

# Penerapan Algoritma Canny Dan LVQ 3 Untuk Klasifikasi Jenis Tanaman Mangga

Elvia Budianita<sup>1</sup>, Dewi Muliani<sup>2</sup>, Febi Yanto<sup>3</sup>, Pizaini<sup>4</sup>

Teknik Informatika Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru  
Jl. Subrantas Km. 15, Pekanbaru, 0761-562223

e-mail: <sup>1</sup>elvia.budianita@gmail.com, <sup>2</sup>dewi.muliani@students.uin-suska.ac.id,  
<sup>3</sup>febi.yanto@uin-suska.ac.id, <sup>4</sup>pizaini@uin-suska.ac.id

## Abstrak

Tanaman Mangga merupakan jenis tanaman buah dengan nama ilmiah *Mangifera Indica* yang termasuk dalam marga *Mangifera*, mangga terdiri dari 35-40 anggota dan termasuk dalam suku *Anacardiaceae*. Identifikasi morfologi daun hasil persilangan antar jenis (*intraspesifik*) maupun antar sifat baru yang berbeda dari kedua tetuanya akan lebih memperkaya keragaman genetik tanaman mangga, sehingga sulit membedakan antar spesies mangga karena setiap jenis memiliki kemiripan satu dengan lainnya. Oleh karena itu, untuk dapat membedakan jenis tanaman mangga lokal dan mangga setelah dilakukan persilangan dibutuhkan suatu aplikasi pengolahan citra. Pengambilan data dilakukan melalui 2 tahapan yaitu menggunakan scanner dan kamera hp. Metode yang digunakan untuk proses deteksi tepi dan ekstraksi ciri adalah algoritma canny dan Gray Level Co-occurrence (GLCM). Proses klasifikasi menggunakan metode Learning Vector Quantization 3. Pengujian tertinggi pada citra data uji baru mencapai akurasi 57 % dan pada data latih memperoleh 68% dengan menggunakan citra kamera daun tampak depan dan jumlah data latih 88 citra serta jumlah data uji 37 citra. Nilai window terbaik yang digunakan pada metode LVQ3 adalah 0.5.

**Kata kunci:** Canny, GLCM, Learning Rate, LVQ 3, Window

## Abstract

Mango plant is a type of fruit plant with the scientific name *Mangifera Indica* which belongs to the *Mangifera* clan, mango consists of 35-40 members and belongs to the *Anacardiaceae* tribe. Identification of leaf morphology from inter-species (*intraspesifik*) crosses and between new characteristics that are different from the two parents will further enrich the genetic diversity of mango plants, so it is difficult to distinguish between mango species because each species has similarities with one another. Therefore, to be able to differentiate between local and mangoes after crossing is needed an image processing application. Data retrieval is done through 2 stages namely using a scanner and a handphone camera. The method used for edge detection and feature extraction is the canny algorithm and Gray Level Co-occurrence (GLCM). The classification process using the Learning Vector Quantization 3 method. The highest test on the image of the new test data achieved an accuracy of 57% and on the training data obtained 68% by using camera and the amount of training data 88 images and the total test data of 37 images. The best window value used in the LVQ3 method is 0.5.

**Keywords:** Canny, GLCM, Learning Rate, LVQ 3, Window

## 1. Pendahuluan

Tanaman mangga merupakan jenis tanaman buah dengan nama ilmiah *Mangifera Indica* yang termasuk dalam marga *Mangifera*, mangga terdiri dari 35-40 anggota dan termasuk dalam suku *Anacardiaceae*. Pengenalan jenis-jenis pohon mangga dapat dinilai dari bentuk daun yang bermacam-macam sesuai dengan varietas mangga itu sendiri [1]. Variasi yang cukup tinggi pada bentuk, ukuran dan warna daun menunjukkan adanya keragaman genetik yang cukup luas. Identifikasi morfologi daun hasil persilangan antar jenis (*intraspesifik*) maupun antar sifat baru yang berbeda dari kedua tetunya akan memperkaya keragaman genetik tanaman mangga, sehingga akan sulit untuk membedakan antar spesies mangga karena setiap jenis memiliki kemiripan satu dengan lainnya [2].

Di Indonesia terdapat banyak jenis oho mangga yang dapat dikenali dari bentuk daun yang dimilikinya. Kurangnya pemanahan terhadap pengenalan jenis-jenis pohon mangga berdasarkan bentuk daun dari setiap jenisnya membuat masyarakat sulit membedakan pohon mangga lokal dan mangga setelah dilakukan persilangan dari daun pada bibit pohon mangga, ini sebabnya masih banyak masyarakat yang tertipu oleh bibit yang dipilih karena pada saat pohon mangga tumbuh besar buah yang dihasilkan tidak sesuai dengan jenis mangga yang diharapkan.

Pada penelitian yang akan dilakukan membutuhkan sistem informasi yang berguna untuk mendeteksi jenis tanaman mangga berdasarkan daunnya. Mangga dapat dibedakan berdasarkan

daunnya karena setiap daun memiliki bentuk yang berbeda dari banyaknya varietas mangga, daun dapat dilihat berdasarkan letak daun, bentuk daun, permukaan daun, lipatan daun, pucuk daun dan dasar daun [3]. Deteksi daun mangga dilakukan dengan menggunakan metode deteksi tepi *canny* untuk mendeteksi tepian daun mangga, ekstraksi fitur menggunakan *Gray Level Co-occurrence* (GLCM) dan metode klasifikasi menggunakan *Learning Vector Quantization 3* (LVQ 3).

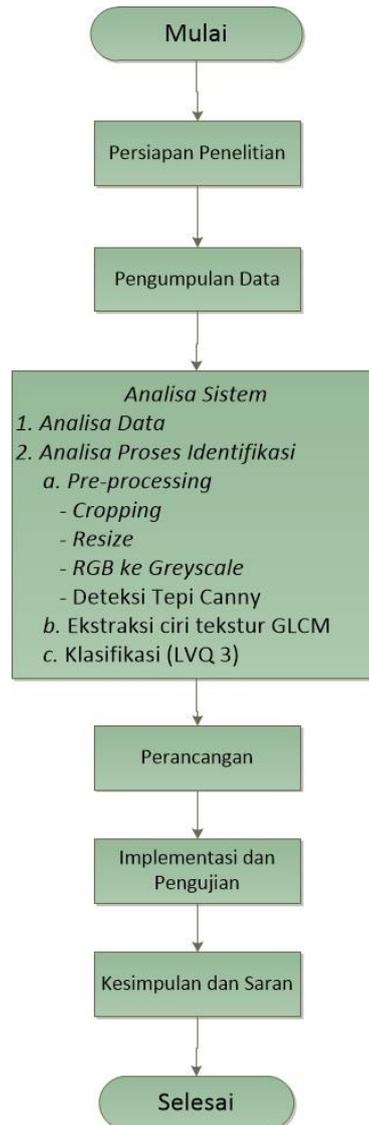
Pada penelitian sebelumnya menggunakan deteksi tepi Canny dilakukan oleh (Ramadhan, Djamal, & Darmanto, 2016) yaitu Klasifikasi Identitas Wajah Untuk Otorisasi Menggunakan Deteksi Tepi dan LVQ. Penelitian ini menggunakan deteksi tepi Canny, penggunaan deteksi tepi *canny* dapat meningkatkan akurasi pengenalan identitas dengan batasan keadaan pencahayaan terang, hasil analisis parameter untuk pelatihan LVQ, dengan menggunakan 25 data uji yang tidak dilatih memperoleh akurasi sebesar 88% dengan hasil pengenalan tiga data uji tidak dikenali dan 22 data uji dikenali. Sedangkan hasil klasifikasi dari 25 data uji yang telah dilatih memperoleh akurasi sebesar 92% dengan hasil pengenalan dua data uji tidak dikenali dan 23 data uji dikenali.

Dalam mengklasifikasi jenis tanaman mangga dilakukan dengan menggunakan metode jaringan syaraf tiruan *Learning Vector Quantization* (LVQ). *Learning Vector Quantization* (LVQ) memiliki beberapa variasi yang masing-masing memiliki perbedaan antara LVQ 1, LVQ 2, LVQ 2.1 dan LVQ 3 yaitu, pada LVQ dasar (LVQ 1) vektor referensi yang paling dekat dengan vektor input saja yang diperbarui. Sedangkan untuk variasi LVQ 2, dua vektor (pemenang dan runner-up) diperbarui jika beberapa kondisi dipenuhi. LVQ 2.1 merupakan perkembangan atau lanjutan dari LVQ 2, dimana dua vektor referensi yang paling dekat dan untuk memperbarui vektor ini adalah bahwa salah satu dari vektor tersebut termasuk kelas yang benar dan vektor lainnya tidak termasuk kedalam kelas tersebut. Dan LVQ 3 memiliki kelebihan selain mencari jarak terdekat, selama pembelajaran unit output diposisikan dengan mengatur dan memperbaharui bobot melalui pembelajaran yang terawasi untuk memperkirakan keputusan klasifikasi. Sedangkan untuk LVQ3, dua vektor (pemenang dan runner-up) diperbaharui jika beberapa kondisi dipenuhi. Ide pengembangan algoritma LVQ adalah jika input memiliki taksiran jarak yang sama dengan vektor pemenang dan runner-up, maka masing-masing vektor tersebut harus melakukan pembelajaran [4]. Klasifikasi jenis tanaman mangga berdasarkan bentuk daun bertujuan untuk menganalisa karakteristik sktruktural citra daun seperti mendeteksi batas-batas tepi. Deteksi Tepi dilakukan untuk mencari batas-batas tepi suatu objek dalam citra. Deteksi tepi berguna untuk proses segmentasi dan identifikasi dalam citra.

Pada penelitian ini, akan dibangun sistem informasi untuk klasifikasi jenis tanaman mangga berdasarkan citra daun mangga yang dibagi menjadi 2 proses pengambilan citra yaitu menggunakan citra kamera Hp dan citra scanner. Dengan jumlah data yang akan digunakan sebagai inputan berjumlah 250 citra daun mangga tampak depan dan tampak belakang menggunakan kamera Hp dengan 5 kelas keluaran (output) jenis tanaman mangga, dan inputan menggunakan data citra scanner berjumlah 450 citra daun tampak depan dengan 10 kelas keluaran (output) jenis tanaman mangga.

## 2. Metode Penelitian

Pada metodologi penelitian berisi rencana kerja yang sesuai dengan alurnya untuk digunakan sebagai pedoman oleh peneliti supaya hasil yang didapatkan sesuai dengan yang diharapkan. Tahapan penelitian bisa dilihat pada Gambar 3.1.



Gambar 1 Tahapan penelitian

### A. Analisa Data

Data yang digunakan diperoleh dari Balai Benih Induk Marpoyan, Kebun penjual tanaman hias dan pohon yang ada pada perkarangan rumah warga. Data diambil secara langsung bertujuan untuk mengetahui jenis daun mangga yang baik dan tidak berpenyakit. Pengambilan dataset daun dari 10 jenis mangga yang diteliti kemudian di ambil gambar dengan menggunakan *scanner* dan 5 jenis mangga di ambil menggunakan kamera hp. Tanaman mangga yang digunakan terdiri dari 10 jenis yaitu mangga Golek, Madu, Arumanis, Apel, Manalagi, Lalijiwa, Udang (Curut), Keweni, dan Pakel (Ambacang), dan Kemang. Pengambilan sampel digunakan sebagai dataset training dan testing dengan jumlah citra daun keseluruhan dari 10 jenis mangga yaitu 450 citra menggunakan scanner dan jumlah citra daun keseluruhan dari 5 jenis mangga yaitu 125 citra daun tampak depan dan 125 citra daun tampak belakang menggunakan kamera hp.

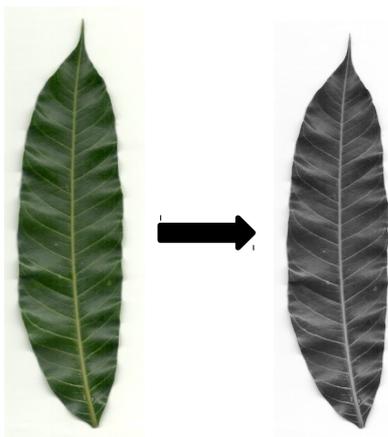
### B. Analisa Proses

Proses yang dilakukan pada penelitian ini dibagi menjadi 3 tahapan yaitu *pre-processing* dilakukan dengan 4 tahapan cropping, resize, konversi citra RGB ke *Grayscale* dan deteksi tepi menggunakan algoritma *canny*, setelah proses *pre-processing* selesai dilakukan proses ekstraksi ciri tekstur menggunakan metode *Gray Level Co-occurrence Matrix* (GLCM) dan proses terakhir yang dilakukan adalah klasifikasi menggunakan metode *Learning Vector Quantization 3* (LVQ 3).

### 1) Pre-processing

Pada tahapan ini akan dilakukan analisa terhadap data yang telah dikumpulkan. Tahapan yang akan dianalisa pada proses pre-processing adalah :

1. Cropping (memotong citra), tujuan dilakukan pemotongan citra adalah untuk membuang bagian yang tidak diperlukan yaitu mengambil bagian daun sesuai dengan batas tepi bentuk daun. Proses cropping dilakukan dengan menggunakan matlab.
2. Rezise (mengubah ukuran) ukuran citra yang besar di resize menjadi 150 × 50 piksel, hal tersebut bertujuan untuk mempercepat perhitungan dalam pemrosesan data pada saat implementasi.
3. Mengubah citra warna menjadi citra grayscale (keabuan) dilakukan menggunakan fungsi yang terdapat pada matlab. Perubahan citra berwarna ke citra grayscale dapat dilihat pada gambar 2.2.



Gambar 2 Perubahan Citra RGB ke Grayscale

4. Mendeteksi tepi citra menggunakan algoritma canny untuk menghilangkan derau pada citra daun mangga, melakukan deteksi tepi menggunakan operator deteksi tepi sobel dan memperkecil garis tepi yang muncul agar menghasilkan garis tepian yang lebih ramping dan jelas untuk deteksi tepi citra daun mangga.

#### Algoritma Canny

Langkah-langkah untuk deteksi tepi terdiri dari enam langkah. Langkah-langkah tersebut adalah sebagai berikut [5].

1. *Smoothing* terhadap citra dengan menggunakan *gaussian filter*

*Smoothing* dilakukan untuk menghilangkan noise yang terdapat dalam citra. Proses *Smoothing* merupakan proses perkalian matrik atau perkalian konvolusi antara citra asli dengan *mask filter gaussian* [6].

$$\frac{1}{115} \times \begin{matrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{matrix} \dots\dots\dots(1)$$

2. *Finding gradien*

Proses ini digunakan untuk mencari tepi gambar dan arah tepi dengan melibatkan operator sobel sebagai *mask* untuk mencari tepi. Mask sobel 3 x 3 digunakan untuk mencari nilai ketetanggaan arah x dan y.

$$G_x = \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \dots\dots\dots(2)$$

Mask sumbu X

$$G_y = \begin{matrix} & 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix}$$

Mask sumbu Y.....(3)

hasil dari mask menggunakan sobel dihitung sebagai Gx dan Gy. Akar kuadrat dari Gx kuadrat ditambah Gy kuadrat dihitung sebagai nilai magnitudo.

$$G = \sqrt{G_x^2 + G_y^2} \dots\dots\dots(4)$$

3. Menghitung nilai tepian

Berdasarkan nilai gradient kemudian akan dicari nilai arah dari tepian citra dengan menggunakan persamaan :

$$\theta = \arctan \left( \frac{G_x}{G_y} \right) \dots\dots\dots(5)$$

4. Menghubungkan garis pada citra berdasarkan nilai arah tepian

Dari nilai tepi yang di dapat, maka dapat ditentukan nilai arah dengan rumus berikut:

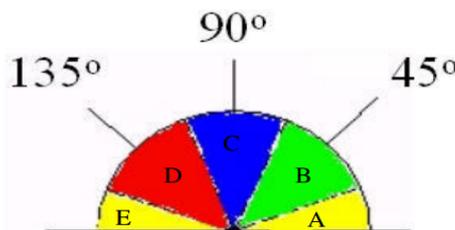
jika  $0 \leq \theta < 22.5$  dan  $157.5 \leq \theta < 180$  maka nilai\_arah = 0

jika  $22.5 \leq \theta < 67.5$  maka nilai\_arah = 45

jika  $67.5 \leq \theta < 112.5$  maka nilai\_arah = 90

jika  $112.5 \leq \theta < 157.5$  maka nilai\_arah = 135.....(6)

Ilustrasi gambar hasil dari proses pencarian nilai arah adalah seperti Gambar 2.3 berikut.



Gambar 3 Ilustrasi Gambar arah

Penjelasan Gambar 2.12 adalah sebagai berikut:

1. A dan E (0 ke 22,5 dan 157,5 ke 180 derajat) diset menjadi 0 derajat
2. B (22,5 ke 67,5 derajat) diset menjadi 45 derajat.
3. C (67,5 ke 112,5 derajat) diset menjadi 90 derajat
4. D (112,5 ke 157,5 derajat) diset menjadi 135 derajat

5. *Non-maximum-suppression*

Langkah selanjutnya adalah mencari tepi selanjutnya dengan nilai arah dan nilai magnitudo yang telah di peroleh dari perhitungan sebelumnya. *Non-maximum-suppression* bertujuan untuk membuang potensi gradien disuatu piksel dari kandidat tepi tetapi bukan merupakan maksimal lokal pada arah tepi menggunakan nilai arah gradien dengan menggunakan rumus berikut:

Jika nilai\_arah (i,j) = 0 maka  $G(i,j) = (\max(G(i,j), (i,j+1), (i,j-1)))$

Jika nilai\_arah (i,j) = 45 maka  $G(i,j) = (\max(G(i,j), (i+1,j-1), (i-1,j+1)))$

Jika nilai\_arah (i,j) = 90 maka  $G(i,j) = (\max(G(i,j), (i+1,j), (i-1,j)))$

Jika nilai\_arah (i,j) = 135 maka  $G(i,j) = (\max(G(i,j), (i+1,j+1), (i-1,j-1))) \dots \dots \dots (7)$

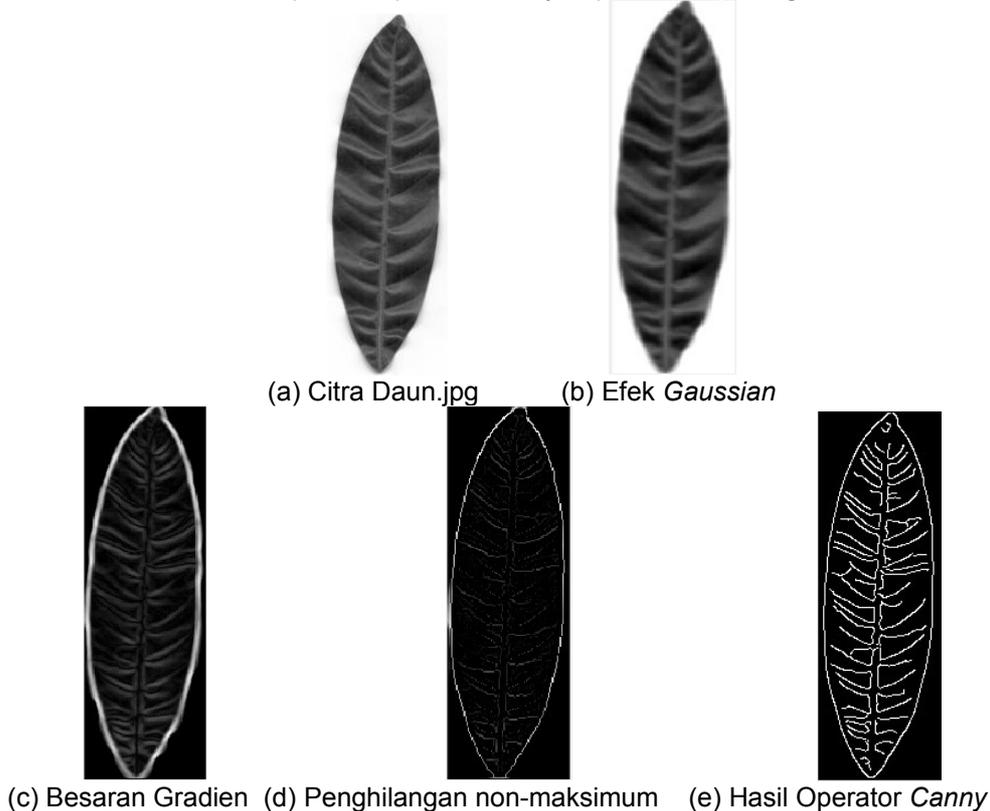
6. *Edge tracking by histeresis*

Tepian final ditentukan dengan menekan semua sisi yang tidak terhubung dengan tepian yang sangat kuat. Proses ini menggunakan dua kali proses *Thresholding* yaitu dengan *High Thresholding* dan *Low Thresholding* terhadap nilai magnitude hasil *Non-Maximum-Suppression*.

Ada beberapa kriteria dalam deteksi tepi paling optimum yang dapat dipenuhi oleh algoritma *Canny* menurut [8] antara lain sebagai berikut:

- a. Mendeteksi dengan baik (kriteria deteksi)  
 Kemampuan untuk meletakkan dan menandai semua tepi yang ada sesuai dengan pemilihan parameter-parameter konvolusi yang dilakukan. Sekaligus memberikan fleksibilitas yang sangat tinggi dalam hal menentukan tingkat deteksi ketebalan tepi sesuai dengan yang diinginkan.
- b. Melokalisasi dengan baik (kriteria lokalisasi)  
 Dengan metode *Canny* dimungkinkan akan dihasilkan jarak yang minimum antara tepi yang dideteksi dengan tepi yang asli.
- c. Respons yang jelas (kriteria respons)  
 Hanya ada satu respons untuk setiap tepi. Sehingga mudah dideteksi dan tidak menimbulkan kerancuan dalam pengolahan citra selanjutnya.

Contoh hasil dari proses operator *canny* dapat dilihat pada gambar 2.4 di bawah ini :



Gambar 4 Contoh hasil antara hingga hasil akhir pada operator *Canny*

2) **Ekstraksi Ciri *Gray Level Co-occurrence (GLCM)***

Untuk mendapatkan fitur *GLCM* terdapat beberapa cara yang dapat dilakukan dengan menggunakan 6 ciri statistik orde dua, yaitu *Angular Second Moment (ASM)*, *Contrast*, *Correlation*, *Variance*, *Inverse different moment (IDM)*, *Entropy*. *ASM* yang merupakan ukuran homogenitas citra dihitung dengan cara seperti berikut:

$$ASM = \sum_i \sum_j \{p(i,j)\}^2 \dots \dots \dots (8)$$

Dalam hal ini, L menyatakan jumlah level yang digunakan untuk komputasi. *Contrast* yang merupakan ukuran keberadaan variasi aras keabuan piksel citra dihitung dengan cara seperti berikut:

$$CON = \sum_k k^2 \left[ \sum_i \sum_j p(i, j) \right] |i, j| = k \dots\dots\dots (9)$$

*Correlation* yang merupakan ukuran ketergantungan linier antar nilai aras keabuan dalam citra dihitung dengan menggunakan rumus:

$$COR = \frac{\sum_i \sum_j (ij) \cdot p(i, j) - \mu_x \mu_y}{\sigma_x \sigma_y} \dots\dots\dots (10)$$

Persamaan diatas didapat dari mean yang merupakan nilai intensitas dari citra keabuan dan standart deviasi terlebih dahulu. Standart deviasi didapat dari akar kuadrat varian yang menunjukkan sebaran nilai piksel dalam citra, dengan rumus sebagai berikut :

$$\mu_x = \sum_i \sum_j i p(i, j) \dots\dots\dots (11)$$

$$\mu_y = \sum_i \sum_j j p(i, j) \dots\dots\dots (12)$$

$$\sigma_x = \sqrt{\sum_i \sum_j p(i, j) (j - \mu_x)^2} \dots\dots\dots (13)$$

$$\sigma_y = \sqrt{\sum_i \sum_j p(i, j) (i - \mu_y)^2} \dots\dots\dots (14)$$

*Variance* digunakan untuk menunjukkan variasi elemen-elemen matriks kookurensi. Citra dengan transisi derajat keabuan kecil akan memiliki variansi yang kecil pula.

$$\sum_j (i - \mu_x)(j - \mu_y) p(i, j) \dots\dots\dots (15)$$

$$VAR = \sum_i i$$

*Inverse Different Momment* (IDM) merupakan nilai yang menunjukkan kehomogenan citra yang berderajat keabuan sejenis. IDM dapat dilakukan dengan cara sebagai berikut :

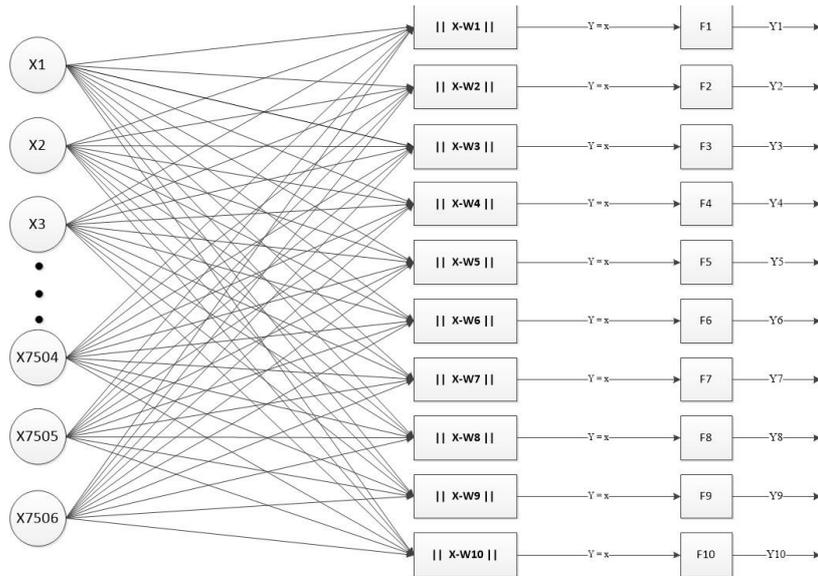
$$IDM = \sum_i \sum_j \frac{1}{1 + |i - j|} p(i, j) \dots\dots\dots (16)$$

*Entropy* merupakan nilai untuk menunjukkan ukuran ketidakaturan bentuk. Harga ENT besar untuk citra dengan transisi derajat keabuan merata dan bernilai kecil jika struktur citra tidak teratur (bervariasi).

$$ENT = - \sum_i \sum_j p(i, j) \log(p(i, j)) \dots\dots\dots (17)$$

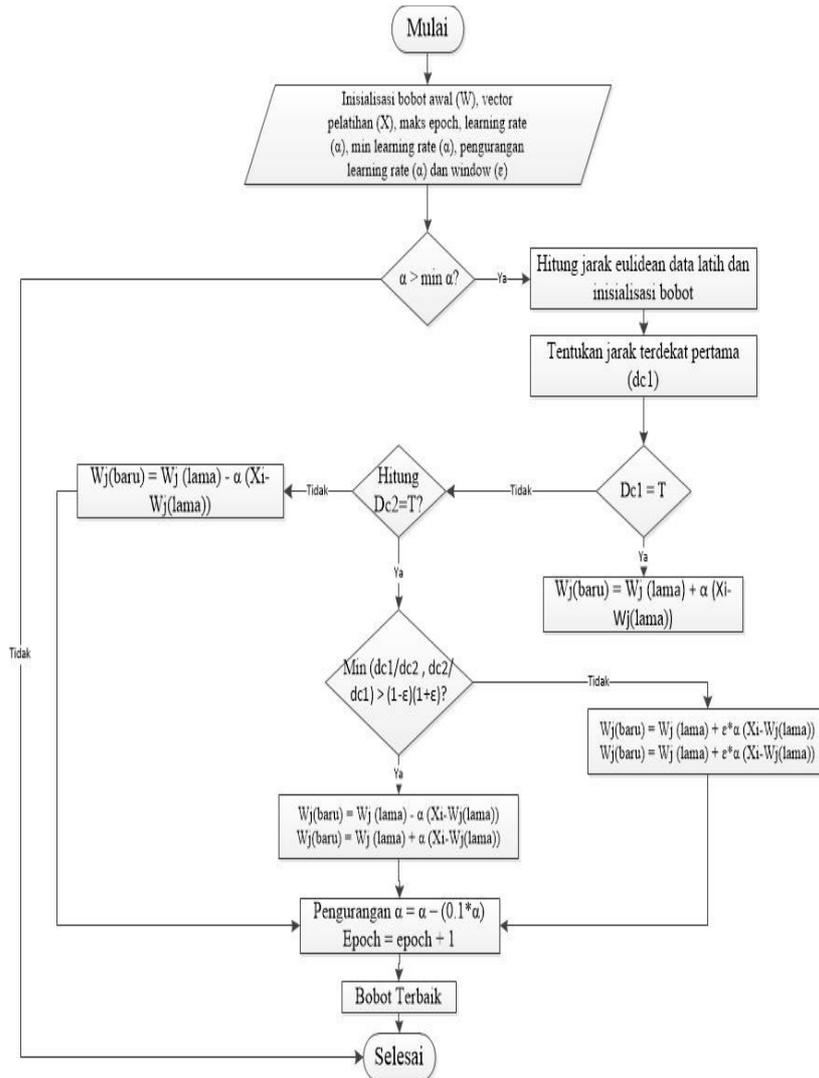
### 3) Learning vector Quantization (LVQ 3)

Pada proses klasifikasi data input layer terdiri dari 7.506 inputan dari perhitungan deteksi tepi canny dan GLCM, dan akan dikelompokkan menjadi 10 kelas sebagai output yaitu ambacang (1), apel (2), golek (3), arum manis (4), manalagi (5), keweni (6), lalijiwa (7), udang (8), madu (9), kemang (10). Arsitektur LVQ 3 untuk klasifikasi jenis tanaman mangga dapat dilihat pada gambar 2.5 berikut ini :



Gambar 5 Arsitektur LVQ 3 Untuk Klasifikasi Jenis Tanaman Mangga

Pada penjelasan sebelumnya telah dijelaskan bahwa metode LVQ 3 tergantung pada jarak antara vektor input dengan bobot dari masing-masing kelas dan vektor input tersebut akan masuk kedalam kelas yang memiliki jarak terdekat [7]. Berikut *flowchart* metode LVQ 3 dapat dilihat pada gambar 2.6.



Gambar 6 Flowchart Pelatihan LVQ 3

Pembelajaran LVQ3 dikembangkan berdasarkan algoritma LVQ1 dan ketentuan dasar teori LVQ3 [7]. Parameter yang dibutuhkan diantaranya adalah:

1. X, vektor-vektor pelatihan ( $X_1, \dots, X_i, \dots, X_n$ ).
2. T, kategori atau kelas yg benar untuk vektor-vektor pelatihan.
3.  $W_j$ , vektor bobot pada unit keluaran ke-j ( $W_{1j}, \dots, W_{ij}, \dots, W_{nj}$ ).
4.  $C_j$ , kategori atau kelas yang merepresentasikan oleh unit keluaran ke-j
5. learning rate ( $\alpha$ ),  $\alpha$  didefinisikan sebagai tingkat pembelajaran. Jika  $\alpha$  terlalu besar, maka algoritma akan menjadi tidak stabil sebaliknya jika  $\alpha$  terlalu kecil, maka prosesnya akan terlalu lama. Nilai  $\alpha$  adalah  $0 < \alpha < 1$ .
6. Nilai pengurangan learning rate, yaitu penurunan tingkat pembelajaran.
7. Nilai minimal learning rate (Mina), yaitu minimal nilai tingkat pembelajaran yang masih diperbolehkan.
8. Nilai window ( $\epsilon$ ), yaitu nilai yang digunakan sebagai daerah yang harus dipenuhi untuk memperbaharui vektor referensi pemenang ( $Y_{c1}$ ) dan runner-up ( $Y_{c2}$ ) jika berada dikelas yang berbeda. Persamaan window ( $\epsilon$ ) :  

$$\text{Min} (dc1/dc2, dc2/dc1) > (1-\epsilon)(1+\epsilon) \dots\dots\dots(18)$$
9. Jika memenuhi kondisi window ( $\epsilon$ ) , maka vektor referensi yang tidak masuk ke dalam kelas yang sama dengan vektor x akan diperbaharui menggunakan persamaan:  

$$Y_{c1} (t+1) = Y_{c1} (t) - \alpha (t) [x(t) - Y_{c1} (t)] \dots\dots\dots(19)$$
10. Sedangkan vektor refrensi yang tidak masuk ke dalam kelas yang sama dengan vektor x akan diperbaharui menggunakan persamaan:

$$Yc_2(t+1) = Yc_2(t) + \alpha(t) [x(t) - Yc_2(t)] \dots\dots\dots(20)$$

11. Jika tidak memenuhi kondisi window ( $\epsilon$ ) maka kedua bobot vector referensi diperbarui menggunakan persamaan (Jatmiko dkk, 2012) :

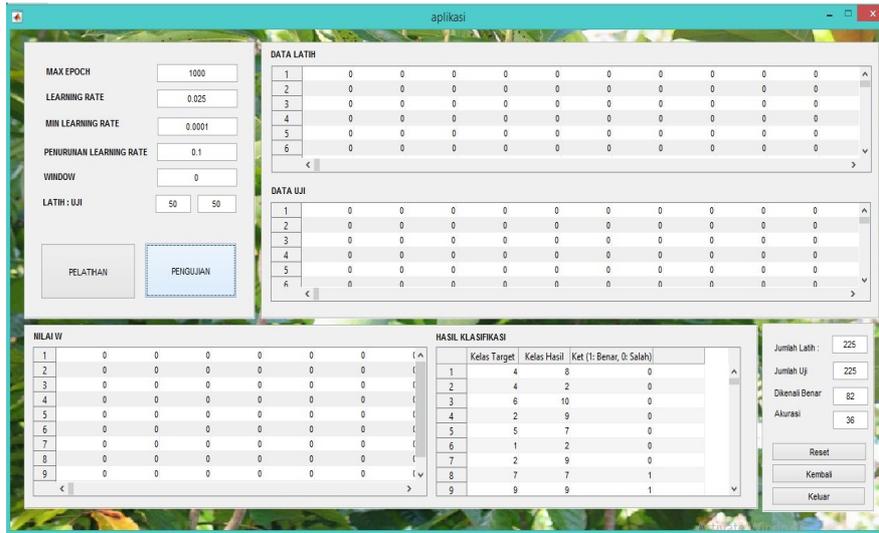
$$Yc_1(t+1) = Yc_1(t) + \epsilon \alpha(t) [x(t) - Yc_1(t)] \dots\dots\dots(21)$$

$$Yc_2(t+1) = Yc_2(t) + \epsilon \alpha(t) [x(t) - Yc_2(t)] \dots\dots\dots(22)$$

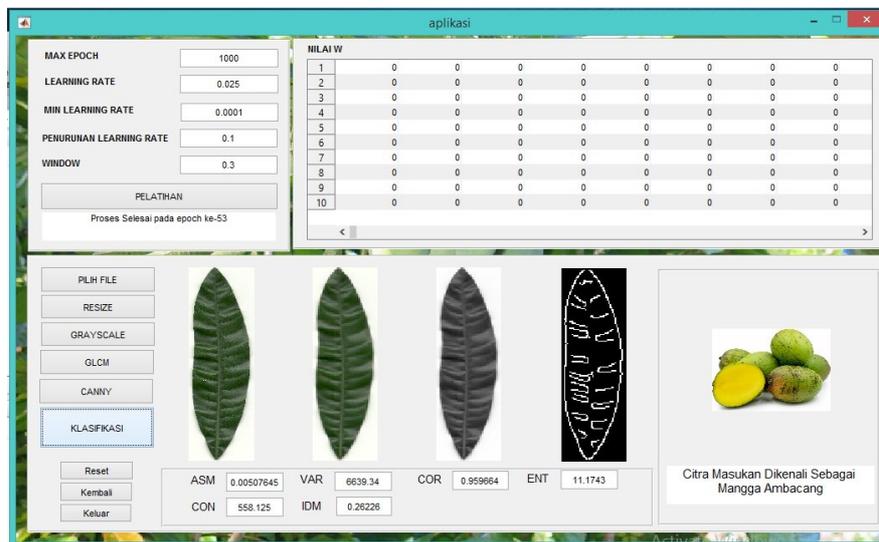
Setelah dilakukan proses pelatihan (training), akan diperoleh bobot-bobot akhir ( $W$ ). Bobot-bobot ini nantinya akan digunakan untuk melakukan pengujian.

### 3. Hasil Dan Analisa

Pada penelitian ini terapat menu pengujian dan pengenalan seperti gambar 3.1 dan gambar 3.2.



Gambar 7 Halaman Pengujian



Gambar 8 Halaman Pengenalan

Pengujian dengan pembelajaran LVQ 3 menggunakan parameter pengujian dengan nilai *learning rate* 0.025, 0.05 dan 0.075, nilai *window* sebesar 0, 0.3 dan 0.5, nilai maksimal epoch sebesar 1000, nilai  $\min\alpha$  0.01 dan pembagian data yaitu 70:30, 60:40, 50:50. Adapun hasil pengujian citra dapat dilihat pada Tabel 3.1

Tabel 1 Hasil pengujian citra daun mangga.

$\epsilon$	$\alpha$	Jumlah Benar	Jumlah Salah	Citra Scanner	Jumlah Benar	Jumlah Salah	Citra Kamera Tampak Depan	Jumlah Benar	Jumlah Salah	Citra Kamera Tampak Belakang
Data 70:30										
0	0.025	55	80	41%	17	20	46%	9	28	24%
0	0.05	52	83	38%	21	16	57%	9	28	24%
0	0.075	36	99	27%	21	16	57%	11	26	30%
0.3	0.025	47	88	35%	17	20	46%	11	26	30%
0.3	0.05	53	82	39%	21	16	57%	11	26	30%
0.3	0.075	36	99	27%	21	16	57%	9	28	24%
0.5	0.025	44	91	33%	17	20	46%	13	24	35%
0.5	0.05	43	92	32%	21	16	57%	17	20	46%
0.5	0.075	42	93	31%	21	16	57%	17	20	46%
Data 60:40										
0	0.025	67	113	37%	22	28	44%	15	35	30%
0	0.05	50	130	28%	26	24	52%	15	35	30%
0	0.075	49	131	27%	26	24	52%	18	32	36%
0.3	0.025	69	111	38%	22	28	44%	18	32	36%
0.3	0.05	62	118	34%	22	28	44%	17	33	34%
0.3	0.075	62	118	34%	27	23	54%	17	33	34%
0.5	0.025	65	115	36%	22	28	44%	19	31	38%
0.5	0.05	67	113	37%	26	24	52%	24	26	48%
0.5	0.075	70	110	39%	26	24	52%	25	25	50%
Data 50:50										
0	0.025	82	143	36%	28	44	45%	22	50	35%
0	0.05	84	141	37%	28	44	45%	22	50	35%
0	0.075	76	149	34%	31	41	50%	22	50	35%
0.3	0.025	94	131	42%	28	44	45%	26	46	42%
0.3	0.05	82	143	36%	28	44	45%	25	47	40%
0.3	0.075	89	136	40%	28	44	45%	22	50	35%
0.5	0.025	76	149	34%	28	44	45%	25	47	40%
0.5	0.05	86	139	38%	28	44	45%	25	50	40%
0.5	0.075	99	126	44%	28	44	45%	33	39	53%

Tabel 2 Hasil Akurasi Pengujian Berdasarkan Data Latih Menggunakan Citra Kamera Hp Daun Tampak Depan

No	Nilai window	Learning rate	Jumlah data latih	Akurasi
1	0.5	0.075	88	66%
2	0.5	0.075	75	68%
3	0.5	0.075	63	65%

Berdasarkan tabel 3.1 dan tabel 3.2 maka dapat diperoleh kesimpulan, pada pengujian LVQ3 dengan nilai window, nilai learning rate dan jumlah data latih dapat mempengaruhi hasil pembelajaran. Pengujian metode canny dan LVQ3 pada citra mangga dapat mengenali citra dengan akurasi tertinggi 68% untuk citra latih yang diuji, dan 57% untuk citra baru yang diuji.

#### 4. Kesimpulan

Berdasarkan dari hasil analisa, perancangan, implementasi dan pengujian yang di lakukan pada penelitian ini, dapat diambil beberapa kesimpulan sebagai berikut :

1. Data scanner dengan jumlah data keseluruhan 450 data dan jumlah kelas klasifikasi 10 kelas jenis mangga akurasi tertinggi mencapai 44 %.
2. Data kamera hp yang terbagi menjadi dua yaitu daun tampak depan dan daun tampak belakang dengan jumlah data 125 citra daun tampak depan dan 125 citra daun tampak belakang dan jumlah kelas klasifikasi 5 kelas jenis mangga dengan akurasi tertinggi mencapai 57 % untuk daun tampak depan dan akurasi tertinggi untuk daun tampak belakang mencapai 53 %.
3. Pengujian tertinggi dari perbandingan keseluruhan data yaitu mencapai akurasi 57 % pada pengujian menggunakan citra kamera daun tampak depan dan pembagian 70 % : 30 % dengan jumlah data latih 88 citra dan jumlah data uji 37 citra.

4. Pada penelitian selanjutnya dengan menggunakan metode deteksi tepi canny agar tidak melakukan resize pada gambar karna gambar yang di resize tidak dapat mendeteksi tulang daun dan mempengaruhi hasil dari deteksi tepi canny.

**Daftar Pustaka:**

- [1] Agustin, S., & Prasetyo, E. (2011). Klasifikasi jenis pohon mangga gadung dan curut berdasarkan tesktur daun. *Sesindo*, 58–64. <https://doi.org/10.13140/RG.2.1.2422.2168>
- [2] Nilasari, A. N. (2013). IDENTIFIKASI KERAGAMAN MORFOLOGI DAUN MANGGA (*Mangifera indica* L .) PADA TANAMAN HASIL PERSILANGAN ANTARA VARIETAS ARUMANIS 143 DENGAN PODANG URANG UMUR 2 TAHUN IDENTIFICATION
- [3] Ramadhan, G., Djamal, E. C., & Darmanto, T. (2016). Klasifikasi Identitas Wajah Untuk Otorisasi Menggunakan Deteksi Tepi dan LVQ. *Seminar Nasional Aplikasi Teknologi Informasi (SNATI) 2016 Yogyakarta*, (August), 37–41.
- [4] Budianita, E., & Prijodiprodjo, W. (2013). Penerapan Learning Vector Quantization (LVQ) untuk Klasifikasi Status Gizi Anak. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, 7(2), 155–166. <https://doi.org/10.22146/ijccs.3354>
- [5] Kadir, A., & Susanto, A. (2013). Pengolahan Citra Teori dan Aplikasi. Yogyakarta.
- [6] Green, B. 2002. Diakses tanggal 01 Juli 2019. *Canny Edge Detection Tutorial*. [www.pages.drexel.edu/~lcan\\_tut.html](http://www.pages.drexel.edu/~lcan_tut.html).
- [7] Kohonen, T. 1990. *The Self-organizing Map*. Proceedings of the IEEE. Vol 78 (9), pp. 1464-1480