
Table of Contents

.....	1
1 - Binarización de regiones	1
2 - Etiquetado de componentes coexas	3
3 - Extracción de regiones por color	5
4 Operaciones morfológicas	6
5 Opcionales	13

% Práctica 5

% Miguel Ascanio Gómez

% Carlos Ballesteros de Andrés

1 - Binarización de regiones

a)

```
clear all; close all;
imagen = imread('Tema05b.bmp','bmp');

figure();

subplot(2,2,1); imshow(imagen); title('Imagen Original')
imagenr = imagen(:,:,1);
subplot(2,2,2); imshow(imagenr); title('Imagen Original (Componente roja)')
subplot(2,2,3); imhist(imagenr); title('Histograma (Componente roja)')

% Con T = 100
umbral = 100 / 255;
imagenBin = im2bw(imagenr, umbral);
subplot(2,2,4); imshow(imagenBin); title('Imagen binarizada, T = 100')

% b)
clear imageBin;
T = graythresh(imagenr);

imagenBin = im2bw(imagenr, T);
figure(); imshow(imagenBin); title('Imagen binarizada, T automatico')
fprintf('Umbral: %f \n', double(T * 255));

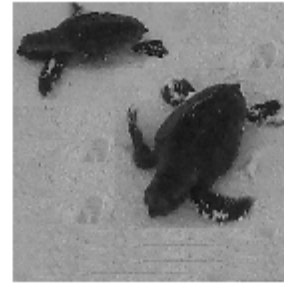
% Se observa como se binariza la imagen: las torugas quedan marcadas en
% negro, mientras que el resto de la imagen queda en blanco. Los dos
% umbrales utilizados son muy parejos, de ahí que las imágenes binarizadas
% sean prácticamente iguales
```

Umbral: 98.000000

Imagen Original



Imagen Original (Componente roja)



Histograma (Componente roja)

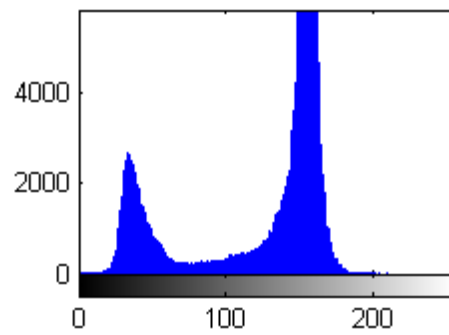
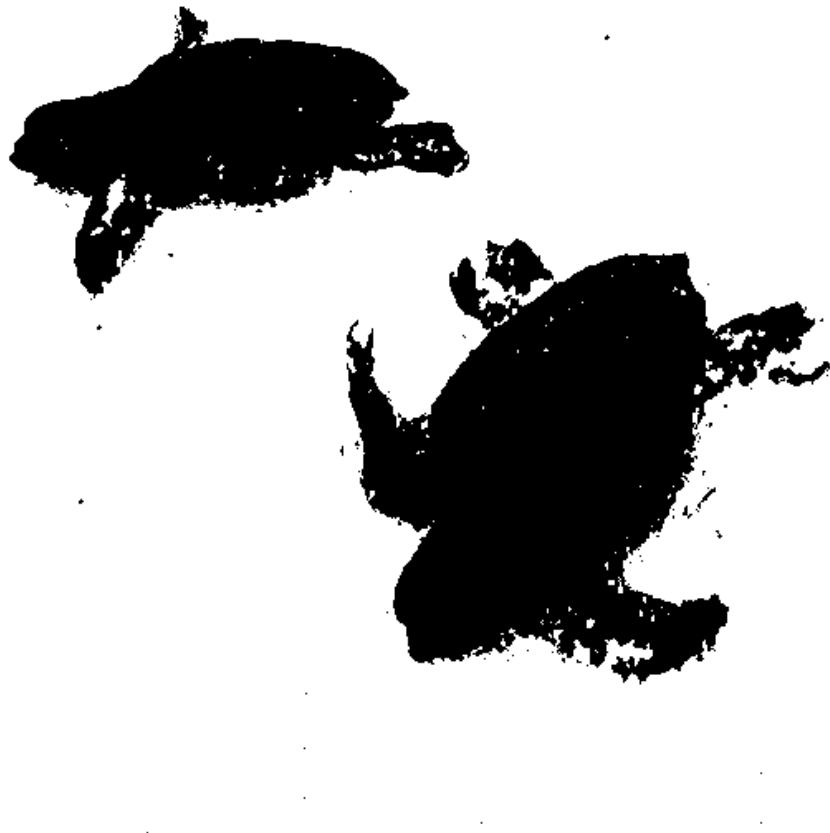


Imagen binarizada, $T = 100$



Imagen binarizada, T automatico



2 - Etiquetado de componentes coenxas

a)

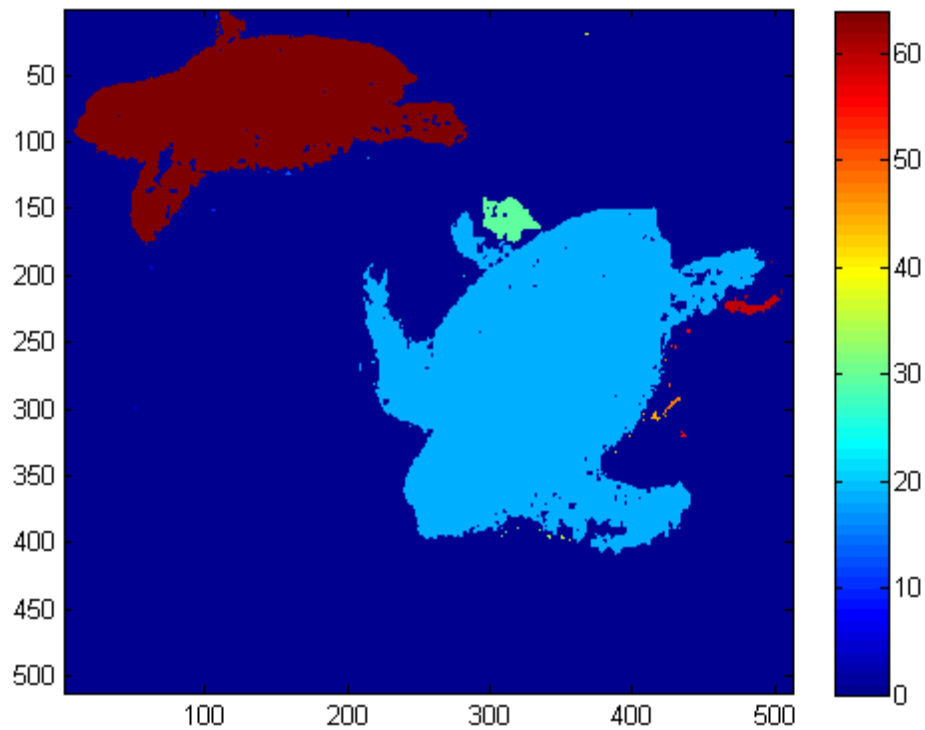
```
[Etiquetas, N] = bwlabel(~imagenBin, 8);  
% Imagen invertida para que la procese correctamente  
  
figure(); imshow(Etiquetas); title('Etiquetas'); impixelinfo();  
% b)  
[x, y] = size (imagenBin);  
Etiquetas2 = Etiquetas;  
  
for i= 1:x  
    for j=1:y  
        if (Etiquetas2(i,j) == 1)  
            Etiquetas2(i,j) = N + 1;  
        end  
    end  
end  
  
figure(); imagesc(Etiquetas2); colorbar;
```

```
% Aquí se observa como cada tortuga, o a "ojos" del ordenador cada
% agrupación de puntos, las componentes conexas, las clasifica con una
% etiqueta, quedando así diferenciadas cada una de las tortugas. Hay que
% tener en cuenta que el "ruido" también lo está agrupando (como las patas
% traseras de la tortuga derecha, que al estar tapadas parcialmente por la
% arena, hay una parte que las clasifica a parte)
```

Etiquetas



Pixel info: (X, Y) Pixel Value



3 - Extracción de regiones por color

```
clear all; close all;
imagen = imread('Tema05b.jpg','jpg');

imagen = imagen(1:4:end, 1:4:end,:);
r = imagen(:,:,1);
g = imagen(:,:,2);
b = imagen(:,:,3);

[M, N] = size(r);

RR = zeros(M,N);
GR = zeros(M,N);
BR = zeros(M,N);

T = 70;

for i=1:M
    for j=1:N
        if (r(i,j) > T && (r(i,j) > g(i,j)) && (r(i,j) > b(i,j)))
            RR(i,j) = 255;
        end
        if (g(i,j) > T && (g(i,j) > r(i,j)) && (g(i,j) > b(i,j)))
            GR(i,j) = 255;
        end
    end
end
```

```

        end
        if (b(i,j) > T && (b(i,j) > r(i,j)) && (b(i,j) > g(i,j)))
            BR(i,j) = 255;
        end
    end
end

subplot(2,2,1); imshow(imagen); title('Imagen original');
subplot(2,2,2); imshow(RR); title('Rojo');
subplot(2,2,3); imshow(GR); title('Verde');
subplot(2,2,4); imshow(BR); title('Azul');

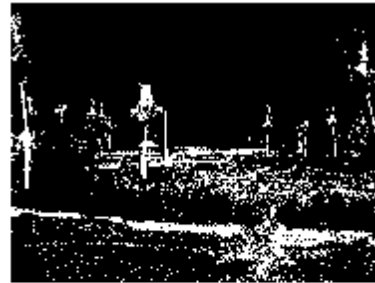
% En este apartado se ven claras diferencias al binarizar usando diferentes
% componentes de color de las imágenes u otras: Binarizando por el verde se
% diferencia entre el césped o las hojas de los árboles, por azul el cielo
% del resto...

```

Imagen original



Rojo



Verde



Azul



4 Operaciones morfológicas

```

clear all; close all;

% Binarización
A = imread('Tema05b.bmp', 'bmp');
B = A(:, :, 1);

T = graythresh(B);

```

```
I = B < 255*T;
clear A B
figure(1); imshow(I); title('imagen original');

% Dilatación
BW = bwmorph(I,'dilate',1);
figure(2); imshow(BW); title('Dilatada');

% Erosión
BW = bwmorph(I,'erode',1);
figure(3); imshow(BW); title('Erosión');

% Apertura
BW = bwmorph(I,'open',1);
figure(4); imshow(BW); title('Apertura');

% Cierre
BW = bwmorph(I,'close',1);
figure(5); imshow(BW); title('Cierre');

% Bordes
B = bwmorph(I,'open',1) - bwmorph(I,'erode',1);
figure(6); imshow(B); title('Bordes');
```

imagen original



Dilatada



Erosión



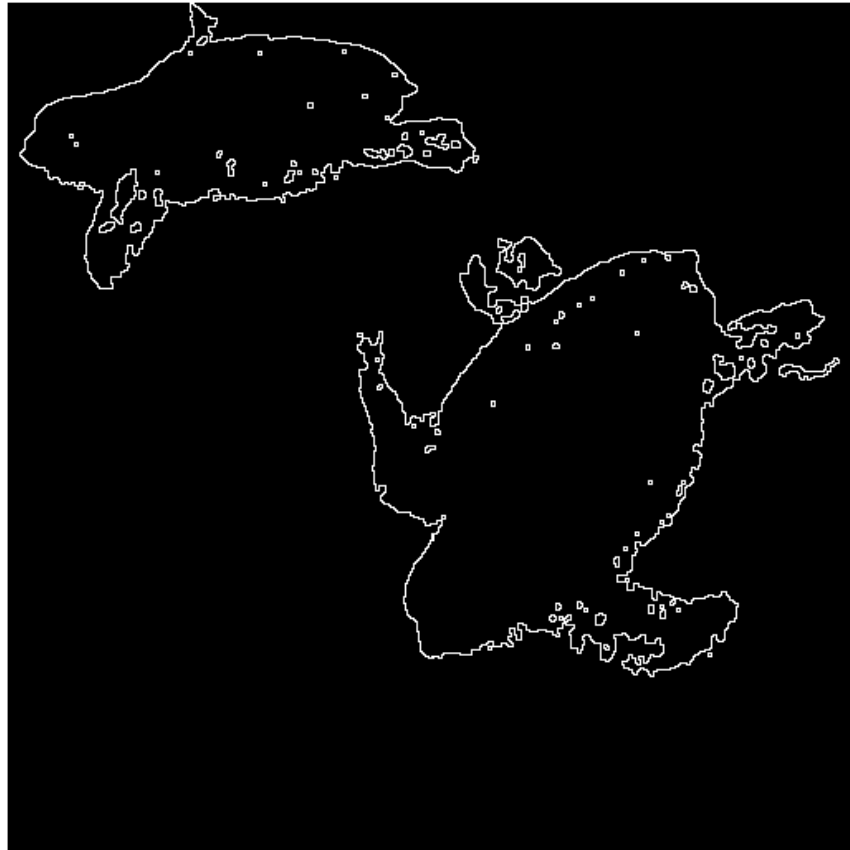
Apertura



Cierre



Bordes



5 Opcionales

```
% Segmentación de regiones método de Ridler-Calvard

clear all; close all
I = imread('Tema05b.bmp','bmp');
A = I(:,:,1);
A = A + 1; % para evitar índices de cero en los arrays

L = 256; % numero de niveles de intensidad

[m,n] = size(A);
e = eps; % desviación

P = zeros(1,L);

for i=1:1:m
    for j=1:1:n
        P(A(i,j)) = P(A(i,j)) + 1;
    end
end
```

```
%pi = P/(m*n);

T1 = mean2(A);

%dividir los datos en dos clases: w1 y w2
h = 1;
for i=1:1:m
    for j=1:1:n
        if A(i,j) <= T1
            w1(h) = A(i,j);
            h = h + 1;
        end
    end
end

h = 1;
for i=1:1:m
    for j=1:1:n
        if A(i,j) > T1
            w2(h) = A(i,j);
            h = h + 1;
        end
    end
end

m1 = mean(w1);
m2 = mean(w2);

T2 = (m1 + m2)/2;

iteracion = 0;

while abs(T1-T2)> e
    T1 = T2;
    h = 1;
    for i=1:1:m
        for j=1:1:n
            if A(i,j) <= T1
                w1(h) = A(i,j);
                h = h + 1;
            end
        end
    end

    h = 1;
    for i=1:1:m
        for j=1:1:n
            if A(i,j) > T1
                w2(h) = A(i,j);
                h = h + 1;
            end
        end
    end
end
```

```

m1 = mean(w1);
m2 = mean(w2);

T2 = (m1 + m2)/2;
iteracion = iteracion + 1;
end
disp('Valor del umbral final:'); disp(T2) %umbral final

subplot(1,2,1); imshow(A-1); title('Imagen Original')

T = (T2-1)/255;
Binaria = im2bw(A-1,T); % los signos - son para corregir la suma por 1 inicial
subplot(1,2,2); imshow(Binaria); title('imagen binarizada T = 101')

% b

clear all; close all;

% Binarización
A = imread('Tema05b.bmp','bmp');
B = A(:,:,1);

T = graythresh(B);
I = B < 255*T;
N = inf;

figure(1); imshow(I); title('imagen original');

% Bothat
BW = bwmorph(I,'bothat',N);
figure(2); imshow(BW); title('Bothat');

% Skel
BW = bwmorph(I,'skel',N);
figure(3); imshow(BW); title('Skel');

% Thin
BW = bwmorph(I,'thin',N);
figure(4); imshow(BW); title('Thin');

% Shrink
BW = bwmorph(I,'shrink',N);
figure(5); imshow(BW); title('Shrink');

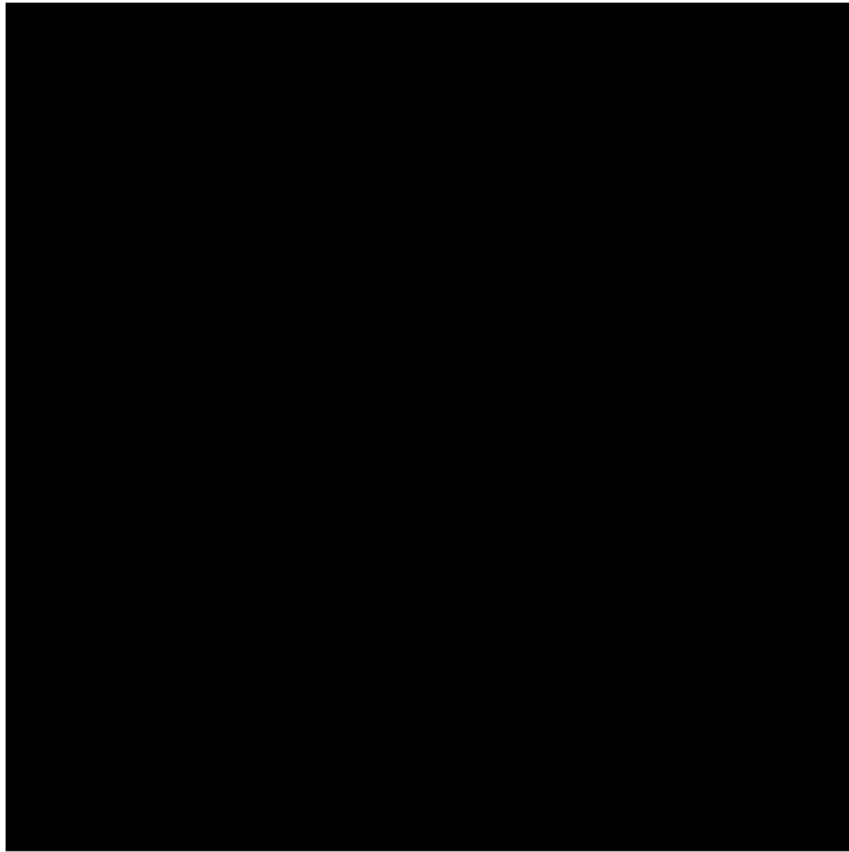
Valor del umbral final:
101.4262

```

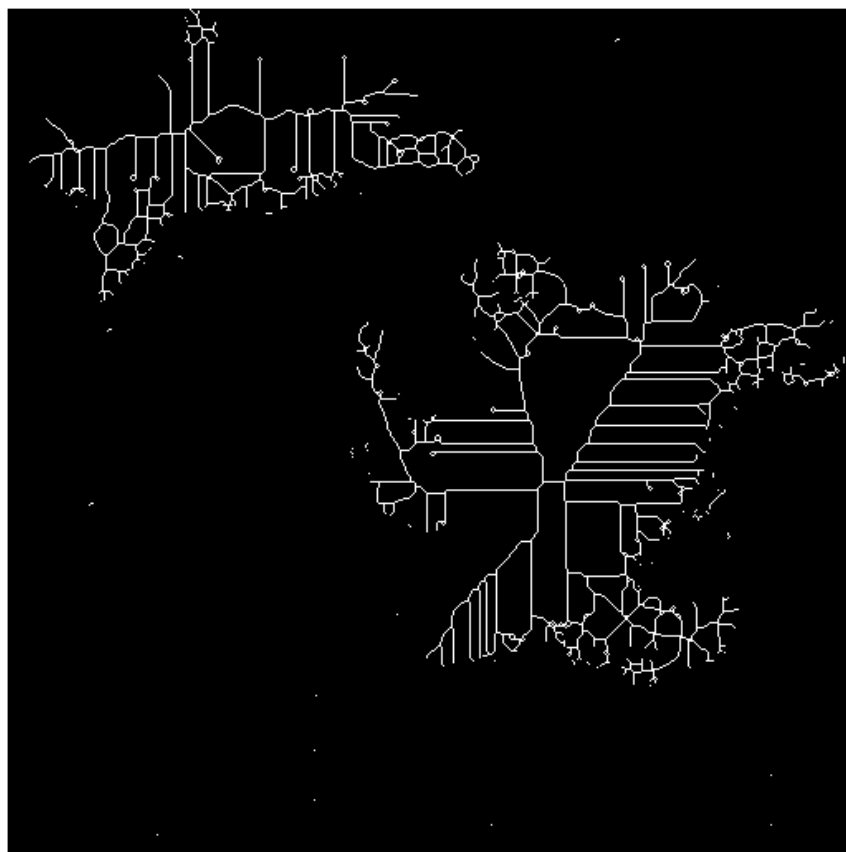
imagen original



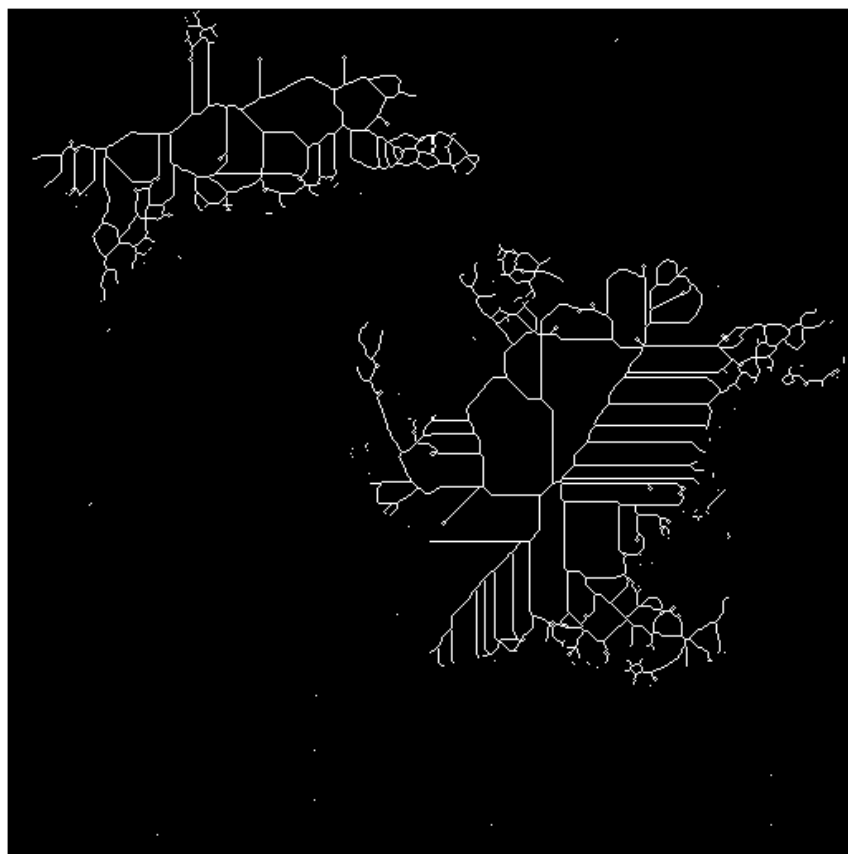
Bothat



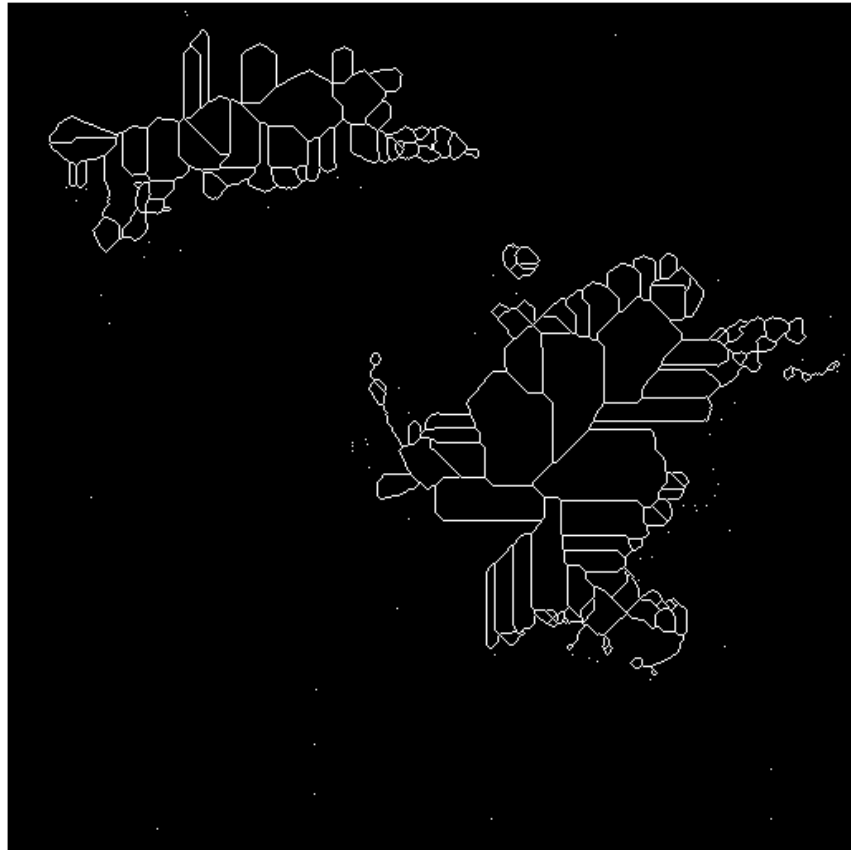
Skel



Thin



Shrink



Published with MATLAB® R2014a