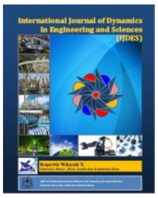
**INTERNATIONAL JOURNAL of DYNAMICS in ENGINEERING and SCIENCES (IJDES)**
**LLDIKTI WILAYAH X**
Vol. 8, No. 2, Oct 2023
e-ISSN: 2502-0692



# DETECTION OF ASPHALT ROAD DAMAGE USING THE GRAY LEVEL CO-OCCURENCE MATRIX METHOD

**Ahmad Yani*[1] , Zulfiqar Tamin [2] , Agung Ramadhanu [3]**
[123]University Putra Indonesia "YPTK" Padang
*Correspondence should be addressed to sdr.ahmadyani@gmail.com


**Article Information**

Submitted :
02 Agust 2023

Accepted :
20 Sept 2023
Published :
01 Oct 2023

**Abstract**

Abstract - This research aims to recognize and classify asphalt road damage based on the level of damage using the Gray Level Co-occurrence Matrix (GLCM) method in image processing. Data on asphalt road damage in Padang City from 2020 to 2022 shows an increase in light damage and a decrease in heavy damage. The GLCM method is used to extract texture features from images of asphalt road damage. The research used eight images of asphalt road damage objects, five with heavy damage and three with light damage. The research results showed that the GLCM method was effective in classifying the level of damage to asphalt roads based on texture characteristics extracted from the images

**Keywords**: Asphalt road damage, GLCM, image classification.

## 1. Introduction

Highways are important infrastructure in people's lives today, whose function is to facilitate people's mobility needs [1] . Over time, roads will experience damage, whether due to natural factors or human factors [2] . Based on data from the West Sumatra Provincial Statistics Agency for 2020 to 2022, data on road damage in the city of Padang is obtained, as in the following table:

| Year | Damage Type | |
|---|---|---|
| | Light Damage | Heavy Damaged |
| 2020 | 0.2 % | 14.80 % |
| 2021 | 0.2 % | 15.00 % |
| 2023 | 1.7 % | 0.20 % |

Table I. Percentage of Road Damage in Padang City

cases. Image methods can be used for both mathematical calculations on pixel and geometric objects. Each image object has a value that can be calculated mathematically, thus showing different characteristics between one object and another. The distinguishing characteristics of each object can be determined based on color, texture, or shape [4] . One of the methods that will be used to identify asphalt road damage is the Gray Level Co-occurrence Matrix (GLCM). GLCM is a statistical method for forming features/characteristics that are not based on pixel values and pixel neighbor relationships (Kadir and Susanto, 2013) [5] .

The characteristic that will be used is texture, which is the difference between one object of asphalt road damage and another. Images of asphalt road damage have pixel patterns that differ from each other to determine the level of damage [6] .

The reference used in this research looks at several studies such as those conducted by Neneng in 2017. The research discusses the classification of types of meat based on image analysis of gray level co-occurence matrix (GLCM) texture and color [6] .

## 2. Research of Methodology

The research discussed is the GLCM feature extraction method for classifying asphalt road damage images [7] . The sample used is an image of asphalt road damage with the extension *.jpg, each image sample is processed and processed using the GLCM method to determine the data information in the image which can then be used for the next process and is expected to produce data output that can be used for the damage image data collection process. Pavement. Data collection for this research was carried out using several methods. observing directly on asphalt roads so that you can find directly the difference between damage to asphalt roads with light levels of damage and heavy damage to asphalt roads. reading literature, books, journals, papers and articles related to research methods on asphalt road damage. The application is used by inputting *.JPG images of asphalt road damage and then processing feature extraction using the GLCM method, then the data is saved in .xls format and can then be used for the classification process of asphalt road damage images.

# 3. RESULT

Shape and texture analysis can be used to design an object identification system. This application uses MATLAB programming to identify types of road damage that are lightly damaged or heavily damaged based on shape and texture analysis. Shape analysis was carried out using metric and eccentricity parameters, while texture analysis was carried out using the Gray Level Co-occurence Matrix (GLCM) method with contrast, correlation, energy and homogeneity parameters [8] .

## 1. Segmentation and feature extraction based on shape and texture analysis

Segmentation and feature extraction based on shape and texture analysis were carried out on eight object images consisting of 5 heavily damaged roads and 3 lightly damaged roads.

| No | Object Image | Information |
|---|---|---|
| 1 |  | Heavy Damaged |
| 2 |  | Light Damage |
| 3 |  | Light Damage |
| 4 |  | Heavy Damaged |
| 5 |  | Heavy Damaged |
| 6 |  | Light Damage |
| 7 |  | Heavy Damaged |
| 8 |  | Heavy Damaged |

Table 2. Object Image

From the eight object images, segmentation and feature extraction data were obtained based on shape and texture analysis. Database of shape and texture characteristics from extracted damaged road images. In the future, the extracted data can be used to identify damaged roads [9] .

| metrics | eccentricity | Contrast | Correlation | Energy | Homogeneous |
|---|---|---|---|---|---|
| 0.659512 | 0.796826 | 0.049372 | 0.95985 | 0.906815 | 0.994074 |
| 0.131913 | 0.765093 | 0.651135 | 0.874316 | 0.578602 | 0.92397 |
| 0.138725 | 0.921274 | 0.722272 | 0.931428 | 0.282618 | 0.875295 |
| 0.033765 | 0.978044 | 0.554629 | 0.893182 | 0.679652 | 0.930315 |
| 0.79782 | 0.899254 | 0.097557 | 0.957598 | 0.842925 | 0.991834 |
| 0.129775 | 0.972491 | 0.500546 | 0.801029 | 0.83162 | 0.958819 |
| 0.813109 | 0.665088 | 0.242219 | 0.938924 | 0.757645 | 0.957809 |

| | | | | | |
|---|---|---|---|---|---|
| 0.150286 | 0.87758 | 0.603553 | 0.893534 | 0.229969 | 0.849709 |

Table 3. Data Base of Shape and Texture Characteristics

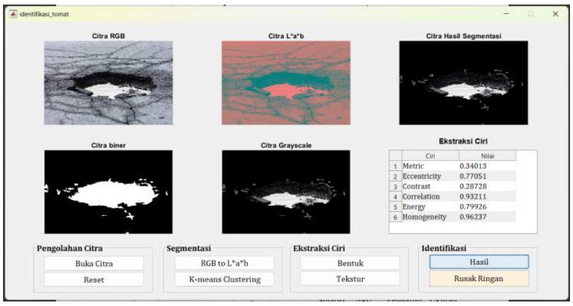



Figure 1. Image Identification Display

Based on the results of image identification using the application above, the results obtained are in accordance with the expected targets [12] .

Algorithm 2. Image Identification

```
gui_Singleton = 1;
gui_State = struct('gui_Name', mfilename,
...
'gui_Singleton', gui_Singleton, ...
'gui_OpeningFcn',
@identify_tomato_OpeningFcn, ...
'gui_OutputFcn', @identify_tomato_OutputFcn,
...
'gui_LayoutFcn', [] , ...
'gui_Callback', []);
if nargin && ischar(varargin{1})
gui_State.gui_Callback =
str2func(varargin{1});
end

if nargout
[varargout{1:nargout}] =
gui_mainfcn(gui_State, varargin{:});
else
gui_mainfcn(gui_State, varargin{:});
end

function identify_tomato_OpeningFcn(hObject,
eventdata, handles, varargin)

%identify_tomato
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
movegui(hObject,'center');

function varargout =
identify_tomato_OutputFcn(hObject,
eventdata, handles)

varargout{1} = handles.output;

function pushbutton6_Callback(hObject,
eventdata, handles)
load characteristic_database
feature_total = handles.character_total;

traits = zeros(1,6);
for i = 1:6
traits(i) = str2double(features_total{i,2});
end

[num,~] = size(database_features);

dist = zeros(1,num);
for n = 1:num
data_base = characteristic_database(n,:);
distance = sum((data_base-feature).^2).^0.5;
dist(n) = distance;
end
[~,id] = min(dist);

if isempty(id)
set(handles.edit1,'String','Unknown')
else
switch id
case 1
level = 'Heavily Damaged';
case 2
level = 'Slightly Damaged';
case 3
level = 'Slightly Damaged';
case 4
level = 'Heavily Damaged';
case 5
level = 'Heavily Damaged';
case 6
level = 'Slightly Damaged';
case 7
level = 'Heavily Damaged';
case 8
level = 'Heavily Damaged';
end
set(handles.edit1,'String',level)
end


function edit1_Callback(hObject, eventdata,
handles)

function edit1_CreateFcn(hObject, eventdata,
handles)
if ispc &&
isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end

function pushbutton4_Callback(hObject,
eventdata, handles)
Img_bw = handles.Img_bw;

axes(handles.axes4)
imshow(Img_bw)
title('Binary image');

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

shape_characteristics = cell(2,2);
shape_characteristic{1,1} = 'Metric';
shape_characteristic{2,1} = 'Eccentricity';
shape_characteristic{1,2} = num2str(metric);
shape_characteristic{2,2} =
num2str(eccentricity);

handles.shape_characteristics =
shape_characteristics;
guidedata(hObject, handles)

set(handles.text2,'String','Feature
Extraction')
set(handles.uitable1,'Data',shape_feature,'R
owName',1:2)

function pushbutton5_Callback(hObject,
eventdata, handles)
Img = handles.Img;
Img_bw = handles.Img_bw;
shape_characteristics =
handles.shape_characteristics;

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

axes(handles.axes5)
imshow(Img_gray)
title('Greyscale Image')

pixel_dist = 1;
GLCM = graycomatrix(Img_gray,'Offset',[0
pixel_dist; -pixel_dist pixel_dist; -
pixel_dist 0; -pixel_dist -pixel_dist]);
statistics =
graycoprops(GLCM,{'contrast','correlation','
```

```
energy','homogeneity'});
Contrast = mean(stats.Contrast);
Correlation = mean(stats.Correlation);
Energy = mean(stats.Energy);
Homogeneity = mean(stats.Homogeneity);

feature_total = cell(6,2);
total_features{1,1} = shape_features{1,1};
total_features{1,2} = shape_features{1,2};
total_features{2,1} = shape_features{2,1};
total_features{2,2} = shape_features{2,2};
feature_total{3,1} = 'Contrast';
feature_total{4,1} = 'Correlation';
feature_total{5,1} = 'Energy';
traits_total{6,1} = 'Homogeneity';
feature_total{3,2} = num2str(Contrast);
feature_total{4,2} = num2str(Correlation);
feature_total{5,2} = num2str(Energy);
traits_total{6,2} = num2str(Homogeneity);

handles.traits_total = traits_total;
guidedata(hObject, handles)

set(handles.text2,'String','Feature
Extraction')
set(handles.uitable1,'Data',feature_total,'R
owName',1:6)

function pushbutton2_Callback(hObject,
eventdata, handles)
Img = handles.Img;

cform = makecform('srgb2lab');
lab = applycform(Img,cform);
axes(handles.axes2)
imshow(lab)
title('L*a*b Image');

handles.lab = lab;
guidedata(hObject, handles)


function pushbutton3_Callback(hObject,
eventdata, handles)
Img = handles.Img;
lab = handles.lab;

ab = double(lab(:,:,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 = cells(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);

tomato = Img;
R = tomato(:,:,1);
G = tomato(:,:,2);
B = tomato(:,:,3);
R(~Img_bw) = 0;
G(~Img_bw) = 0;
B(~Img_bw) = 0;
tomato_rgb = cat(3,R,G,B);
axes(handles.axes3)
imshow(tomato_rgb)
```

```
title('Segmentation Result Image');

handles.Img_bw = Img_bw;
guidedata(hObject, handles)


function pushbutton1_Callback(hObject,
eventdata, handles)
[filename,pathname] = uigetfile('*.jpg');

if ~isequal(filename,0)
Img = imread(fullfile(pathname,filename));
axes(handles.axes1)
imshow(Img)
title('RGB Image')
else
returns
end

handles.Img = Img;
guidedata(hObject, handles)

function pushbutton7_Callback(hObject,
eventdata, handles)
axes(handles.axes1)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])

axes(handles.axes2)
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.text2,'String',[])
set(handles.uitable1,'Data',[])
set(handles.edit1,'String',[])
```

# 4. CONCLUSION

1. Highways are important infrastructure in people's lives and facilitate people's mobility.
2. Highways experience damage over time due to natural and human factors.
3. This research aims to recognize asphalt road damage objects using image processing methods with a focus on texture using GLCM.
4. The GLCM method is used to extract texture characteristics from images of asphalt road damage to classify damage into light damage and heavy damage.
5. This research was inspired by previous research which used the GLCM method to classify other objects based on texture and color.
6. This research uses image samples of asphalt road damage in *.jpg format and the MATLAB application for processing.

# References

[1] B. Sasmito, BH Setiadji, and R. Isnanto, "Road Damage Detection Using Deep Learning

Image Processing in Semarang City," *Engineering* , vol. 44, no. 1, pp. 7–14, 2023, doi: 10.14710/teknik.v44i1.51908.

[2] Y. Arum Sari and C. Dewi, "Image Retrieval System for Asphalt Potholes Based on Damage Level Using Gray Level Co-occurrence Matrix Feature Extraction Identifying Quality of Patchouli Leaves through Its Leave Image Using Learning Vector Quantization View project Smart Wh, ” no. July, 2018, [Online]. Available: https://www.researchgate.net/publication/326571615

[3] Neneng and Y. Fernando, "Classification of Meat Types Based on Gray Level Co-Occurrence Matrices (Glcm) and Color Texture Image Analysis," *Semin. Nas. Science and Technology. 2017* , no. November, pp. 1–7, 2017.

[4] RA Surya, A. Fadlil, and A. Yudhana, "Feature Extraction of the Gray Level Co-Occurrence Matrix (GLCM) Method and Gabor Filter for Pekalongan Batik Image Classification," *J. Inform. Developer IT (JPIT, Vol. 02, No. 02, July 2017* , vol. 02, no. 02, pp. 23–26, 2017.

[5] N. Neneng, K. Adi, and R. Isnanto, "Support Vector Machine for Image Classification of Meat Types Based on Texture Using Gray Level Co- Occurrence Matrices (GLCM) Feature Extraction," *J. Sist. Inf. Business* , vol. 6, no. 1, p. 1, 2016, doi: 10.21456/vol6iss1pp1-10.

[6] N. Neneng, AS Puspaningrum, and AA Aldino, "Comparison of Meat Type Classification Results Using Texture Characteristic Extraction of Gray Level Co-occurrence Matrices (GLCM) and Local Binary Pattern (LBP)," *Smatika J.* , vol. 11, no. 01, pp. 48–52, 2021, doi: 10.32664/smatica.v11i01.572.

[7] S. Enggari, A. Ramadhanu, and H. Marfalino, "Improved Digital Image Processing in Describing Fungal Plants Using Color Segmentation, Edge and Contour Detection," *J. Teknol. And Sis. Inf. Business* , vol. 4, no. 1, pp. 70–75, 2022, doi: 10.47233/jteksis.v4i1.358.

[8] FM Nejad, FZ Motekhases, H. Zakeri, L.-J. Wang, and JM Monaghan, “An Image Processing Approach to Asphalt Concrete Feature Extraction,” *J. Ind. Intel. Inf.* , vol. 3, no. 1, 2014, doi: 10.12720/jiii.3.1.54-60.

[9] Johan Wahyudi and Ihdahubbi Maulida, "Recognition of Traditional Fabric Image Patterns Using Glcm and Knn," *J. Teknol. Inf. Univ. Gastric Mangkurat* , vol. 4, no. 2, pp. 43–48, 2019, doi: 10.20527/jtiulm.v4i2.37.

[10] M. Vadood, MS Johari, and AR Rahaei, “Introducing a simple method to determine aggregate gradation of hot mix asphalt using image processing,” *Int. J. Pavement Eng.* , vol. 15, no. 2, pp. 142–150, 2014, doi: 10.1080/10298436.2013.786076.

[11] ASA Hamid, "Use of Digital Image Processing with Edge Detection Algorithms in Identifying Damage to Road Contours," *Semin. Nas. Science and Technology. Apply. V* , no. June, pp. 149–154, 2017.

[12] MRS Zawad, MFS Zawad, MA Rahman, and SN Priyom, “A comparative review of image processing based crack detection techniques on civil engineering structures,” *J. Soft Comput. Civ. Eng.* , vol. 5, no. 3, pp. 58–74, 2021, doi: 10.22115/SCCE.2021.287729.1325.