 

**School of Information Science and Engineering**

**2019－2020 school year**

Digital Image Processing Experiment Report

Course Name： Digital Image Processing

Title of Experiment： Morphological Image Processing

Major and Class Class 3, Communication Engineering

Student ID 201600121116

Studeng Name 张子超

Date 17-05-2019

# 1． Objectives:

1. Study the frequently used algorithms of binary morphological image processing.

2. Master the usage of basic operation functions for morphological image processing.

3. Know the fundamental application of morphological.

# 2． Experiment Content:

1.Realize the fundamental morphological processing for binary images(erosion, dilation, opening and closing) by programming. Filter the image object by choosing different structure element.

2. Implement the noise smoothing and image edge extraction for grayscale images with morphological operation.

# 3． Experiment Steps:

## 1. Morphological transform for a binary image



structw =

0 0 1 0 0

0 0 1 0 0

0 0 1 0 0

0 0 1 0 0

0 0 1 0 0

0 0 1 0 0

0 0 1 0 0

0 0 1 0 0

Erosion:



Dilation:



structw =

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

1 1 1 1 1 1 1 1

0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0

Erosion:



Dilation:



Morphological structing element:

Shape: disk

Radius: 10

Operation: open



Radius=11



Radius=12



Radius=13:



Radius=14



Radius=15



The open operation returns the union of all translations of struct element that fit entirely within the areas which is set to 1 in the original image. As we can see, the number of circules gradually reduces with radius growing.

Morphological structing element:

Shape: disk

Radius: 10

Operation: close



Radius=15



Radius=20



Radius=25



Radius=30



Similarly, the remained region expands as the radius grows. Thus, this algorithm can be used in removing tiny dos on the image.

## 2. Extract pixels fusion with boundry of the image

Firstly use the region stuff algorithm, after convertiong the image to binary image, then invert it, conduct a region stuff, the result is particles connected to boundry.



## 3. Extract particles overlap to each other

Put an erosion to original image, for the overlapped particles are bigger in size, the remainder is the overlapped part. Then conduct an image reconstruct on this mask.





## 4. Extract the unoverlapped particles

The individual part can be easily extracted by a subdivision.



# 4． Appendix（Code）:

Mission1

clc

clear all

[y,mapy]=imread('G:\desktop documents\2018´óÈýÏÂ\Êý×ÖÍ¼Ïñ´¦Àí×ÊÁÏFor students\For students\ÊµÑé\±ê×¼Í¼Æ¬\FigP0936(bubbles\_on\_black\_background).tif');

out=BwFilter(y,'filter');

imshow(out)

stru=[0 0 1 0 0 ; 0 0 1 0 0; 0 0 1 0 0; 0 0 1 0 0; 0 0 1 0 0; 0 0 1 0 0; 0 0 1 0 0 ; 0 0 1 0 0 ];

stru2=[0 0 0 0 0 0 0 0;0 0 0 0 0 0 0 0 ; 1 1 1 1 1 1 1 1 ; 0 0 0 0 0 0 0 0 ; 0 0 0 0 0 0 0 0];

out=BwFilter(y,'dilate',stru);

imshow(out);

title('struct1')

set(gca,'position',[0 0 1 1])

stru=strel('disk',30);

out=BwFilter(y,'close',stru);

imshow(out)

set(gca,'position',[0 0 1 1])

Mission 2~4

clc

clear all

[y,mapy]=imread('G:\desktop documents\2018´óÈýÏÂ\Êý×ÖÍ¼Ïñ´¦Àí×ÊÁÏFor students\For students\ÊµÑé\±ê×¼Í¼Æ¬\FigP0936(bubbles\_on\_black\_background).tif');

y=~y;

g=imfill(y,'holes');

g=~g;

imshow(g)

set(gca,'position',[0 0 1 1])

stru=strel('disk',30);

stru=[1 1 1 1 1 1 1;

1 1 1 1 1 1 1;

1 1 1 1 1 1 1;

1 1 1 1 1 1 1;

1 1 1 1 1 1 1;

1 1 1 1 1 1 1;

1 1 1 1 1 1 1;

1 1 1 1 1 1 1];

stru2=[0 0 0 0 0 0 0 0;0 0 0 0 0 0 0 0 ; 1 1 1 1 1 1 1 1 ; 0 0 0 0 0 0 0 0 ; 0 0 0 0 0 0 0 0];

stru3=ones(17);

out=BwFilter(y,'erosion',stru3);

imshow(out)

set(gca,'position',[0 0 1 1])

figure

out2=imreconstruct(out,y);

imshow(out2)

set(gca,'position',[0 0 1 1])

out3= y-out2;

imshow(out3)

set(gca,'position',[0 0 1 1])