

Analisis Kematangan Buah Pisang dengan Metode Gray Level Co-Occurrence Matrix (GLCM)

Banana Fruit Ripeness Analysis Using the Gray Level Co-Occurrence Matrix (GLCM) Method

M. Tri Wahyudi^{1*}, Sri Lestari²

^{1,2}Prodi Matematika Universitas Jambi - Indonesia

*e-mail: lestari.20702@gmail.com

Abstrak

Perkembangan teknologi yang semakin maju khususnya komputer vision telah menjadi sebuah tantangan dalam menganalisis dan memproses suatu citra. Algoritma yang digunakan pada computer vision sangat sulit untuk mengenali tekstur dari suatu citra. Berbeda dengan penglihatan manusia yang dapat mengenali tekstur dengan mudah. Menganalisis suatu citra diperlukan perhitungan secara matematis pada objek baik itu secara piksel maupun geometris. Sehingga digunakanlah suatu metode pengenalan tekstur yaitu Gray Level Co-Occurrence Matrix atau GLCM. Metode GLCM adalah suatu metode ekstraksi order kedua pada fitur statistik tekstur. Metode GLCM memiliki beberapa parameter seperti kontras, korelasi, energi dan homogenitas sebagai fitur ekstraksi ciri dalam pemrosesan citra. Tujuan dilakukannya penelitian tentang analisis kematangan ini adalah untuk mengetahui apakah metode Gray Level Co-Occurrence Matrix (GLCM) dapat digunakan dan akurat dalam menganalisis kematangan buah pisang. Dalam penggunaan metode GLCM (Gray Level Co-Occurrence) kematangan buah pisang dapat dianalisis. Dengan hasil akhir berupa sederat angka yang akan dikonversikan menjadi informasi tingkat kematangan buah pisang tersebut. Namun analisis kematangan buah dengan menggunakan GLCM pada program ini belum akurat 100%.

Kata Kunci: Citra Digital, tekstur, GLCM.

Abstract

The development of increasingly advanced technology, especially computer vision, has become a challenge in analyzing and processing an image. The algorithm used in computer vision is very difficult to recognize the texture of an image. In contrast to human eyesight which can recognize textures easily. Analyzing an image requires mathematical calculations on objects, both pixels and geometrically. So that a texture recognition method is used, namely the Gray Level Co-Occurrence Matrix or GLCM. The GLCM method is a second order extraction method for statistical features of textures. The GLCM method has several parameters such as contrast, correlation, energy and homogeneity as feature extraction features in image processing. The purpose of this research on ripeness analysis is to determine whether the Gray Level Co-Occurrence Matrix (GLCM) method can be used and is accurate in analyzing the ripeness of bananas. Using the GLCM (Gray Level Co-Occurrence) method, banana ripeness can be analyzed. With the final result in the form of a series of numbers that will be converted into information on the level of maturity of the banana. However, fruit ripeness analysis using GLCM in this program is not 100% accurate.

Keywords: Digital Image, texture, GLCM.

Pendahuluan

Perkembangan teknologi yang kian pesat khususnya computer vision telah menjadi suatu tantangan besar dalam menganalisis dan memproses suatu citra. Penggunaan computer vision banyak digunakan dengan tujuan sebagai perangkat keras yang dapat mengenali suatu objek dengan penglihatan atau pengenalan layaknya manusia. Menurut Bhosle (2013), Tekstur merupakan hal yang sangat penting dalam sebuah aplikasi computer vision. Algoritma pada computer vision sangat sulit untuk mengenali tekstur pada citra, berbeda dengan penglihatan manusia yang dengan mudah dapat mengenali tekstur. Maka dengan bantuan program pengolah citra dengan aplikasi matlab, nantinya suatu citra dapat dianalisis sehingga memudahkan dalam pengelompokan atau pengklasifikasian citra tersebut.

Citra digital merupakan sebuah *array* dimana berisi nilai-nilai yang nyata maupun kompleks yang direpresentasikan dengan deratan bit tertentu. Citra dapat berupa citra RGB, citra Lab, dan citra *Grayscale*. Citra RGB merupakan pencampuran 3 warna dasar yaitu merah, hijau, biru dan dengan saturasi penuh akan menghasilkan warna baru yaitu putih (Pratt, 2007). Citra grayscale adalah citra yang ditampilkan dalam warna abu-abu, dan akan bervariasi warna putih pada intensitas terkuat serta bervariasi warna hitam pada intensitas terlemah.

Menganalisis suatu citra diperlukan perhitungan yang bersifat matematis pada objek baik secara piksel ataupun geometris. Pada penelitian sebelumnya yang dilakukan oleh Wayan Pandri Adnyana tahun 2018, suatu citra dapat dikenali dengan metode *Statistical Texture Descriptor (STD)*. Namun pada penelitian kali ini pisang akan menjadi objek atau citra yang akan dianalisis dalam tiga kategori yaitu mentah, matang, dan busuk menggunakan metode *GLCM*. *GLCM* atau *Gray Level Co-Occurrence Matrix* adalah metode yang digunakan untuk membentuk ciri atau fitur yang tidak didapat pada nilai piksel saja (Kadir dan Susanto, 2013). Pada penelitian ini, ciri yang akan digunakan yaitu ciri bentuk dan ciri tekstur, dimana ciri ini menjadi pembeda objek pisang satu dengan yang lainnya. Serta akan dilihat pula keakuratan penggunaan metode *GLCM* ini.

Pada metode *GLCM* terdapat beberapa fitur yang digunakan sebagai ekstraksi ciri yaitu kontras, korelasi, energi, dan homogenitas. Adapun rumus matematis dalam ekstraksi ciri tersebut sebagai berikut.

- a) Kontras = $\sum_i \sum_j (i - j)^2 p(i, j)$
- b) Korelasi = $\sum_i \sum_j \frac{(i - \mu_i) * (j - \mu_j) * GLCM(i, j)}{\sigma_i * \sigma_j}$
- c) Energi = $\sum_i \sum_j p^2(i, j)$
- d) Homogenitas = $\sum_i \sum_j \frac{p(i, j)}{1 + |i - j|}$

Dengan :

i = tingkat keabuan baris ke-i

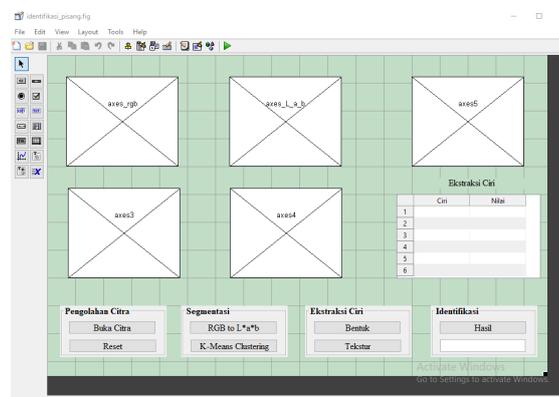
j = tingkat keabuan kolom ke-j

$P(i,j)$ = peluang keabuan baris ke- i , kolom ke- j

Metode Penelitian

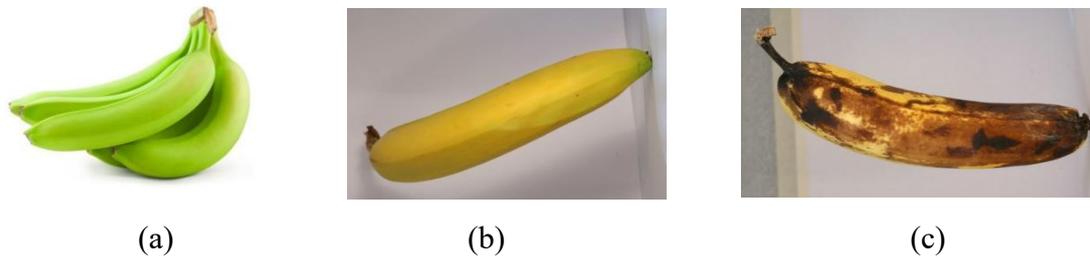
Dalam perancangan program analisis kematangan buah pisang ini memanfaatkan sistem *Graphical User Interface (GUI)* yang ada pada aplikasi *Matlab*. Dalam proses perancangan program analisis kematangan buah ini terdiri dari beberapa proses, yaitu pengambilan citra, segmentasi, ekstraksi ciri, dan identifikasi keluaran. Proses pengambilan citra menggunakan kamera ponsel biasa lalu gambar disimpan dalam format jpg. Pengambilan citra berupa citra *RGB* yang kemudian akan diproses ke citra lain sehingga didapat keluaran (hasil akhir). Ciri yang diekstrak berupa ciri bentuk yang dikonversikan ke dalam bentuk citra *biner* dan ciri tekstur akan dikonversikan dalam bentuk *grayscale*.

Kegunaan segmentasi citra adalah untuk memisahkan antara latar depan (*foreground*) dengan latar belakang menggunakan metode *k-means clustering*. Proses *clustering* dilakukan dengan mengkonversikan citra *RGB* menjadi citra Lab. Dengan komponen a dan b dari citra Lab tersebut sebagai nilai masukan dalam algoritma *k-means*. Citra hasil segmentasi dengan *k-means clustering* hanya berupa latar depan saja, dan berlatar belakang warna hitam. Citra hasil segmentasi tersebut diproses dengan tahap ekstraksi ciri. Hasil ekstraksi ciri dari fitur *GLCM* tersebut kemudian dibandingkan dengan hasil ekstraksi ciri dari basis data. Pada analisis kematangan buah ini menggunakan fitur *GUI (Graphical User Interface)* pada aplikasi *Matlab*. Kemudian dalam fitur *GUI* tersebut akan dibuat suatu pemrograman penganalisan kematangan buah. Setelah pemrograman dan perancangan *GUI* selesai barulah *GUI* dapat dijalankan. *GUI* akan menampilkan gambar yang telah diambil, kemudian mengkonversikan gambar tersebut ke berbagai macam citra sesuai dengan perintah. Kemudian citra tersebut akan dianalisis dan diekstrak cirinya sehingga didapat hasil berupa teks pengklasifikasian objek. Secara keseluruhan dapat dilihat pada gambar berikut:



Gambar 1. Tampilan *GUI*

Penelitian ini menggunakan 3 klasifikasi yang berbeda pada pisang. Adapun data yang digunakan dalam penelitian ini, yaitu : Pisang Mentah, Pisang Matang, dan Pisang Busuk yang diambil melalui website textures.com. Berikut ini merupakan 3 sampel yang berbeda.



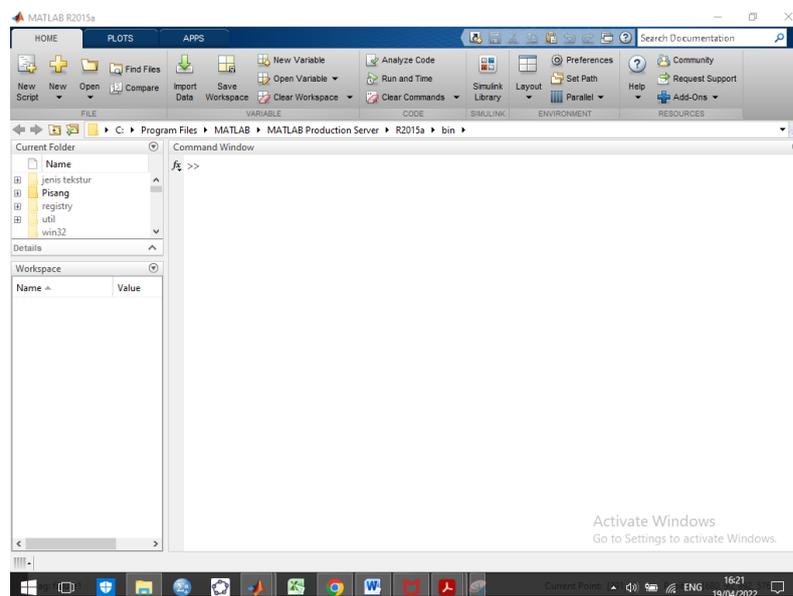
Gambar 2. (a) Pisang Mentah. (b) Pisang Matang. (c) Pisang Busuk

Hasil dan Pembahasan

Pengujian Program Analisis Kematangan Buah

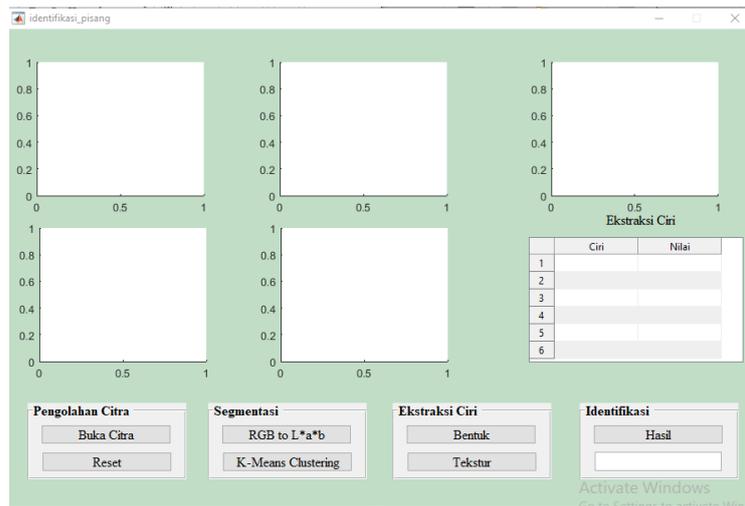
Pengujian program bertujuan untuk mengetahui apakah program dapat berjalan dengan baik sesuai dengan perintah user. Proses pengujian program analisis kematangan buah yaitu sebagai berikut :

1. Langkah pertama, yaitu membuka aplikasi *Matlab*.
2. Setelah beberapa saat, maka aplikasi *Matlab* akan terbuka dengan tampilan awal sebagai berikut :



Gambar 1. Tampilan awal Matlab

3. Untuk menjalankan program ketik “`identifikasi_pisang`” pada *command window*, kemudian *enter*. Maka tampilan *GUI* yang telah dirancang akan muncul seperti berikut :



Gambar 2. Tampilan *GUI* pada Program Analisis Kematangan Buah

Tombol Buka Citra

Tombol “Buka Citra” berfungsi untuk mencari atau menemukan gambar yang akan di uji pada folder komputer. Adapun proses tombol “Buka Citra” sebagai berikut :

```

% --- Executes on button press in pushbutton browse.
function pushbutton_browse_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_browse (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
[filename,pathname] = uigetfile('*.jpg');

if ~isequal(filename,0)
    Img = imread(fullfile(pathname,filename));
    axes(handles.axes_rgb)
    imshow(Img)
    title('Citra RGB')
else
    return
end

handles.Img = Img;
guidata(hObject, handles)
    
```

Gambar 3. Script tombol Buka Citra

Tahap awal yaitu dengan membuka file yang berformat jpg dengan proses `[filename,pathname]=uigetfile('*.jpg');`. Script diatas akan menampilkan gambar dalam bentuk Citra *RGB* pada `axes_rgb`.

Tombol Reset

Tombol “Reset” berfungsi untuk menghapus semua tampilan serta gambar yang ada pada jendela *GUI*. Atau dengan kata lain mengembalikan program ke bentuk semula sebelum ada gambar. Proses tombol “Reset” sebagai berikut :

```

% handles structure with handles and user data (see GUIDATA)
axes(handles.axes_rgb)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes_L_a_b)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes3)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes4)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes5)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

set(handles.uitable_ciri,'Data',[])
set(handles.edit_buahan,'String',[])
    
```

Gambar 4. Script tombol *Reset*

Tombol *RGB to L*a*b*

Tombol ini berfungsi untuk mengubah gambar uji yang sebelumnya berada pada ruang warna *RGB* menjadi ruang warna *Lab*.

```

function pushbutton_RGB_to_L_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton_RGB_to_L (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
Img = handles.Img;

% Color-Based Segmentation Using K-Means Clustering
cform = makecform('rgb2lab');
lab = applycform(Img,cform);
axes(handles.axes_L_a_b)
imshow(lab)
title('Citra L*a*b');

handles.lab = lab;
guidata(hObject,handles)
    
```

Gambar 5. Script tombol *RGB to L*a*b*

Langkah pertama yaitu memanggil variabel *Img* yang telah tersimpan di *handles*. Kemudian *Img* akan diubah ke ruang warna *Lab* dengan perintah `lab = applycform(Img,cform)`. Hasil dari perubahan ruang warna tersebut akan ditempatkan di `axes_L_a_b` dengan perintah `axes(handles.axes_L_a_b)`. Perintah `imshow` berguna untuk menampilkan variabel *lab* dengan judul *Citra L*a*b*.

Tombol *K-Means Clustering*

Tombol ini berfungsi untuk menghilangkan *background* yang ada pada objek. Sehingga hanya terdapat *foreground* pada segmentasi citra. Adapun prosesnya sebagai berikut :

```

% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton8 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
Img = handles.Img;
lab = handles.lab;

ab = double(lab(:,1:2:3));
nrows = size(ab,1);
ncols = size(ab,2);
ab = reshape(ab,nrows*ncols,2);

nColors = 2;
[cluster_idx, ~] = kmeans(ab,nColors,'distance','sqEuclidean', ...
'Replicates',3);

pixel_labels = reshape(cluster_idx,nrows,ncols);
segmented_images = cell(1,3);
rgb_label = repmat(pixel_labels,[1 1 3]);

for k = 1:nColors
color = Img;
color(rgb_label == k) = 0;
segmented_images(k) = color;
end

area_cluster1 = sum(find(pixel_labels==1));
    
```

```

area_cluster2 = sum(find(pixel_labels==2));

[~,cluster_min] = min([area_cluster1,area_cluster2]);

Img_bw = (pixel_labels==cluster_min);
Img_bw = imfill(Img_bw,'holes');
Img_bw = bwareaopen(Img_bw,50);

p1asang = Img;
R = p1asang(1,1);
G = p1asang(1,2);
B = p1asang(1,3);
R(-Img_bw) = 0;
G(-Img_bw) = 0;
B(-Img_bw) = 0;
p1asang_rgb = cat(3,R,G,B);
axes(handles.axes3);
imshow(p1asang_rgb);
title('Citra Hasil Segmentasi');

handles.Img_bw = Img_bw;
guidata(hObject, handles)
    
```

Gambar 6. Script tombol *K-Means Clustering*

Tombol Bentuk

Tombol ini berfungsi untuk ekstraksi ciri bentuk berdasarkan parameter *metric* dan *eccentricity*. Nilai dari parameter tersebut akan dapat dilihat di baris pertama dan kedua pada tabel tampilan *GUI*. Adapun prosesnya sebagai berikut:

```

function pushbutton3_Callback(hObject,eventdata,handles)
% hObject handle to pushbutton3 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
Img_bw = handles.Img_bw;

axes(handles.axes3);
imshow(Img_bw);
title('Citra Biner');

stats = regionprops(Img_bw,'Area','Perimeter','Eccentricity');
area = stats.Area;
perimeter = stats.Perimeter;
metric = 4*pi*area/(perimeter^2);
eccentricity = stats.Eccentricity;

ciri_bentuk = cell(2,2);
ciri_bentuk(1,1) = 'Metric';
ciri_bentuk(1,2) = 'Eccentricity';
ciri_bentuk(2,1) = num2str(metric);
ciri_bentuk(2,2) = num2str(eccentricity);

handles.ciri_bentuk = ciri_bentuk;
guidata(hObject, handles)

set(handles.uitable_ciri,'Data',ciri_bentuk,'ColumnName',1:2)
    
```

Gambar 7. Script tombol Bentuk

Tombol Tekstur

Tombol ini berfungsi untuk mengekstraksi ciri terhadap citra *grayscale* yang ada berdasarkan parameter *GLCM*, yaitu berupa *Contrast*, *Correlation*, *Energy*, *Homogeneity*. Adapun prosesnya sebagai berikut :

```

Img = handles.Img;
Img_bw = handles.Img_bw;
ciri_bentuk = handles.ciri_bentuk;

Img_gray = rgb2gray(Img);
Img_gray(-Img_bw) = 0;

axes(handles.axes4);
imshow(Img_gray);
title('Citra Grayscale');

pixel_dist = 1;
GLCM = graycomatrix(Img_gray,'Offset',[0 pixel_dist;-pixel_dist
pixel_dist;-pixel_dist 0;-pixel_dist pixel_dist]);
stats = graycoprops(GLCM,{'contrast','correlation','energy','homogeneity'});
Contrast = mean(stats.Contrast);
Correlation = mean(stats.Correlation);
Energy = mean(stats.Energy);
Homogeneity = mean(stats.Homogeneity);

ciritotal = cell(6,2);
ciritotal(1,1) = ciri_bentuk(1,1);
ciritotal(1,2) = ciri_bentuk(1,2);
ciritotal(2,1) = ciri_bentuk(2,1);
ciritotal(2,2) = ciri_bentuk(2,2);
ciritotal(3,1) = 'Contrast';
ciritotal(4,1) = 'Correlation';
ciritotal(5,1) = 'Energy';
ciritotal(6,1) = 'Homogeneity';
ciritotal(3,2) = num2str(Contrast);
ciritotal(4,2) = num2str(Correlation);
ciritotal(5,2) = num2str(Energy);
ciritotal(6,2) = num2str(Homogeneity);

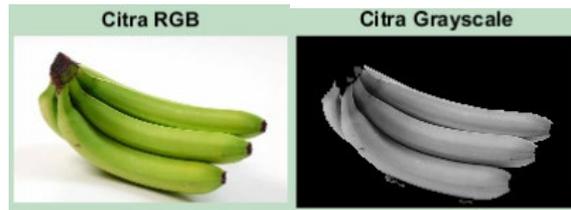
handles.ciritotal=ciritotal;
guidata(hObject,handles)

set(handles.uitable_ciri,'Data',ciritotal,'ColumnName',1:6)
    
```

Gambar 8. Script tombol Tekstur

Tahap awal yaitu pemanggilan variabel yang telah disimpan di *handles* sebelumnya. Kemudian perubahan citra *RGB* ke citra *grayscale* menggunakan perintah *rgb2gray*. Dilanjutkan

dengan ekstraksi tekstur berdasarkan parameter kontras, korelasi, energi, dan homogenitas. Hasil pengubahan citra *RGB* ke *grayscale* dapat dilihat sebagai berikut :



Tombol Hasil

Tombol ini berfungsi untuk memberikan hasil keluaran berupa teks hasil identifikasi gambar uji. Teks ini berasal dari ekstraksi ciri bentuk dan tekstur.

```

ciritotal= handles.ciritotal;
load ciri_database
cirill = handles.cirill;

ciri = zeros(1,6);
for i = 1:6
    ciri(i) = str2double(ciritotal(i,2));
end

[num,-] = size(cirill);
dist = zeros(1,num);
for n = 1:num
    data_base = cirill(n,:);
    jarak = sum((data_base-ciri).^2).*0.5;
    dist(n) = jarak;
end

[~,id] = min(dist);
if isempty(id)
    set(handles.edit_luaran,'String','Unknown')
else
    switch id
        case 1
            tingkat = 'Pisang Mentah';
        case 2
            tingkat = 'Pisang Mentah';
        case 3
            tingkat = 'Pisang Mentah';
        case 4
            tingkat = 'Pisang Mentah';
        case 5
            tingkat = 'Pisang Mentah';
        case 6
            tingkat = 'Pisang Matang';
        case 7
            tingkat = 'Pisang Matang';
        case 8
            tingkat = 'Pisang Matang';
        case 9
            tingkat = 'Pisang Matang';
        case 10
            tingkat = 'Pisang Matang';
        case 11
            tingkat = 'Pisang Busuk';
        case 12
            tingkat = 'Pisang Busuk';
        case 13
            tingkat = 'Pisang Busuk';
        case 14
            tingkat = 'Pisang Busuk';
        case 15
            tingkat = 'Pisang Busuk';
    end
    
```

Gambar 9. Script tombol Hasil

Pada proses ini terdapat pemanggilan *data base* menggunakan perintah *load*. Dalam program ini terdapat 3 teks keluaran yang mewakili tiap kondisi pisang. Teks keluaran ini didapat berdasarkan perbandingan dengan basis data. Berikut adalah nilai fitur basis data :

Tabel 1. Nilai Fitur GLCM basis data

	Metric	Eccentricity	Contrast	Correlation	Energy	Homogeneity
Mentah	0.55915	0.84714	0.1731	0.98323	0.47583	0.97962
Matang	0.29975	0.95838	0.043353	0.9948	0.56043	0.99242
Busuk	0.18847	0.96198	0.088348	0.98667	0.59029	0.97432

Setelah melakukan pengujian program dengan data latih/basis data. Selanjutnya akan dilakukan pengujian dengan sampel yang berbeda dan jauh lebih banyak dari basis data. Hasil pengujian dapat dilihat pada tabel dibawah ini.

Tabel 2. Hasil Pengujian Data Uji

File	Metric	Ecentricity	Contrast	Correlation	Energy	Homogeneity	Hasil
1	0.55915	0.84714	0.1731	0.98323	0.47583	0.97962	Mentah
2	0.8037	0.66144	0.057061	0.98743	0.27772	0.97539	Mentah
3	0.4582	0.89043	0.16648	0.97812	0.43442	0.97262	Mentah
4	0.35724	0.85005	0.18089	0.98614	0.35471	0.97171	Mentah
5	0.43916	0.89038	0.1204	0.98419	0.43829	0.97908	Mentah
6	0.29975	0.95838	0.043353	0.9948	0.56043	0.99242	Matang
7	0.30312	0.95847	0.062481	0.9925	0.55914	0.99066	Matang
8	0.3265	0.98008	0.093769	0.99056	0.45565	0.98538	Matang
9	0.28104	0.95894	0.041799	0.99511	0.58549	0.99174	Matang
10	0.45586	0.97564	0.040693	0.99417	0.55534	0.99282	Matang
11	0.18847	0.96198	0.088348	0.98667	0.59029	0.97432	Busuk
12	0.15357	0.96065	0.099311	0.97426	0.55321	0.97069	Busuk
13	0.15357	0.96065	0.099311	0.97426	0.55321	0.97069	Busuk
14	0.18017	0.97182	0.086529	0.98862	0.46675	0.97076	Matang
15	0.1427	0.96657	0.083588	0.98712	0.53917	0.9726	Busuk

Hasil Pengujian dan Analisis

Setelah proses pengujian program dengan data latih atau *data base* dan data uji, didapatkan hasil bahwa program ini belum akurat 100%. Terdapat kesalahan pengenalan kematangan buah di data uji “latih busuk 4”. Hal ini dikarena *intensitas* warna yang ada pada gambar di objek tersebut. Jadi dapat disimpulkan bahwa warna, cahaya sangat mempengaruhi dalam analisis kematangan buah.

Simpulan

Metode GLCM (*Gray Level Co-Occurance*) dapat digunakan untuk mengkonversikan data citra buah pisang menjadi *numerik*. Dengan parameter *Contrast*, *Correlation*, *Energy*, dan *Homogeneity* sebagai ekstraksi tekstur. Dan parameter *Metric* dan *Ecentricity* sebagai ekstraksi bentuk. Analisis kematangan buah dengan menggunakan *GLCM* pada program ini belum akurat 100%.

Ucapan Terima Kasih

Ucapan terimakasih di berikan kepada semua pihak yang telah berperan dalam penulisan Analisis Kematangan Buah Pisang dengan Metode *Gray Level Co-Occurrence Matrix* (GLCM), sehingga analisis ini dapat dituangkan ke dalam bentuk tulisan dan dapat diinformasikan kepada para pembaca.

Daftar Rujukan

- [1] Adnyana, W.P., 2018, Pengenalan Tekstur dengan Statistical Texture Descriptor, Tugas Akhir, Jurusan Teknik Elektro, FST, Universitas Sanata Dharma, Yogyakarta.
- [2] Bhosle, V.V., Pawar, V.P., 2013. Texture Segmentation: Different Methods. *International Journal of Soft Computing and Engineering (IJSCE)*. vol. 3, issue 5, hal 69-74.
- [3] Fadlisyah. 2008. Pengolahan Citra Digital. Yogyakarta: PT Elex Media Komputindo.
- [4] Fathansyah. 2012. Basis Data. Bandung: Informatika.
- [5] Fatta, H.A. 2007. Konversi Format Citra RGB ke Format Grayscale menggunakan Visual Basic. Yogyakarta: STMIK AMIKOM.
- [6] Honeycutt, C.E., Plotnick, R. 2008. Image analysis techniques and gray-level cooccurrence matrices (*GLCM*) for calculating bioturbation indices and characterizing biogenic sedimentary structures, *Computers & Geosciences*, vol. 34, hal 1461-1472.
- [7] Kadir, A. 2009. Dasar Perancangan & Implementasi Database Relasional. Yogyakarta: Andi.
- [8] Kadir, Abdul, Susanto, Adhi. 2013. Teori dan Aplikasi Pengolahan Citra. Yogyakarta: Andi.
- [9] Prasetyo, E., 2011. Pengolahan Citra Digital Dan Aplikasinya Menggunakan Matlab. Yogyakarta: Penerbit Andi.
- [10] Pratt, W.K. 2007, Digital Image Processing. WileyInterscience, A John Wiley & Sons, Inc.
- [11] Putra, Darma. 2010. Pengolahan Citra Digital. Yogyakarta: Andi Offset.
- [12] RastegarSani, M., dan Modarres, A. F. A. 2019. Playfield Extraction in Soccer Video Based on Lab. Yazd, Iran, IEEE, pp. 1999-2003.
- [13] Rosa & Shalahuddin, M. 2011. Modul Pembelajaran Rekayasa Perangkat Lunak. Bandung: Modula.
- [14] Santi, C.N., S.Pd, M.Kom Mengubah Citra Berwarna Menjadi Gray-Scale dan Citra Biner, Jurnal, FTI, Universitas Stikubank Semarang.
- [15] Szeliski, R. 2010. Computer Vision: Algorithms and Applications. 1 st Edition. Springer.