

Klasifikasi Jenis-Jenis Buah Nanas Menggunakan Learning Vector Quantization (LVQ)

K Azhari¹, Supatman²

^{1,2}Jurusan Teknik Informatika, Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta

E-mail: khililazhari98@gmail.com¹, supatman@mercubuana-yogya.ac.id²

Abstrak. Nanas adalah sejenis tumbuhan tropis yang berasal dari Brasil, Bolivia, dan Paraguay Tumbuhan ini termasuk dalam famili nanas. Perawakan tumbuhannya rendah, herba dengan 30 atau lebih daun yang panjang, berujung tajam, tersusun dalam bentuk roset mengelilingi batang yang tebal. Nanas juga salah satu buah paling populer di Dunia maupun di Indonesia karena memiliki nutrisi yang sangat baik bagi manusia. Nanas juga memiliki banyak jenis, nama lain dari nanas adalah nenas, dan ananas. Banyak masyarakat yang tidak bisa membedakan jenis nanas jika di lihat dari jenis tekstur kulit dan warna nanas. Penelitian ini menggunakan metode Learning Vector Quantization untuk klasifikasi jenis-jenis nanas, dengan mengumpulkan data pelatihan sebanyak 120 data gambar buah nanas, yang terdiri dari 30 gambar buah nanas batu, 30 gambar buah nanas honi dan 30 gambar nanas madu, untuk data pengujian menggunakan 30 data gambar terdiri dari 10 gambar buah nanas batu, 10 gambar buah nanas honi dan 10 gambar buah nanas madu. Didapatkan hasil akurasi buah nanas terbaik menggunakan parameter dengan nilai learning rate = 0.01, 0,005 pada iterasi ke-100, 500, dan 1000 dengan hasil citra benar 80 salah 10 dengan persentase kumulatif yaitu 88.8889%. terjadi pada data pelatihan, sedangkan pada data uji menghasilkan benar 25 salah 5 dengan persentase kumulatif yaitu 83.3333%.

Kata kunci: *Jenis-jenis nanas, Jaringan Saraf Tiruan, Learning Vector Quantization.*

Abstract. Pineapple is a type of tropical plant native to Brazil, Bolivia, and Paraguay This plant belongs to the family of pineapples. The stature of the plant is low, herbaceous with 30 or more long leaves, sharp-tipped, arranged in the form of rosettes around a thick stem. Pineapple is also one of the most popular fruits in the world as well as in Indonesia because it has excellent nutrition for humans. Pineapple also has many types, other names of pineapple are pineapple, and ananas. Many people cannot distinguish the type of pineapple if seen from the type of skin texture and color of pineapple. This study used the Learning Vector Quantization method to classify the types of pineapples, by collecting training data as much as 120 images of pineapples, consisting of 30 pictures of stone pineapples, 30 images of honi pineapples and 30 images of honey pineapples, for testing data using 30 image data consisting of 10 pictures of stone pineapples, 10 images of honi pineapples and 10 images of honey pineapples. Obtained the best pineapple accuracy results using parameters with a learning rate = 0.01, 0.005 in the 100th, 500th, and 1000th iterations with a correct image yield of 80 one 10 with a cumulative percentage of 88.8889%. occurs in the training data, while in the test data produces 25 wrong 5 with a cumulative percentage of 83.3333%.

Keywords: Types of pineapples, Artificial Neural Networks, Learning Vector Quantization.

1. Pendahuluan

Indonesia adalah negara tropis yang kaya akan berbagai buah-buahan, iklim tropis di Indonesia adalah keuntungan alami, sehingga dapat dibuat sebagai salah satu produsen buah tropis terbesar di dunia. Salah satu komoditas buah tropis penting di Indonesia yang menjadi produk andalan di Tanah Air adalah nanas. Indonesia memiliki berbagai macam kultivar nanas yang telah dibudidayakan oleh petani mulai dari Sumatera hingga Irian Jaya [1].

Nanas (*Ananas comosus* L. Merr) adalah tanaman buah yang berasal dari Amerika tropis yaitu Brasil, Argentina dan Peru. Tanaman nanas telah menyebar ke seluruh dunia, terutama di sekitar wilayah khatulistiwa yang antara 25 °LU dan 25 °LS. Di Indonesia tanaman nanas sangat terkenal dan banyak dibudidayakan di tegalan dari dataran rendah ke dataran tinggi. Daerah penghasil nanas yang terkenal di Indonesia adalah Subang, Bogor, Riau, Palembang, dan Blitar[2].

Jaringan syaraf tiruan merupakan model adaptif yang dapat mengubah strukturnya untuk memecahkan masalah berdasarkan informasi eksternal maupun internal yang mengalir melalui model tersebut[3].

Klasifikasi buah adalah salah satu studi yang paling kompleks. Ini karena jenis buah, nanas misalnya bisa berbeda dengan nanas lain dalam hal ukuran, warna, tekstur, bentuk dan lain-lain, selain kesamaan bentuk, ukuran dan warna antara jenis buah juga menambah kompleksitas dalam proses klasifikasi buah[4]. Buah sebagai objek yang akan diklasifikasikan, memiliki banyak ciri yang dapat digunakan sebagai fitur. Salah satunya adalah fitur warna. Fitur warna dapat dikatakan sebagai ciri dominan dari jenis buah yang dapat membedakan satu jenis buah dari yang lain.

Penggunaan metode Learning Vector Quantization dalam klasifikasi spesies nanas karena metode ini dapat melakukan pembelajaran di lapisan kompetitif yang dipantau sehingga lapisan secara otomatis belajar melakukan klasifikasi vektor input. Metode ini digunakan dengan baik untuk mengklasifikasikan objek. Penggunaan metode ini diharapkan dapat membantu membedakan jenis buah nanas[5].

2. Kajian Pustaka

2.1. Jaringan Saraf Tiruan (*Artificial Neural Network*)

Jaringan Syaraf Tiruan (JST) adalah sistem pemroses informasi yang memiliki karakteristik unjuk kerja tertentu yang menyerupai jaringan syaraf biologis[6].

2.2. Definisi Citra

Citra atau gambar dapat didefinisikan sebagai sebuah fungsi dua dimensi, $f(x,y)$, dimana x dan y adalah koordinat bidang datar, dan harga fungsi f di setiap pasangan koordinat (x,y) disebut intensitas atau level keabuan (grey level) dari gambar di titik itu [7].

2.3. Citra Digital

Citra digital adalah salah satu teknologi yang dikembangkan untuk mendapatkan informasi dari citra dengan cara memodifikasi bagian dari citra yang diperlukan sehingga menghasilkan citra lain yang lebih informatif[8].

2.4. Citra Grayscale

Gambar skala abu-abu disimpan dalam format 8 bit untuk setiap sampel piksel, memungkinkan sebanyak 256 intensitas. Format ini sangat membantu dalam pemrograman karena manipulasi tidak terlalu banyak bit. Dalam aplikasi lain seperti pencitraan medis dan aplikasi penginderaan jauh juga digunakan format 10.12 atau 16 bit[9]. Ini rumusnya.[10]

$$\text{Grayscale} = ((R * 0.2989) + (G * 0.5870) + (B * 0.1140)) \quad (1)$$

Deskripsi:

R = Red

G = Green

B = Blue

2.2 Ekstraksi Ciri

Ekstraksi ciri adalah proses pengambilan ciri yang terkandung dalam objek dalam gambar. Beberapa proses ekstraksi ciri mungkin perlu mengubah gambar input sebagai citra biner, pola penipisan, dan sebagainya. Ekstraksi ciri adalah langkah pertama dalam menentukan parameter sebagai pencapaian analisis tekstur gambar. Tujuan ekstraksi ciri adalah untuk mengurangi data aktual dengan mengukur ciri tertentu yang membedakan pola input satu sama lain. Dalam penelitian ini dilakukan dengan GLCM, yang merupakan matriks antara menyajikan hubungan bertetangga antar piksel dalam citra ke berbagai arah orientasi dan jarak spasial. Langkah setelah proses *pre-processing* adalah membentuk GLCM. GLCM yang digunakan adalah GLCM 4 arah (0 derajat, 45 Derajat, 90 Derajat dan 135 Derajat) hanya digunakan 0 derajat. Ekstraksi ciri yang digunakan adalah Kontras, Korelasi, energi dan homogenitas[11].

1. Kontras

Mengukur frekuensi spasial citra glcm dan perbedaan momen. Perbedaan yang dimaksudkan adalah perbedaan tinggi dan piksel rendah. Kontras akan menjadi 0 jika piksel tetangga memiliki nilai yang sama. Rumus diperlihatkan dalam persamaan (2).

$$\text{Kontras} = \sum_{i,j=0}^{N-1} (i - j)^2 p(i, j) \quad (2)$$

2. Korelasi

Mengukur linearitas (probabilitas sendi) dari sepasang piksel. Rumus diperlihatkan dalam persamaan (3).

$$\text{Korelasi} = \sum_{i,j=0}^{N-1} p_{i,j} \frac{(i-\mu_i)(j-\mu_j)}{\sqrt{\sigma_i^2 \sigma_j^2}} \quad (3)$$

3. Energi

Energi adalah hasil perhitungan yang terkait dengan jumlah keragaman intensitas keabu-abuan dalam gambar. Rumus diperlihatkan dalam persamaan (4).

$$\text{Energi} = \sum_{i,j} i \cdot j^d (i, j) \quad (4)$$

4. Homogenity

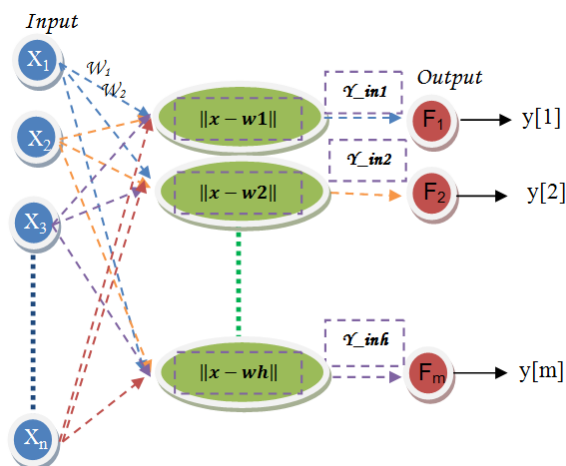
Homogenitas adalah representasi dari ukuran nilai kesamaan variasi intensitas gambar. Jika semua nilai piksel memiliki nilai yang seragam maka homogenitas memiliki nilai maksimum. Rumus diperlihatkan dalam persamaan (4).

$$\text{Homogenitas} = \sum_i \sum_j \frac{p_d(i,j)}{i+|i-j|} \quad (5)$$

2.3 Learning Vector Quantization

Learning Vector Quantization (LVQ) adalah suatu metode pelatihan untuk melakukan pembelajaran pada lapisan kompetitif yang terawasi (supervised learning) yang arsitektur jaringannya berlayer tunggal (single layer). Kelas-kelas yang didapatkan sebagai hasil dari lapisan kompetitif ini hanya tergantung pada jarak antara vektor-vektor input[12].

Learning Vector Quantization (LVQ) Arsitektur Learning Vector Quantization dapat dilihat pada gambar di bawah ini yang menunjukkan bahwa yang bertindak sebagai dendrite atau data input adalah X1 hingga X6, kemudian yang bertindak sebagai sinapsis atau bobot adalah W, sedangkan soma atau badan sel jaringan ini adalah perhitungan $[x-w1]$ hingga $[x-w2]$, dan kemudian yang bertindak sebagai axon atau data output adalah Y[13].



Gambar 1 Arsitektur LVQ

Algoritmanya sebagai berikut:

1. Tentukan maksimum *epoch* (banyaknya proses pelatihan yang akan diulangi), *eps* (*error minimum*) yang diharapkan dan nilai *alpha*.
2. Hasil ekstraksi ciri pertama dari masing-masing pola digunakan sebagai data awal (inisialisasi). Data inisialisasi ini akan diisi sebagai nilai bobot awal (*w*).
3. *Epoch* = 0.
4. Selama (*Epoch* <= *MaxEpoch*) dan (*Alpha* >= *Eps*), maka lakukan hal berikut:
 - a. *Epoch* = *Epoch* + 1
 - b. Untuk setiap data hasil ekstraksi ciri, lakukan hal berikut:
 - 1) Set *X* = hasil ekstraksi ciri fitur
 - 2) Set *T* = target dari setiap kelas
 - 3) Hitung hasil ekstraksi ciri fitur saat ini dengan masing-masing bobot. Hitung jarak hasil ekstraksi ciri fitur pertama dengan setiap bobot, maka rumus yang digunakan adalah sebagai berikut:

$$Jarak = \sqrt{(x_{11} - w_{11})^2 + (x_{12} - w_{12})^2 + \dots + (x_{1m} - w_{1m})^2} \quad (6)$$

Dengan:

X_{1m} = ekstraksi ciri dari pola-1 sampai ke-*m*

W_{1m} = bobot $W(1,m)$

M = banyaknya ekstraksi ciri

- 4) Bila nomor kelas pada bobot yang memiliki jarak terkecil sama dengan nilai target (*T*), maka hitung:

$$w_j(\text{baru}) = w_j(\text{lama}) + \alpha(-w_j(\text{lama})) \quad (7)$$

- 5) Bila tidak, maka hitung:

$$w_j(\text{baru}) = w_j(\text{lama}) - \alpha(x - w_j(\text{lama})) \quad (8)$$

- c. Kurangi nilai *Alpha*:

$$\alpha = \alpha - (0.1 * \alpha) \quad (9)$$

5. Simpan bobot hasil pelatihan (*w*). Arsitektur jaringan LVQ dapat dilihat pada Gambar dibawah ini:

2.4 Euclidean Distance

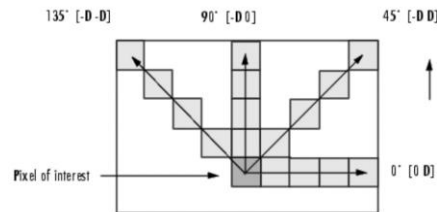
Euclidean distance adalah perhitungan jarak dari 2 buah titik dalam Euclidean space. Euclidean ini berkaitan dengan Teorema Pythagoras dan biasanya diterapkan pada 1, 2 dan 3 dimensi[14].

2.5 Metode GLCM (Gray Level Co-Occurrence Matrix)

Gray Level Co-occurrence Matrix (GLCM) adalah matriks yang menjelaskan jumlah pasangan piksel terhadap frekuensi pada jarak dan variasi dalam sudut kemiringan 1 dengan tujuan menghitung nilai

fitur *Gray Level Cooccurrence Matrix* (GLCM). Metode ini digunakan untuk klasifikasi gambar, mengetahui tingkat keabu-abuan, pengenalan tekstur, segmentasi gambar, pengenalan objek, dan analisis warna pada citra.[15]

GLCM adalah metode untuk mengekstraksi ciri berbasis statistik. Perolehan ciri diperoleh dari nilai piksel matriks yang memiliki nilai tertentu dan membentuk sudut pola. Untuk sudut yang terbentuk dari nilai piksel citra menggunakan GLCM adalah 00, 450, 900, 1350.

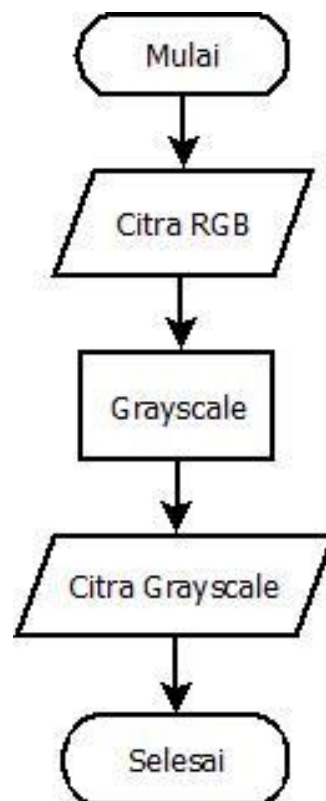


Gambar 2. GLCM

3. Metodologi

3.1 Citra Grayscale

Pada tahap ini, proses mengubah warna gambar RGB (Red Green Blue) menjadi skala abu-abu. Transformasi ini dilakukan agar prosesnya difokuskan pada satu lapisan dan perhitungannya akan lebih cepat. Proses penghitungan sama dengan perhitungan pada konversi citra RGB (Red Green Blue) menjadi skala abu-abu, Proses dilihat pada *flowchart* berikut:



Gambar 3. RGB to Grayscale Color Transformation Process

3.2 Ekstraksi Ciri

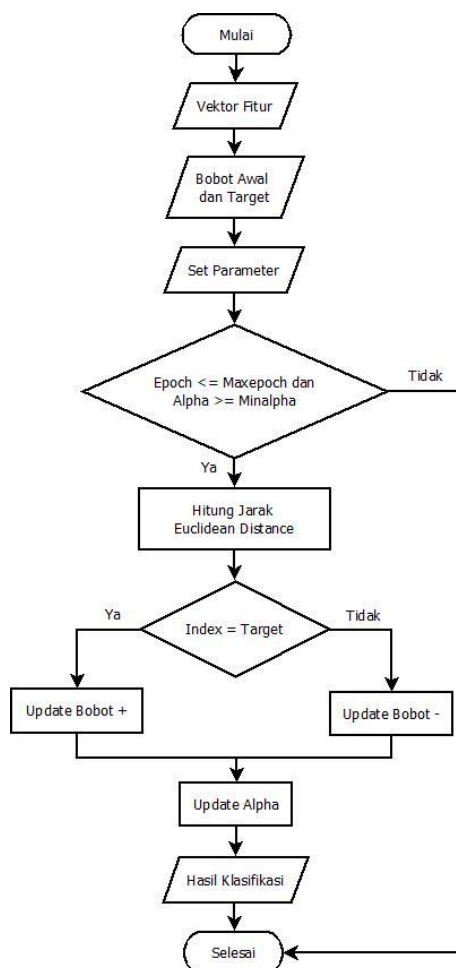
tahap ekstraksi ciri adalah proses pengambilan ciri-ciri yang terdapat pada objek didalam citra. Beberapa proses ekstraksi ciri mungkin perlu mengubah citra masukan sebagai citra biner, melakukan penipisan pola, dan sebagainya. dalam penelitian ini, ekstraksi ciri dilakukan menggunakan metode Gray-Level Co-Occurrence Matrix (GLCM) dengan ciri tekstur yang digunakan ada empat yaitu Kontras, Korelasi, Energi dan Homogenitas.

Tabel 1. Parameter

Parameter	Nilai
Jumlah Data Pelatihan	90
Jumlah Data Uji	30
Jumlah Target	3 (1 = batu, 2 = honi, 3 = madu)
Alfa / Learning Rate	0.01, 0.005,
Learning function	'learnlv1'
Hidden size	10
Epoch / Epoh	100, 500, 1000

3.3 Pelatihan LVQ

Ekstraksi GLCM dari proses sebelumnya menghasilkan 4 (empat) data tekstur (Kontras, Korelasi, Energi dan Homogenitas.) berupa matriks. Matriks tersebut kemudian disimpan ke dalam dataset. Dataset akan digunakan sebagai data masukan pada proses pelatihan menggunakan *Learning Vector Quantization*. Tahapan-tahapan untuk pelatihan terdapat pada *flowchart* berikut:



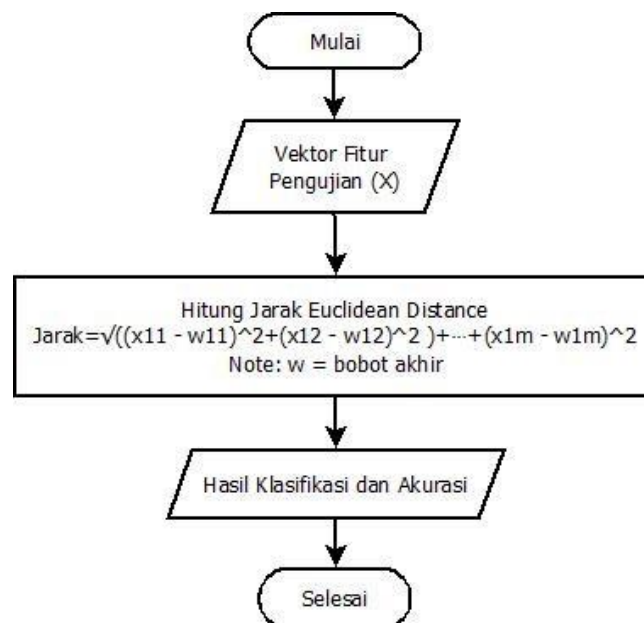
Gambar 4. Learning Vector Quantization Training

Proses pelatihan dimulai dari pengenalan data *input*. Tiga inputan akan diinisialisasi menjadi bobot *training* ke-1, ke-2 dan ke-3, yaitu data ciri nanas batu, nanas honi dan nanas madu. Langkah berikutnya adalah menentukan nilai *epoch*, *learning rate* (*alpha*), dan penurunan nilai *alpha* (*Dec alpha*). Setelah nilai *epoch* ditentukan, pada *epoch* ke-1, ke-2 dan ke-3 akan dihitung jarak terpendek dengan bobot *training* ke-1, ke-2 dan ke-3. setelah jarak terpendek pada bobot ke-, ke-2 dan ke-3

didapat, maka kemudian dipilih jarak yang terkecil untuk proses perkalian dengan nilai *learning rate*. Hasil dari perhitungan tersebut akan digunakan sebagai bobot yang baru, proses dilakukan hingga nilai *input* pembelajaran terakhir didapat pada *epoch* ke-1. Sebelum masuk ke *epoch* selanjutnya, nilai *learning rate* (α) akan diperbaharui dengan cara mengalikan $Dec \alpha$ dengan *learning rate* yang lama. Setelah didapat nilai *learning rate* yang baru akan dilakukan proses perhitungan yang sama seperti *epoch* ke-1 hingga mencapai jumlah *epoch* yang ditentukan. Bobot akhir *training* akan diperoleh setelah tercapainya jumlah *epoch* yang ditentukan untuk digunakan pada proses selanjutnya.

3.4 Pengujian LVQ

Setelah tahap pelatihan dengan LVQ telah selesai maka proses selanjutnya yaitu melakukan pengujian terhadap data uji untuk mencocokkan hasil pengenalan yang didapatkan dari pelatihan dengan pengujian. Bobot akhir dari proses pelatihan akan digunakan untuk menghitung jarak pada data pengujian. Berikut ini *flowchart*nya:

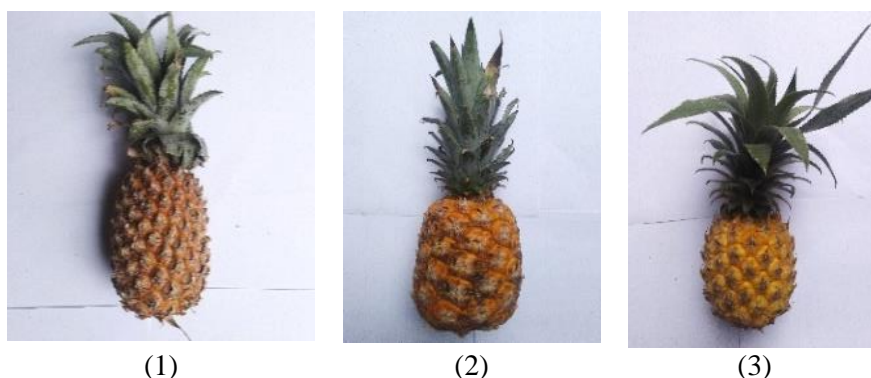


Gambar 5 Pengujian LVQ

4. Hasil dan Pembahasan

4.1 Pembahasan

Pada penelitian ini digunakan 120 citra buah nanas yang terbagi dari 3 kelas data yaitu (nanas batu, nanas honi dan nanas madu), kemudian dibagi menjadi 90 citra sebagai data pelatihan dan 30 citra sebagai data uji. Berikut merupakan sampel data dari masing-masing kelas, yang ditunjukkan pada gambar berikut:



Gambar 6 Citra Nanas batu, honi dan madu

Tabel 2 Percobaan

No	Epoch	Learning rate	Running ke-	Akurasi	
				Pelatihan	Pengujian
1	100	0.01	1	88.8889%	83.3333%
			2	82.2222%	76.6667%
			3	82.2222%	76.6667%
			4	82.2222%	76.6667%
			5	82.2222%	76.6667%
2	100	0,005	1	86.6667%	80%
			2	84.4444%	80%
			3	84.4444%	80%
			4	86.6667%	80%
			5	86.6667%	80%
3	500	0.01	1	88.8889%	83.3333%
			2	86.6667%	80%
			3	84.4444%	76.6667%
			4	84.4444%	76.6667%
			5	84.4444%	76.6667%
4	500	0.005	1	80%	76.6667%
			2	80%	76.6667%
			3	80%	76.6667%
			4	82.2222%	76.6667%
			5	82.2222%	76.6667%
5	1000	0.01	1	84.4444%	76.6667%
			2	84.4444%	76.6667%
			3	88.8889%	83.3333%
			4	82.2222%	80%
			5	80%	73.3333%
	1000	0.005	1	86.6667%	80%
			2	84.4444%	76.6667%
			3	86.6667%	80%
			4	86.6667%	80%
			5	84.4444%	76.6667%

4.2 Hasil Klasifikasi

4.2.1. Untuk hasil klasifikasi data pelatihan dapat dilihat pada tabel berikut:

Tabel 3 Hasil Klasifikasi Data Pelatihan

No	Citra masukan	Learning Vector Quantization		
		Nanas batu	Nanas honi	Nanas madu
1	batu1	1=✓		
2	batu2	1=✓		
3	batu3	3=✗ Terdeteksi (nanas madu)		
4	batu4	1=✓		
5	batu5	1=✓		
6	batu6	2=✗ Terdeteksi (nanas honi)		
7	batu7	1=✓		
8	batu8	1=✓		
9	Batu9	1=✓		
10	Batu10	1=✓		
11	Batu11	1=✓		
12	Batu12	1=✓		
13	Batu13	1=✓		
14	Batu14	3=✗ Terdeteksi		

No	Citra masukan	Learning Vector Quantization		
		Nanas batu	Nanas honi	Nanas madu
		(nanas madu)		
15	Batu15	1=✓		
17	Batu17	1=✓		
16	Batu16	1=✓		
18	Batu18	1=✓		
19	Batu19	1=✓		
20	Batu20	1=✓		
21	Batu21	1=✓		
22	Batu22	1=✓		
	Batu23	3=✗ Terdeteksi		
23		(nanas madu)		
24	Batu24	1=✓		
	Batu25	3=✗ Terdeteksi		
25		(nanas madu)		
	Batu26	2=✗ Terdeteksi		
26		(nanas honi)		
27	Batu27	1=✓		
28	Batu28	1=✓		
29	Batu29	1=✓		
30	Batu30	1=✓		
31	Honi1		2=✓	
32	Honi2		2=✓	
33	Honi3		2=✓	
34	Honi4		2=✓	
35	Honi5		2=✓	
36	Honi6		2=✓	
37	Honi7		2=✓	
38	Honi8		2=✓	
39	Honi9		2=✓	
40	Honi10		2=✓	
41	Honi11		2=✓	
			1=✗ Terdeteksi	
42	Honi12		(nanas batu)	
43	Honi13		2=✓	
44	Honi14		2=✓	
45	Honi15		2=✓	
46	Honi16		2=✓	
47	Honi17		2=✓	
48	Honi18		2=✓	
49	Honi19		2=✓	
50	Honi20		2=✓	
51	Honi21		2=✓	
	Honi22		1=✗ Terdeteksi	
52			(nanas batu)	
53	Honi23		2=✓	
54	Honi24		2=✓	
55	Honi25		2=✓	
56	Honi26		2=✓	
57	Honi27		2=✓	
58	Honi28		2=✓	
59	Honi29		2=✓	
60	Honi30		2=✓	
61	Madu1			3=✓
62	Madu2			3=✓
63	Madu3			3=✓
64	Madu4			3=✓
65	Madu5			3=✓
66	Madu6			3=✓

No	Citra masukan	Learning Vector Quantization		
		Nanas batu	Nanas honi	Nanas madu
67	Madu7			3= ✓
68	Madu8			3= ✓
69	Madu9			3= ✓
70	Madu10			3= ✓
71	Madu11			3= ✓
72	Madu12			3= ✓
73	Madu13			3= ✓
74	Madu14			3= ✓
75	Madu15			3= ✓
76	Madu16			3= ✓
77	Madu17			3= ✓
78	Madu18			3= ✓
79	Madu19			3= ✓
80	Madu20			2= ✗ Terdeteksi (nanas honi)
81	Madu21			3= ✓
82	Madu22			3= ✓
83	Madu23			3= ✓
84	Madu24			3= ✓
85	Madu25			3= ✓
86	Madu26			3= ✓
87	Madu27			3= ✓
88	Madu28			3= ✓
89	Madu29			2= ✗ Terdeteksi (nanas honi)
90	Madu30			3= ✓

Untuk proses pelatihan dengan menggunakan 90 data pelatihan, terdapat 90 data terklasifikasi dengan benar, di mana citra yang di kenali yang benar 80 salah 10 dengan persentase kumulatif yaitu 88.8889%.

4.2.2 Untuk hasil klasifikasi data Pengujian dapat dilihat pada tabel berikut:

Tabel 4 Hasil Klasifikasi Data Pengujian

No	Citra masukan	Learning Vector Quantization		
		Nanas batu	Nanas honi	Nanas madu
1	batu1	1= ✓		
2	batu2	1= ✓		
3	batu3	1= ✓		
4	batu4	3= ✗ Terdeteksi (nanas madu)		
5	batu5	3= ✗ Terdeteksi (nanas madu)		
6	batu6	2= ✗ Terdeteksi (nanas honi)		
7	batu7	1= ✓		
8	batu8	1= ✓		
9	batu9	1= ✓		
10	batu10	1= ✓		
11	Honi1		2= ✓	
12	Honi2		2= ✓	
13	Honi3		1= ✗ Terdeteksi (nanas batu)	
14	Honi4		2= ✓	
15	Honi5		2= ✓	
16	Honi6		2= ✓	
17	Honi7		2= ✓	

No	Citra masukan	Learning Vector Quantization		
		Nanas batu	Nanas honi	Nanas madu
18	Honi8		2=✓	
19	Honi9		2=✓	
20	Honi10		2=✓	
21	Madu1			3=✓
22	Madu2			3=✓
23	Madu3			3=✓
24	Madu4			3=✓
25	Madu5			3=✓
26	Madu6			3=✓
27	Madu7			3=✓
28	Madu8			3=✓
	Madu9			2=✗ Terdeteksi (nanas honi)
29				
30	Honi10			3=✓

Proses pengujian dengan menggunakan 30 data uji, terdapat 30 data terklasifikasi dengan benar, di mana citra yang di kenali yang benar 25 salah 5 dengan persentase kumulatif yaitu 83.3333%.

5. Kesimpulan

Berdasarkan data hasil penelitian dan pembahasan yang telah diuraikan maka terdapat beberapa hal penting yang dapat disimpulkan dari penelitian ini, diantaranya adalah sebagai berikut:

1. Pada penelitian ini 120 citra nanas yang di pakai, terdiri 90 citra sebagai data latih masing-masing 30 citra nanas batu, 30 citra nanas honi, 30 citr nanas madu dan 30 citra sebagai data uji, masing-masing 10 citra nanas batu, 10 citra nanas honi, 10 citra nanas madu.
2. Pada data pelatihan dengan algoritma LVQ (Learning Vector Quantization) untuk mengklasifikasikan jenis-jenis nanas dimana citra yang di kenali menghasilkan citra benar 80 salah 10 dengan persentase kumulatif yaitu 88.8889%.
3. Pada data pengujian dengan algoritma LVQ (Learning Vector Quantization) untuk mengklasifikasikan jenis-jenis nanas dimana citra yang di kenali menghasilkan citra benar 25 salah 5 dengan persentase kumulatif yaitu 83.3333%.

6. Referensi

- [1] Y. S. Amelia, "Yolla Shara Amelia, 2013 Karakterisasi Morfologi Dan Hubunagn Filogenetik Sepuluh Kultivar Nanas (*Ananas comosus* (L.) Merr) Di Kabupaten Subang Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu," pp. 1–7, 2008.
- [2] Rahmat, "Bagian tanaman nenas meliputi akar, batang, daun, tangkai buah, buah, mahkota dan anakan (tunas tangkai buah (slip), tunas yang muncul di ketiak daun (shoots), tunas yang muncul dari batang di bawah permukaan tanah (suckers). 4," pp. 4–12, 2008.
- [3] E. Simatupang, "Jaringan syaraf tiruan menggunakan metode perceptron untuk menentukan penyakit pada tanaman buah nanas," vol. 14, pp. 55–60, 2019.
- [4] A. Lustini, "Klasifikasi Tingkat Kematangan Buah Nanas Menggunakan Ruang Warna Red – Green – Blue Dan Hue – Saturation – Intensity," *J. Digit. Teknol. Inf.*, vol. 2, no. 1, p. 1, 2019, doi: 10.32502/digital.v2i1.2283.
- [5] E. Permata and A. Suherman, "Klasifikasi Kualitas Buah *Garcinia Mangostana* L. Menggunakan Metode Learning Vector Quantization," *Semin. Nas. Teknol. Inf. dan Komun.*, vol. 2015, no. March, p. 425, 2015.
- [6] D. Y. Qur'ani and S. Rosmalinda, "Jaringan Syaraf Tiruan Learning Vector Quantization Untuk Aplikasi Pengenalan Tanda Tangan," *Snatika*, vol. 1, no. 1, pp. 93–100, 2010.
- [7] D. Putra, "Pengolahan Citra Digital," no. April, p. 420, 2010.
- [8] A. Harjoko and U. G. Mada, "Pemrosesan Citra Digital untuk Klasifikasi Mutu Buah Pisang Menggunakan Jaringan Saraf Tiruan," *IJEIS (Indonesian J. Electron. Instrum. Syst.)*, vol. 4, no. 1, pp. 57–68, 2014, doi: 10.22146/ijeis.4222.
- [9] H. Al Fatta, "Konversi Format Citra Rgb Ke Format Grayscale Menggunakan Visual Basic," *Semin. Nas. Teknol.*, vol. 2007, no. November, pp. 1–6, 2007, [Online]. Available:

- <http://p3m.amikom.ac.id/p3m/51> - KONVERSI FORMAT CITRA RGB KE FORMAT GRAYSCALE.pdf.
- [10] “Convert RGB Image to Grayscale Image by Using OpenCV Importer - MATLAB & Simulink.” https://www.mathworks.com/help/vision/ug/convert-an-rgb-image-to-gray-scale-using-opencv_mw_a0f09983-6875-47a8-bed2-d2943c58d7dc.html?s_tid=srchtitle (accessed Jan. 14, 2021).
- [11] Anggun Nugroho, “Bab Ii Landasan Teori,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 8–24, 2018.
- [12] R. Hamidi, M. T. Furqon, and B. Rahayudi, “Implementasi Learning Vector Quantization (LVQ) untuk Klasifikasi Kualitas Air Sungai,” *J-Ptiik*, vol. 1, no. 12, pp. 1758–1763, 2017, [Online]. Available: <http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/635>.
- [13] A. S. Romadhon and V. T. Widyaningrum, “Klasifikasi mutu jeruk nipis dengan metode learning vector quantization (LVQ),” *Rekayasa*, vol. 8, no. 2. p. 121, 2015.
- [14] “Teori Pengukuran Jarak | Anugrah Adiwilaga Blogs.” <https://blogs.itb.ac.id/anugraha/2014/09/10/teori-pengukuran-jarak/> (accessed Mar. 09, 2021).
- [15] R. A. Surya, A. Fadlil, A. Yudhana, M. T. Informatika, P. T. Informatika, and U. A. Dahlan, “Ekstraksi Ciri Metode Gray Level Co-Occurrence Matrix (GLCM) dan Filter Gabor Untuk Klasifikasi Citra Batik Pekalongan,” vol. 02, no. 02, pp. 23–26, 2017.