1)	abs(4-1)*1/6	0,5
	(5,7)	abs(5-7)*1/6	0,3333333333
	(6,3)	abs(6-3)*1/6	0,5
Total			2,333333333
45	(2,5)	abs(2-5)*1/4	0,75
	(3,1)	abs(3-1)*1/4	0,5
	(4,7)	abs(4-7)*1/4	0,75
	(6,4)	abs(6-4)*1/4	0,5
Total			2,5
90	(1,7)	abs(1-7)*1/6	1
	(2,3)	abs(2-3)*1/6	0,1666666667
	(3,4)	abs(3-4)*1/6	0,1666666667
	(4,5)	abs(4-5)*1/6	0,1666666667
	(5,1)	abs(5-1)*1/6	0,6666666667
	(6,2)	abs(6-2)*1/6	0,6666666667
Total			2,833333333
135	(1,5)	abs(1-5)*1/4	1
	(3,2)	abs(3-2)*1/4	0,25
	(3,2)	abs(3-2)*1/4	0,25
	(5,4)	abs(5-4)*1/4	0,25
Total			1,75

Tabel 3.9. membahas mengenai penghitungan ASM pada ekstraksi fitur GLCM, penghitungan ini dilakukan dengan mengambil *array* pada tabel 3.1. lalu akan dikonversi menjadi fitur *ASM* menggunakan formula (2,6).

$$ASM = \sum_{i,j=0}^{levels-1} P_{i,j}^2$$

i = Nilai baris

j = Nilai kolom

P = Probabilitas kemunculan nilai *array* GLCM

Tabel 3. 9. Penghitungan ASM

ASM		
Derajat	Rumus	Hasil
0	$(1^2/6 + 2^2/6 + 1^2/6 + 1^2/6 + 1^2/6)/6$	0,2222222222
45	$(1^2/4 + 1^2/4 + 1^2/4 + 1^2/4)/4$	0,25
90	$(1^2/6 + 1^2/6 + 1^2/6 + 1^2/6 + 1^2/6 + 1^2/6)/6$	0,1666666667
135	$(1^2/4 + 1^2/4 + 1^2/4 + 1^2/4)/4$	0,25

Tabel 3.10. membahas mengenai penghitungan *Energy* pada ekstraksi fitur GLCM, penghitungan ini dilakukan dengan mengambil *array* pada tabel 3.1. lalu akan dikonversi menjadi fitur *Energy* menggunakan formula (2,10).

$$Energy = \sqrt{\sum_{i,j=0}^{levels-1} P_{i,j}^2}$$

i = Nilai baris

j = Nilai kolom

P = Probabilitas kemunculan nilai *array* GLCM

Tabel 3. 10. Penghitungan Energy

Energy		
Derajat	Rumus	Hasil
0	$\text{sqrt}((1^2/6 + 2^2/6 + 1^2/6 + 1^2/6 + 1^2/6)/6)$	0,4714045208
45	$\text{sqrt}((1^2/4 + 1^2/4 + 1^2/4 + 1^2/4)/4)$	0,5
90	$\text{sqrt}((1^2/6 + 1^2/6 + 1^2/6 + 1^2/6 + 1^2/6 + 1^2/6)/6)$	0,4082482905
135	$\text{sqrt}((1^2/4 + 1^2/4 + 1^2/4 + 1^2/4)/4)$	0,5

Tabel 3.11. membahas mengenai penghitungan *Correlation* pada ekstraksi fitur GLCM, penghitungan ini dilakukan dengan mengambil *array* pada tabel 3.1. lalu akan dikonversi menjadi fitur *Correlation* menggunakan formula (2,7). Berdasarkan fitur yang ada pada GLCM, fitur *correlation* memiliki penghitungan yang paling panjang daripada fitur lainnya.

$$Correlation = \sum_{i,j=0}^{levels-1} P_{i,j} \left[\frac{(i - \mu_i)(i - \mu_j)}{\sqrt{(\sigma_i^2)(\sigma_j^2)}} \right]$$

i = Nilai baris

j = Nilai kolom

P = Probabilitas kemunculan nilai *array* GLCM

μ_i = Rata – rata penghitungan baris

μ_j = Rata – rata penghitungan kolom

σ_i = Penghitungan rata rata baris dikurangi kolom

σ_j = Penghitungan rata rata kolom dikurangi baris

Tabel 3. 11. Penghitungan Correlation

Correlation 0°								
Rata - rata			Rumus			Hasil		
i			$(2*1 + 3*2 + 4*1 + 5*1 + 6*1) / 6$			3,833333333		
j			$(1*1 + 3*1 + 4*1 + 5*2 + 7*1) / 6$			4,166666667		
Penghitungan i			Penghitungan j			Penghitungan korelasi		
Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil
sig i (2,4)	$(2-3,83)^2 * 1/6$	0,56018 51852	sig j (4,1)	$(1-4,16)^2 * 1/6$	1,67129 6296	corr (2,4)	$((2-3,83) * (4-4,16) * 1/6) / (1,34 * 1,86)$	0,02033 900719
sig i (3,5)	$(3-3,83)^2 * 2/6$	0,23148 14815	sig j (6,3)	$(3-4,16)^2 * 1/6$	0,22685 18519	corr (3,5)	$((3-3,83) * (5-4,16) * 2/6) / (1,34 * 1,86)$	- 0,09245 00327
sig i (4,1)	$(4-3,83)^2 * 1/6$	0,00462 962963	sig j (2,4)	$(4-4,14)^2 * 1/6$	0,00462 962963	corr (4,1)	$((4-3,83) * (1-4,16) * 1/6) / (1,34 * 1,86)$	- 0,03513 101243
sig i (5,7)	$(5-3,83)^2 * 1/6$	0,22685 18519	sig j (3,5)	$(5-4,14)^2 * 2/6$	0,23148 14815	corr (5,7)	$((5-3,83) * (7-4,16) * 1/6) / (1,34 * 1,86)$	0,22003 10778
sig i (6,3)	$(6-3,83)^2 * 1/6$	0,78240 74074	sig j (5,7)	$(7-4,14)^2 * 1/6$	1,33796 2963	corr (6,3)	$((6-3,83) * (3-4,16) * 1/6) / (1,34 * 1,86)$	- 0,16825 90595
sqrt sig i		1,34370 9625	sqrt sig j		1,86338 9981	Corr 0		- 0,05547 001962
Correlation 45°								
Rata - rata			Rumus			Hasil		
i			$(2*1+3*1+4*1+6*1)/4$			3,75		

Correlation 45°								
Rata - rata			Rumus			Hasil		
j			$(1*1+4*1+5*1+7*1)/4$			4,25		
Penghitungan i			Penghitungan j			Penghitungan korelasi		
Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil
sig i (2,5)	$(2-3,75)^2 * 1/4$	0,76562 5	sig j (2,5)	$(5-4,25)^2 * 1/4$	0,14062 5	corr (2,5)	$((2-3,75)*(5-4,25)*1/4)/(1,4*7*2,16)$	- 0,10246 95077
sig i (3,1)	$(3-3,75)^2 * 1/4$	0,14062 5	sig j (3,1)	$(1-4,25)^2 * 1/4$	2,64062 5	corr (3,1)	$((3-3,75)*(1-4,25)*1/4)/(1,4*7*2,16)$	0,19030 05142
sig i (4,7)	$(4-3,75)^2 * 1/4$	0,01562 5	sig j (4,7)	$(7-4,25)^2 * 1/4$	1,89062 5	corr (4,7)	$((4-3,75)*(7-4,25)*1/4)/(1,4*7*2,16)$	0,05367 450401
sig i (6,4)	$(6-3,75)^2 * 1/4$	1,26562 5	sig j (6,4)	$(4-4,25)^2 * 1/4$	0,01562 5	corr (6,4)	$((6-3,75)*(4-4,25)*1/4)/(1,4*7*2,16)$	- 0,04391 550328
sqrt sig i		1,47901 9946	sqrt sig j		2,16506 3509	Corr 45		0,09759 000729
Correlation 90°								
Rata - rata			Rumus			Hasil		
I			$(1*1+2*1+3*1+4*1+5*1+6*1)/6$			3,5		
J			$(1*1+2*1+3*1+4*1+5*1+7*1)/6$			3,666666667		
Penghitungan i			Penghitungan j			Penghitungan korelasi		
Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil
sig i (1,7)	$(1-3,5)^2 * 1/6$	1,04166 6667	sig j (1,7)	$(7-3,66)^2 * 1/6$	1,85185 1852	corr (1,7)	$((1-3,5)*(7-3,66)*1/6)/(1,7*0*1,97)$	- 0,41239 30494
sig i (2,3)	$(2-3,5)^2 * 1/6$	0,375	sig j (2,3)	$(3-3,66)^2 * 1/6$	0,07407 407407	corr (2,3)	$((2-3,5)*(3-3,66)*1/6)/(1,7*0*1,97)$	0,04948 716593
sig i (3,4)	$(3-3,5)^2 * 1/6$	0,04166 666667	sig j (3,4)	$(4-3,66)^2 * 1/6$	0,01851 851852	corr (3,4)	$((3-3,5)*(4-3,66)*1/6)/(1,7*0*1,97)$	- 0,00824 786098
sig i (4,5)	$(4-3,5)^2 * 1/6$	0,04166 666667	sig j (4,5)	$(5-3,66)^2 * 1/6$	0,29629 62963	corr (4,5)	$((4-3,5)*(5-3,66)*1/6)/(1,7*0*1,97)$	0,03299 144395
sig i (5,1)	$(5-3,5)^2 * 1/6$	0,375	sig j (5,1)	$(1-3,66)^2 * 1/6$	1,18518 5185	corr (5,1)	$((5-3,5)*(1-3,66)*1/6)/(1,7*0*1,97)$	- 0,19794 86637
sig i (6,2)	$(6-3,5)^2 * 1/6$	1,04166 6667	sig j (6,2)	$(2-3,66)^2 * 1/6$	0,46296 2963	corr (6,2)	$((6-3,5)*(2-3,66)*1/6)/(1,7*0*1,97)$	- 0,20619 65247

Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil
sqrt sig i		1,70782 5128	sqrt sig j		1,97202 6594	Corr 90		- 0,74230 7489
Correlation 135°								
Rata - rata			Rumus			Hasil		
I			$(1*1+3*1+4*1+5*1)/4$			3,25		
j			$(2*1+3*1+4*1+5*1)/4$			3,5		
Penghitungan i			Penghitungan j			Penghitungan korelasi		
Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil	Lok asi	Rumus	Hasil
sig i (1,5)	$(1-3,25)^2 * 1/4$	1,26562 5	sig j (1,5)	$(5-3,5)^2 * 1/4$	0,5625	corr (1,5)	$((1-3,25)*(5-3,5)^{1/4}) / (1,47 * 1,11)$	- 0,51025 20386
sig i (3,2)	$(3-3,25)^2 * 1/4$	0,01562 5	sig j (3,2)	$(2-3,5)^2 * 1/4$	0,5625	corr (3,2)	$((3-3,25)*(2-3,5)^{1/4}) / (1,47 * 1,11)$	0,05669 467095
sig i (4,3)	$(4-3,25)^2 * 1/4$	0,14062 5	sig j (4,3)	$(4-3,5)^2 * 1/4$	0,0625	corr (4,3)	$((4-3,25)*(3-3,5)^{1/4}) / (1,47 * 1,11)$	- 0,05669 467095
sig i (5,4)	$(5-3,25)^2 * 1/4$	0,76562 5	sig j (5,4)	$(4-3,5)^2 * 1/4$	0,0625	corr (5,4)	$((5-3,25)*(4-3,5)^{1/4}) / (1,47 * 1,11)$	0,13228 75656
sqrt sig i		1,47901 9946	sqrt sig j		1,11803 3989	Corr 135		- 0,37796 4473

Tabel 3.12. merupakan hasil rekap dari seluruh ekstraksi fitur GLCM yang diterapkan pada penelitian ini. Berdasarkan *array* pada tabel 4.1, maka seluruh fitur yang ada akan dikonversi menjadi dimensi yang lebih pendek dengan mempertahankan representatif data yang ada. Sehingga, berpengaruh kepada peningkatan performa training dengan penghitungan yang jauh lebih sedikit.

Tabel 3. 12. Hasil GLCM

Deg ree	Feature					
	Contrast	Homogeneity	Dissimilarity	ASM	Energy	Correlation
0	5,66666 6667	0,1666666667	2,333333333	0,2222222 222	0,4714045 208	-0,05547001962
45	6,5	0,15	2,5	0,25	0,5	0,09759000729
90	11,8333 3333	0,2741123476	2,833333333	0,1666666 667	0,4082482 905	-0,742307489
135	4,75	0,3897058824	1,75	0,25	0,5	-0,377964473

3.9. Evaluasi DWKNN

Sistem klasifikasi penelitian ini adalah *binary classification*, pada table 3.13 dimana kelas yang akan diprediksi dan diujikan adalah *fresh* dan *rotten*. Evaluasi model dalam penelitian ini menggunakan *confusion matrix*. Evaluasi *confusion matrix* dapat menghasilkan akurasi model, presisi, *recall*, dan *f1-score* dari kelas *fresh oranges* dan *rotten oranges*. Sehingga, *confusion matrix* berfungsi untuk mengetahui performa secara lengkap dan mendetail dari seluruh model yang ada.

Selain menggunakan *confusion matrix*, penelitian ini membandingkan akurasi latih dan akurasi validasi, untuk mengidentifikasi performa *overfitting*, *underfitting*, dan *good fit*. Identifikasi performa dapat dicapai melalui formula standar deviasi (2,11). Semakin jauh hasil jarak antara akurasi latih dan valid maka model dapat dinyatakan memiliki performa *overfitting/underfitting*. Sebaliknya, ketika hasil akurasi latih dan valid mendekati 0, maka model dapat dinyatakan memiliki performa *good fit*.

Tabel 3. 13. Confussion matrix penelitian

<i>Confusion Matrix</i>		
<i>Actual Values</i>	<i>Predicted Values</i>	
	<i>Fresh</i>	<i>Rotten</i>
<i>Fresh</i>	<i>True Fresh</i>	<i>False Fresh</i>
<i>Rotten</i>	<i>False Rotten</i>	<i>True Rotten</i>