A Thesis/Project/Dissertation Report

on

IMAGE SEGMENTATION

Submitted in partial fulfillment of the
requirement for the award of the degree of

BACHELOR OF ENGINEERING IN COMPUTER SCIENCE &ENGINEERING

GALGOTIAS UNIVERSITY
(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Mr. Rajkamal Kishor Gupta.
Assistant Professor

Submitted By

Sweta Singh -20SCSE1010305

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
MONTH, YEAR

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **"Image Segmentation"** in partial fulfillment of the requirements for the award of the **Bachelor of engineering in computer science** submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of August 2021Year to December 2021, under the supervision of **Mr. Rajkamal Kishor Gupta, Assistant Professor** , Department of Computer Science and Engineering, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by us for the award of any other degree of this or any other places.

Sweta Singh (20SCSE1010305)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Mr. Rajkamal Kishor Gupta.
Assistant Professor

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of Sweta Singh (20SCSE1010305) has been held on 21/12/2021-24/12/2021 and our work is recommended for the award of Bachelor of Engineering in Computer Science.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date: 22 December, 2021
Place: Greater Noida

Acknowledgement

One of the pleasant aspects of preparing a project report is the opportunity to thank those who have contributed to make thisproject possible .
We are extremely thankful to **Mr. Rajkamal Kishor Gupta** sir whose active interest in the project &insight helped us to formulate, redefine implement our approach to the project Furthermore.
I would also like to acknowledge with much appreciation the crucial role of my teammates and all those who have guided and helped.

Sweta Singh
20SCSE1010096

Abstract

Image segmentation is an important image processing step, and it is used everywhere if we want to analyze what is inside the image. Image segmentation, basically provide the meaningful objects of the image. This paper represents the various image segmentation techniques that could be used in the segmentation algorithm. Whenever we work with the image in any application, initial step is to segment the image in order to solve its complexity. The segmentation of images is the basic thing for understanding the images. It is used in the Image processing applications, Computer vision etc. In this paper, two categories are emphasized: Edge based and region based segmentation, which further includes their respective techniques.
The image segmentation problem can be stated as the division of an image into regions that separate different objects from each other, and from the background. The problem of segmenting a single image in a completely bottom-up fashion, that is, without prior knowledge of what specific objects will be found in the scene. This means that the segmentation algorithms can not use intensity, colour, texture, or feature descriptors that are specific to a known object, or object class to produce the segmentation.

Keywords: Image segmentation, Edge-Based Technique, Region Based technique.

There are some techniques can often be used to successfully deal with these problems. For some images it is not possible to set segmentation process parameters, such as a threshold value, so that all the objects of interest are extracted from the background or each other without over segmenting the data.

The tools used for the design and methodology include Algorithm and Flow chart. From the flow chart below, the steps carried out by the system administrator are as follow:
Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture.

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image (see edge detection). Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity,
or texture.

There are still some things we can do for future works. At first, we will improve the stability of program. We want to modify the code of function to let programs more stable. Secondly, it’s a chance to get a better adaptive method for image segmentation even that we have an adaptive method already. The third one is to generate the post-processing mechanism for region merging. We can write a code about merging groups with the same texture into a single group.

Contents

Title		Page No.
Candidates Declaration		**I**
Acknowledgement		**II**
Abstract		**III**
Contents		**IV**
List of Table		**V**
List of Figures		**VI**
Acronyms		**VII**
Chapter 1	**Introduction**	**10**
	1.1 Introduction	
	1.2 Formulation of Problem	
	1.2.1 Tool and Technology Used	
Chapter 2	**Literature Survey**	**11-12**
	2.1 Methedology	**12**
Chapter 3	**Functionality/Working of Project**	**13-26**
	3.1 Design	**13**
	3.2 Data flow Diagram	**14**
	3.3 UML Diagram	**15**
	3.4 Code	**16-26**
Chapter 4	**Results and Discussion**	**27**
Chapter 5	**Conclusion and Future Scope**	**28**
	5.1 Conclusion	
	5.2 Future Scope	
	Reference	**29-30**

List of Table

S.No.	Caption	Page No.
1	Design	13

List of Figures

S.No.	Title	Page No.
1	Design	13
2	Data Flow Diagram	14
3	UML Diagram	15

Acronyms

B.Tech.	Bachelor of Technology
M.Tech.	Master of Technology
BCA	Bachelor of Computer Applications
MCA	Master of Computer Applications
B.Sc. (CS)	Bachelor of Science in Computer Science
M.Sc. (CS)	Master of Science in Computer Science
SCSE	School of Computing Science and Engineering

CHAPTER-1
Introduction

Image segmentation is the process of segmenting the image into various segments that could be used for the further applications such as Image understanding model , Robotics , Image analysis, Medical diagnosis, etc. Image segmentation is the process of partitioning an image into multiple segments, so as to change the representation of an image into something that is more meaningful and easier to analyse. Image segmentation means assigning a label to each pixel in the image such that pixels with same labels share common visual characteristics. It makes an image easier to analyse in the image processing tasks. There are many different techniques available to perform image segmentation. My motive is to implement almost the same concept as we humans try to implement, while understanding the image which we visualize. In human vision, the complex image is immediately segmented into the simple objects on the basis of color, texture, patterns, shapes, etc. This same thing is constructed with the help of the image segmentation techniques in the computer vision system. We could segment the digital image on the basis of these features, so that the task of understanding of image could be done in a simple and humanly way. On the basis of these defined features, there are several image segmentation techniques which would provide the segmented results. Image segmentation refers to the process of partitioning a digital image into multiple segments set of pixels , pixels in a region are similar according to some homogeneity criteria such as color , intensity or texture, so as to locate and identify objects and boundaries in an image. The choice of a segmentation technique over another and the level of segmentation are decided by the particular type of image and characteristics of the problem being considered.

1.2 Formulation of Problem

There are some techniques can often be used to successfully deal with these problems. For some images it is not possible to set segmentation process parameters, such as a threshold value, so that all the objects of interest are extracted from the background or each other without over segmenting the data.

1.2.1 Tool and Technology Used

The tools used for the design and methodology include Algorithm and Flow chart. From the flow chart below, the steps carried out by the system administrator are as follow:
Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture.
The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image (see edge detection). Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture.

CHAPTER-2
Literature Survey

Sharmila and kirubakaran developed a method to extract human gait feature automatically even in low resolution. They applied two techniques for silhouette extraction called Image Based Gait Recognition and Formula Based Gait Recognition in free motion features from video sequence. The Image based gait technique can deal with clothing, lighting, segmentation and tracking.

The formula based approach deals with height and stride parameters of walking gait. They used their own video frame and applied for Identification purposes. Bo and Yumei proposed to extract the silhouette and improved the recognition rate. They applied Principle Component Analysis (PCA) with LDA to process spatial template. They are able to make silhouette from the original image after applying few steps of silhouette extraction. For experimental purposes, they used NLPR database and achieved 87% recognition rate.

Honggui and Xingguo proposed dimension reduction technique for gait images. They found that Gaussian technique is better than normal background subtraction technique. They also applied gait alignment technique. The different styles of gait cycle have been applied and finally FFT module of 1D LLE present for gait recognition purposes. They applied CMU MoBo gait database and achieved 92% recognition rate in rank 5 .

Xi et al. explained about extraction of silhouette from video sequence. They have been able to get perfect silhouette after applying few steps such as background update, shadow removal, morphological filter and others parts. They have described robust algorithm for extracting the human silhouette.

Qiong et al. proposed gait recognition based on PCA and LDA. PCA is mainly used for dimensional reduction technique and LDA is performed to optimize the pattern class. For the experiment, they used their own database and they achieved better recognition rate from PCA compared to LDA .

In this paper, we will apply PCA technique and use CMU MoBo database for person identification in different walking style.

2.1 Methodology

- **Problem formulation** :

The image segmentation problem can be started as the division of an image into regions that separate different objects from each other, and from the background. The problem of segmenting a single image in a completely bottom up fashioned that is without prior knowledge of what specific objects will be found in the scene. This means that that segmentation algorithms cannot use intensity,colour,texture or feature descriptors that are specific to a known object, or object class to produce the segmentation.

- **Proposed solution :**

There are some techniques can often be used to successfully deal with these problems. For some images it is not possible to set segmentation process parameters, such as a threshold value, so that all the objects of interest are extracted from the background or each other without over segmenting the data.

- **Tools and technology used :**

The tools used for the design and methodology include Algorithm and Flow chart. From the flow chart below, the steps carried out by the system administrator are as follow:
Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture.

- **Result and output :**

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image (see edge detection). Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture.

Chapter 3
Functionality/Working of Project

Image segmentation is a large aspect of computer vision and has many applications in numerous industries. Some of the notable areas where image segmentation is used profusely are:

- **Face Recognition**

The facial acknowledgment innovation present in your iPhone and progressed security frameworks utilizes picture division to distinguish your face. It should have the option to distinguish the special highlights of your face with the goal that any undesirable party can't get to your telephone or framework.

- **Number Plate Identification**

Many traffic signals and cameras utilize number plate recognizable proof to accuse fines and help of searches. Number plate distinguishing proof innovation permits a traffic framework to perceive a vehicle and get its possession related data. It utilizes picture division to isolate a number plate and its data from the remainder of the articles present in its vision. This innovation has improved on the fining system significantly for state run administrations.

3.1 DESIGN

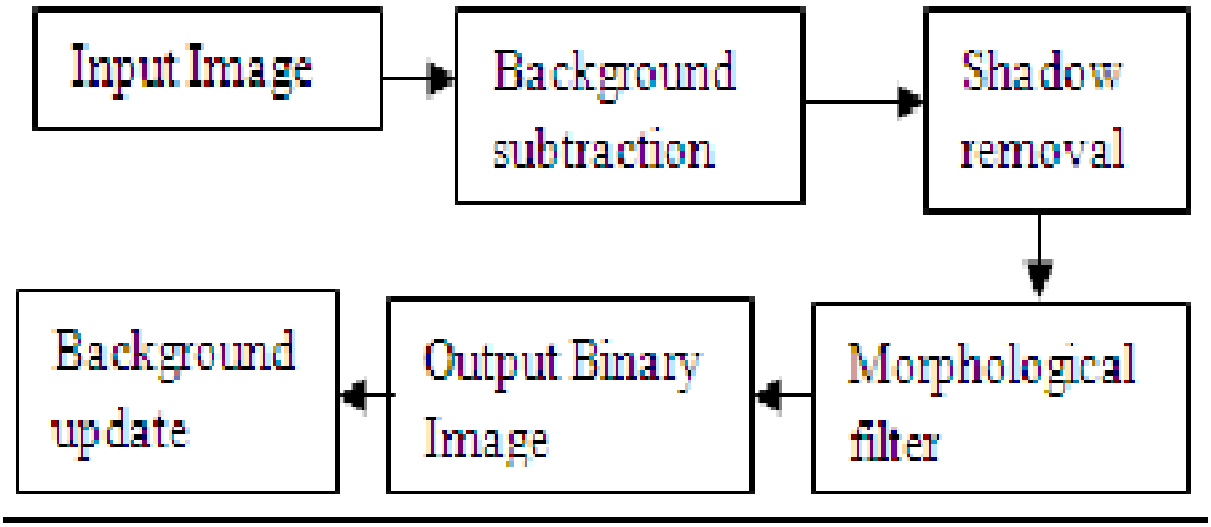

FIG: DESIGN

3.2 Data Flow Diagram

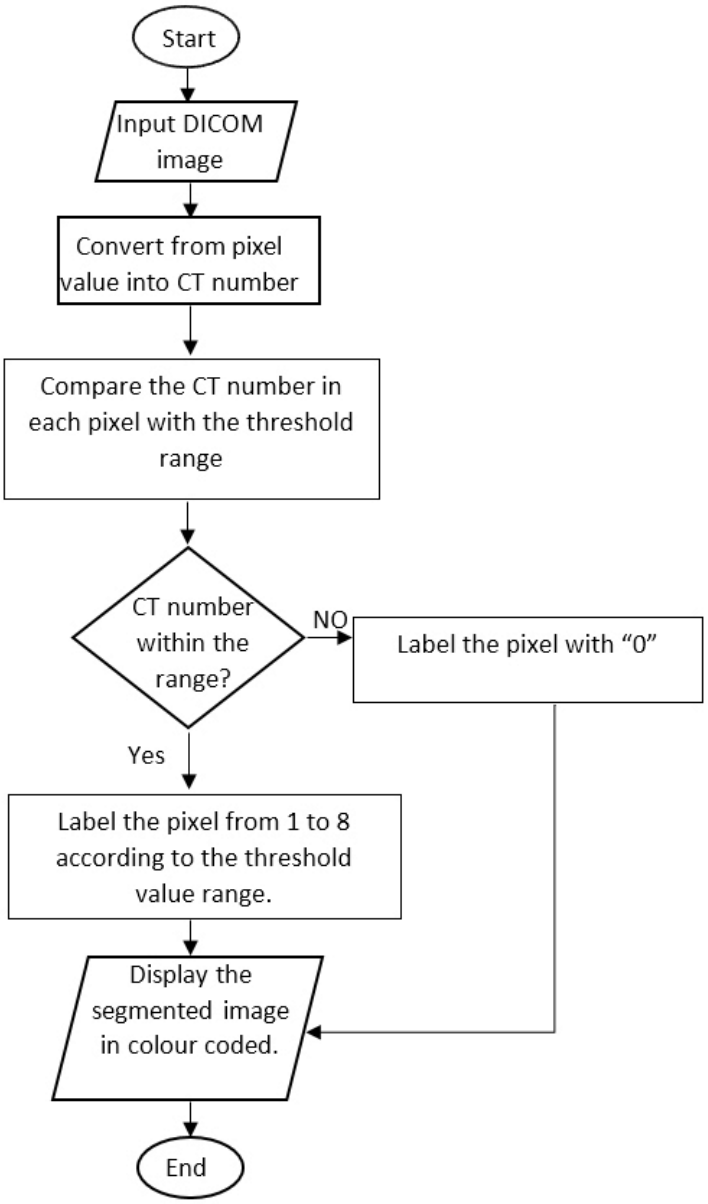

FIG: DATA FLOW DIAGRAM

3.3 **UML DIAGRAM**

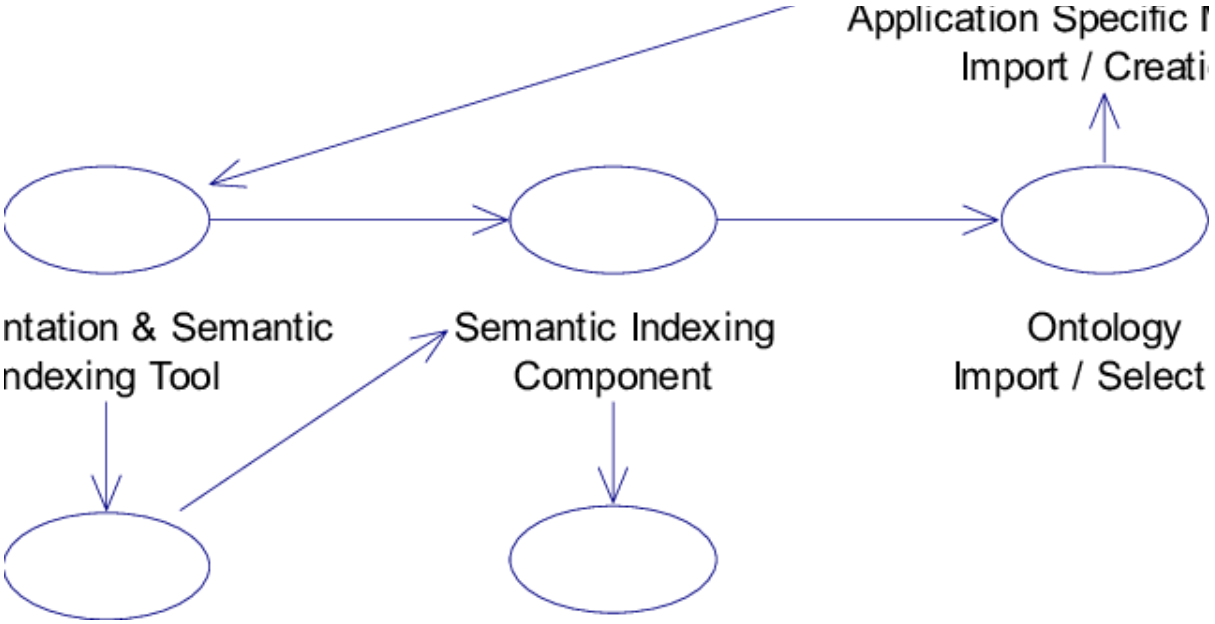

FIG: UML DIAGRAM

3.4 CODE

APPENDIX A

```
%% Segmenting 10_27a.tif using Otsu's Method
project_root_dir = strcat(getenv('HOME'),
'/PIMAGES');
inclass_image_root_dir = strcat(project_root_dir, '/inclass');
extras_image_root_dir = strcat(project_root_dir, '/extras');
output_image_root_dir = strcat(project_root_dir, '/output/');
%%Read the Image
I = imread(strcat(inclass_image_root_dir,'/10_27a.tif'));

% Calculate a threshold using graythresh.
% The threshold is normalized to the range [0, 1].
level = graythresh(I);
% Convert the image into a binary image using the
threshold. Output = imbinarize(I,level);
%% Write the file to the output folder
imshow(Output)
fullFileName = fullfile(output_image_root_dir,
'10_27a_op.jpg'); imwrite(Output, fullFileName);
```

APPENDIX B1

```
%% Segmenting 10_27d.tif using Split and Segment Method
project_root_dir = strcat(getenv('HOME'), '/PIMAGES');
inclass_image_root_dir = strcat(project_root_dir, '/inclass');
extras_image_root_dir = strcat(project_root_dir, '/extras');
output_image_root_dir = strcat(project_root_dir, '/output/');
%% Read the image
I = imread(strcat(inclass_image_root_dir, '/10_27d.tif'));
%% Measure the
size [r,c]=size(I);
%% Partition and
Threshold OUT = [ ];
br=15
0;
bc=12
7;
bseg=
block(I);
kk=0;
for i=1:2
    for j=1:2
        z=kk+j;
        k=graythresh(bseg(:,:,z));
        m=
        im2bw(bseg(:,:,z),k);
```

```
                subplot(2,2,z)
                imshow(m)
                if i==1 &&
                    j==1 o1 =
                    m;
                else if i==1 &&
                        j==2 o2
                        =m;
                    else if i==2 &&
                            j==1 o3
                            =m;
                        else if i==2 && j==2
                                o4 = m;
                            end
                        ; end;
                    end
                ; end;

            end
            kk=kk+(2);
        end
        %Stitch the  image
        out = [o1 o2; o3 o4];
        figure
        imshow(out)
        %Create a disk shaped structuring element (SE) of size
        2 se = strel('disk',2);
        %Morphologically open the image with using
        the SE out_enhanced = imopen(out,se);
        figure
        imshow(out_enhan
        ced)
        %% Write the image to output folder
        fullFileName = fullfile(output_image_root_dir, '10_27d_op.jpg');

imwrite(out_enhanced, fullFileName);

function [outp]= block(I)

        kk=0;
        for i=1:2
            for j=1:2
                feature(:,:,kk+j)=(I((127*(i-1)+1:127*(i-1)+127),(150*(j-1)+1:150*(j-1)+150)));
            end
        end kk=kk+(2
            );
            outp=feat
```

```
outp=feature
; end
```

APPENDix B2

```
%% Segmenting Group of Disjoint Coins
project_root_dir = strcat(getenv('HOME'),
'/PIMAGES');
inclass_image_root_dir = strcat(project_root_dir, '/inclass');
extras_image_root_dir = strcat(project_root_dir, '/extras');
output_image_root_dir = strcat(project_root_dir, '/output/');
%% Read the image
I = imread(strcat(extras_image_root_dir,'/Group_of_Disjoint_Coins.png'));

% Calculate a threshold using graythresh.
% The threshold is normalized to the range [0, 1].
level = graythresh(I);
% Convert the image into a binary image using the
threshold. Output = imbinarize(I,level);
se = strel('disk',2);
%Morphologically close the image with using the
SE out_enhanced = imclose(Output,se);
figure
imshow(out_enhan
ced)
%% Write the file to the output folder
fullFileName = fullfile(output_image_root_dir,
'Group_of_Disjoint_Coins_OP.png'); imwrite(out_enhanced, fullFileName);
```

APPENDix B3

```
%% Segmenting Group of Coins 1
project_root_dir = strcat(getenv('HOME'), '/PIMAGES');
inclass_image_root_dir = strcat(project_root_dir, '/inclass');
extras_image_root_dir = strcat(project_root_dir, '/extras');
output_image_root_dir = strcat(project_root_dir, '/output/');
%Read the image
I1 = imread(strcat(extras_image_root_dir,'/group_of_coins_1.jpg'));
%Invert the  image
I=
imcomplement(I1);
%Decrease the
brightness I = I-45;
%Calculate
Threshold level =
graythresh(I);
%Threshold the
image BW =
im2bw(I,level);
```

```
%Create a line shaped structuring
element se = strel('line',4,80);
%Erode the image with the
SE erodedBW =
imerode(BW,se);
%Fill holes
BW2 = imfill(erodedBW,'holes');
%Write the image to a file in the outout folder
imshow(BW2)
fullFileName =
fullfile(output_image_root_dir,'Group_of_Coins_1_OP.png');
imwrite(BW2, fullFileName);
```

APPENDix C1

```
%% Segmenting Jet Fighter Swarm
project_root_dir = strcat(getenv('HOME'), '/PIMAGES');
inclass_image_root_dir = strcat(project_root_dir, '/inclass');
```

```
extras_image_root_dir = strcat(project_root_dir, '/extras');
output_image_root_dir = strcat(project_root_dir, '/output/');
%% Read the image
I= imread(strcat(extras_image_root_dir, '/jet_fighter_swarm.png'));
%Generate Background
BG = imgaussfilt(I,100);
%Get
foreground FG
= BG - I;
%Get threshold
level = graythresh(FG);
%Threshold the
image FG = FG -
10;
BW = imbinarize(FG,level);
%Create a disk shaped structuring
element se = strel('disk',2);
%Morphologically close the image with the
SE close_BW = imclose(BW,se);
%Invert
close_BW =
imcomplement(close_BW);
imshow(close_BW)
%Write the image to a file in the outout folder
fullFileName =
fullfile(output_image_root_dir,'Swarm_of_Jets_OP.png');
imwrite(close_BW, fullFileName);
```

APPENDix C2

```
%% Segmenting Sonnet for Lena
project_root_dir = strcat(getenv('HOME'), '/PIMAGES');
inclass_image_root_dir = strcat(project_root_dir, '/inclass');
extras_image_root_dir = strcat(project_root_dir, '/extras');
output_image_root_dir = strcat(project_root_dir, '/output/');
% Read the image
[im1,map]=imread(strcat(inclass_image_root_dir,'/son1.gif')
); if ~isempty(map)
    im1 = ind2rgb(im1,map);
end
%Convert matrix to grayscale
image im1=mat2gray(im1);
%Generate background
bwim1=imfilter(im1,fspecial('average',11),'replicate');
%Obtain
Foreground
sIM=bwim1-im1-
0.025;
%Threshold
bwim2 = im2bw(sIM,0);
%Invert
bwim2=imcomplement(b
wim2);
```

```
%Write file to the output
folder imshow(bwim2)
fullFileName =
fullfile(output_image_root_dir,'son1_OP.png');
imwrite(bwim2, fullFileName);
```

APPENDIX D1

```
%Segmenting papir using method 1
project_root_dir = strcat(getenv('HOME'), '/PIMAGES');
inclass_image_root_dir = strcat(project_root_dir, '/inclass');
extras_image_root_dir = strcat(project_root_dir, '/extras');
output_image_root_dir = strcat(project_root_dir, '/output/');
%Read the image
I = imread(strcat(inclass_image_root_dir,'/papir.jpg'));

%Roberts edge
detection R=
edge(I,'Roberts');
%Uncomment if intermediate output is desired
%fullFileName = fullfile(output_image_root_dir, 'papir_2.jpg');
%imwrite(R, fullFileName);

%Remove letters
BW2 = bwareaopen(R, 40);
```

```
%Uncomment if intermediate output is desired
%fullFileName = fullfile(output_image_root_dir, 'papir_3.jpg');
%imwrite(BW2, fullFileName);

%Create line shaped structural
element se= strel('line',16,120);
%Morphologically close the image to close open boundaties of the
hair BW2=imclose(BW2,se);
%Uncomment if intermediate output is desired
%fullFileName = fullfile(output_image_root_dir, 'papir_4.jpg');
%imwrite(BW2, fullFileName);

%Create line shaped structural
element se = strel('line',11,90);
%Morphologically dilate the image to thicken the
hair m = imdilate(BW2,se);
%Uncomment if intermediate output is desired
%fullFileName = fullfile(output_image_root_dir, 'papir_dilate.jpg');
%imwrite(m, fullFileName);

%Perform adaptive thresholding on the ooriginal
image T =
adaptthresh(I,0.45,'ForegroundPolarity','dark'); c =
imbinarize(I,T);

%Add the thresholded image to the image with just the hair to
remove Z=c+m;
%Write the file to output folder
imshow(Z);
fullFileName = fullfile(output_image_root_dir,
'papir_OP_M1.jpg'); imwrite(Z, fullFileName);
```

APPENDix D2

```
import cv2
import numpy as np

# Read in image, grayscale, and Otsu's threshold
image =
cv2.imread('C:/PIMAGES/text_with_lines.tif') gray
= cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
thresh = cv2.threshold(gray, 0, 255,cv2.THRESH_BINARY_INV + cv2.THRESH_OTSU)[1]

# Create diagonal kerneld
kernel = np.array([[0, 0, 1],
                   [0, 1, 0],
                   [1, 0, 0]], dtype=np.uint8)
opening = cv2.morphologyEx(thresh, cv2.MORPH_OPEN, kernel, iterations=1)

# Find contours and filter using contour area to remove noise
```

```
cnts = cv2.findContours(opening, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE) cnts = cnts[0] if len(cnts) == 2 else
cnts[1]
for c in cnts:
    area =
    cv2.contourArea(c) if
    area < 200:
        cv2.drawContours(opening, [c], -1, (0,0,0), -1)

# Bitwise-xor with original image
opening = cv2.merge([opening, opening,
opening]) result = cv2.bitwise_xor(image,
opening)

cv2.imshow('thresh', thresh)
cv2.imshow('opening',
opening) cv2.imshow('result',
result) cv2.waitKey()
```

APPENDIX E

```
%% Segmenting Nuts and Bolts | Requires DIP IMAGE
project_root_dir = strcat(getenv('HOME'), '/PIMAGES');
inclass_image_root_dir = strcat(project_root_dir, '/inclass');
extras_image_root_dir = strcat(project_root_dir, '/extras');
output_image_root_dir = strcat(project_root_dir, '/output/');
dipstart;
```

```
%% Read the image as a DIP_IMAGE object, simply replace nuts_bolts1
with nuts_bolts2 tosegment the other image
a = readim(strcat(extras_image_root_dir,'/nuts_bolts1.png'));
%Threshold the image
[b,thres] = threshold(a,'minerror',Inf);

%Measure the size
msr = measure(c,[],'Size',[],Inf,0,0);
%Determine the max
and min bolt and nut
sizes bolt_min =
round(min(msr)+max(
msr)/2); bolt_max =
max(msr);
for i=1:m
nut_max = max(nuts);
%Label the bolts
d = label(c,Inf,bolt_min,bolt_max);
%Threshold the bolts
[e,thres] = threshold(d,'isodata',bolt_max); bolts = e;
%Overlay image showing
segmented nuts and bolts
overlay(bolts,nuts)
```

APPENDIX F

```
from matplotlib import pyplot as plt

img =
cv2.imread('C:/PIMA
GES/coins.jpg') gray
=
cv2.cvtColor(img,cv2
.COLOR_BGR2GRA
Y)
ret, thresh =
cv2.threshold(gray,0,255,cv2.THRESH_BINARY_IN
V+cv2.THRESH_OTSU) cv2.imshow('Thresholded
Image',thresh)

# noise removal
kernel = np.ones((3,3),np.uint8)
opening =
cv2.morphologyEx(thresh,cv2.MORPH_OPEN,ke
rnel, iterations = 2)
cv2.imshow('Opening',opening)

# sure background area
sure_bg =
cv2.dilate(opening,kernel,it
erations=3)
cv2.imshow('SURE
BG',sure_bg)
```

```
# Finding sure foreground area
dist_transform =
cv2.distanceTransform(opening,cv2.DI
ST_L2,5) cv2.imshow('Distance
Transform',dist_transform)
ret, sure_fg =
cv2.threshold(dist_transform,0.7*dist_transform.max
(),255,0) cv2.imshow('SURE FG',sure_fg)

#
Findi
ng
unkn
own
regio
n
sure_
fg =
np.ui
nt8(s
ure_f
g)
unknown =

cv2.subtract(sure

_bg,sure_fg)

cv2.imshow('UN

KNOWN',unkno

wn)

# Marker labelling
ret, markers = cv2.connectedComponents(sure_fg)
# Add one to all labels so that sure
background is not 0, but 1 markers =
markers+1
# Now, mark the
region of unknown
with zero
markers[unknown==
255] = 0
markers =
cv2.watershed
(img,markers)
img[markers
```

CHAPTER-4
RESULTS

Image segmentation is a process of dividing an image into its constituent homogeneous regions to extract data from the attributes of the image. As a result, a good segmentation should result in regions in which the image elements should have uniform properties in terms of brightness, color or texture etc. Though the image is to be portioned into regions, the considerable changes within the regions should be observable visually. The measurement of quality of segmentation is that the elements of the same region should be similar and should have clear difference between elements of the other regions.The segmentation process can be divided into various category based on the parameter selected for segmentation like pixel intensity, homogeneity, discontinuity, cluster data, topology etc. Each approach has its own advantages and disadvantages. The result obtained using one approach may not be the same as compared with other approach. Methods that are specialized to particular applications can often achieve better performance and this selection of an appropriate approach to a segmentation problem can be a difficult dilemma. Basically the segmentation can be semi-interactive or fully automatic. The algorithms developed for segmentation lies in either of this category. With the major difficulty of ill-posed nature of segmentation it is hard to obtain single answer for segmentation of given image as the interpretation varies from individual approaches. In some cases manual interaction to segment the image may be error-prone (for example, in case of seed selection) while the fully automated approach can give error output (for example in case of watershed segmentation)and in some cases interactive methods can be laborious and time consuming. So a single approach to segment all variety of images may be practical unachievable. The prior knowledge on the image can give better results and gives user the choice to decide proper method to segment the image.

CHAPTER-5
CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION

In this paper, a study based representation of different segmentation approaches is defined. Throughout this study of the various techniques, we concluded out various facts:

- First, the image segmentation is the crucial part of the image understanding/image processing model.
- Second, the segmentation technique of the image could be used as per the required application or the usage as image is segmented on the basis of different features.
- Third, the segmentation techniques are broadly categorized on the basis of detection of discontinuity and similarity of the image.
- Fourth, Opting a single technique or method would not provide better optimized results.

5.2 FUTURE SCOPE

In our future work, we will use the watershed technique along with the edge detector operator (1st derivative) to get the improved technique for the segmentation purpose. It would help to detect the segments in a quite simple and enhanced way.

There are still some things we can do for future works. At first, we will improve the stability of program. We want to modify the code of function to let programs more stable. Secondly, it's a chance to get a better adaptive method for image segmentation even that we have an adaptive method already. The third one is to generate the post-processing mechanism for region merging. We can write a code about merging groups with the same texture into a single group.

REFERENCES

1. Haralick R M, Shapiro L G. Image segmentation techniques[J]. Computer vision, graphics, and image processing, 1985, 29(1): 100-132.

2. Marr D, Hildreth E. Theory of edge detection[J]. Proceedings of the Royal Society of London B: Biolog- ical Sciences, 1980, 207(1167): 187-217.

3. Wu Z, Leahy R. An optimal graph theoretic ap- proach to data clustering: Theory and its application to image segmentation[J]. IEEE transactions on pattern analysis and machine intelligence, 1993, 15(11): 1101- 1113.

4. Lin D, Dai J, Jia J, et al. Scribblesup: Scribble-su- pervised convolutional networks for semantic segmen- tation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 3159- 3167.

5. Davis L S, Rosenfeld A, Weszka J S. Region ex- traction by averaging and thresholding[J]. IEEE Trans- actions on Systems, Man, and Cybernetics, 1975 (3): 383-388.

[7]§fi, $ƒ#, 晓'', 等. ƒ½¤*ëfl Otsu 的 y G+ 研y[J]. ?6fl‡¼, 2016, 42(11): 255-260,266.

[8] Kohler R. A segmentation system based on thresh- olding[J]. Computer Graphics and Image Processing, 1981, 15(4): 319-338.

[9] ═. ƒE 的 y G+ y[D]. 的研¿D$,2009.DOI:10.7666/d.y1576181.

[10] Pal N R, Pal S K. A review on image segmentation techniques[J]. Pattern recognition, 1993, 26(9): 1277-
1294.

[11] Wani M A, Batchelor B G. Edge-region-based segmentation of range images[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1994, 16(3): 314-319.

[12] Fan W. Color image segmentation algorithm based on region growth[J]. Jisuanji Gongcheng/ Com- puter Engineering, 2010, 36(13).

[13] Angelina S, Suresh L P, Veni S H K. Image seg- mentation based on genetic algorithm for region growth and region merging[C]//Computing, Electron- ics and Electrical Technologies (ICCEET), 2012 Inter- national Conference on. IEEE, 2012: 970-974.

[14] Ugarriza L G, Saber E, Vantaram S R, et al. Auto- matic image segmentation by dynamic region growth and multiresolution merging[J]. IEEE transactions on image

processing, 2009, 18(10): 2275-2288.

[15] Davis L S. A survey of edge detection tech- niques[J]. Computer graphics and image processing, 1975, 4(3): 248-270.

[16] Senthilkumaran N, Rajesh R. Edge detection techniques for image segmentation–a survey of soft computing approaches[J]. International journal of re- cent trends in engineering, 2009, 1(2): 250-254.

[17] Kundu M K, Pal S K. Thresholding for edge de- tection using human psychovisual phenomena[J]. Pat- tern Recognition Letters, 1986, 4(6): 433-441.

6. Haddon J F. Generalised threshold selection for

[18] edge detection[J]. Pattern Recognition, 1988, 21(3): 195-203.

[19] @晓¿. ƒ<7yfl的 y 6+研y
[D]. -‡¼$,2012.

[20] Sulaiman S N, Isa N A M. Adaptive fuzzy-K- means clustering algorithm for image segmentation[J]. IEEE Transactions on Consumer Electronics, 2010, 56(4).

[21] Kumar S, Singh D. Texture Feature Extraction to Colorize Gray Images[J]. International Journal of Computer Applications, 2013, 63(17).

[22]Chuang K S, Tzeng H L, Chen S, et al. Fuzzy c- means clustering with spatial information for image segmentation[J]. computerized medical imaging and graphics, 2006, 30(1): 9-15.

[23]Celebi M E, Kingravi H A, Vela P A. A compara- tive study of efficient initialization methods for the k- means clustering algorithm[J]. Expert Systems with Applications, 2013, 40(1): 200-210.

[24] Zhang L, Gao Y, Xia Y, et al. Representative dis- covery of structure cues for weakly-supervised image segmentation[J]. IEEE Transactions on Multimedia, 2014, 16(2): 470-479.

[25] Chen L C, Papandreou G, Kokkinos I, et al. Deeplab: Semantic image segmentation with deep con- volutional nets, atrous convolution, and fully con- nected crfs[J]. arXiv preprint arXiv:1606.00915, 2016.