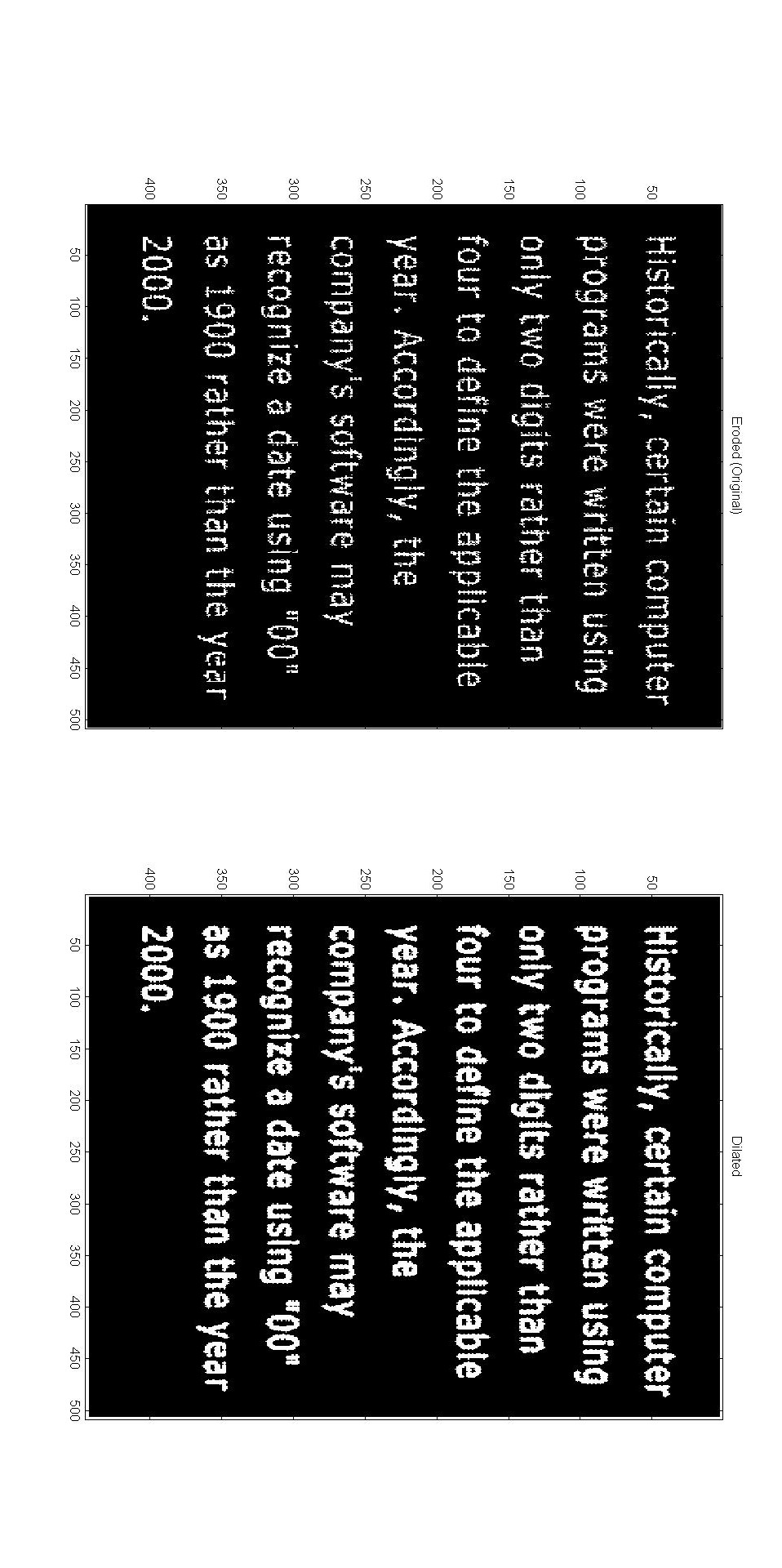
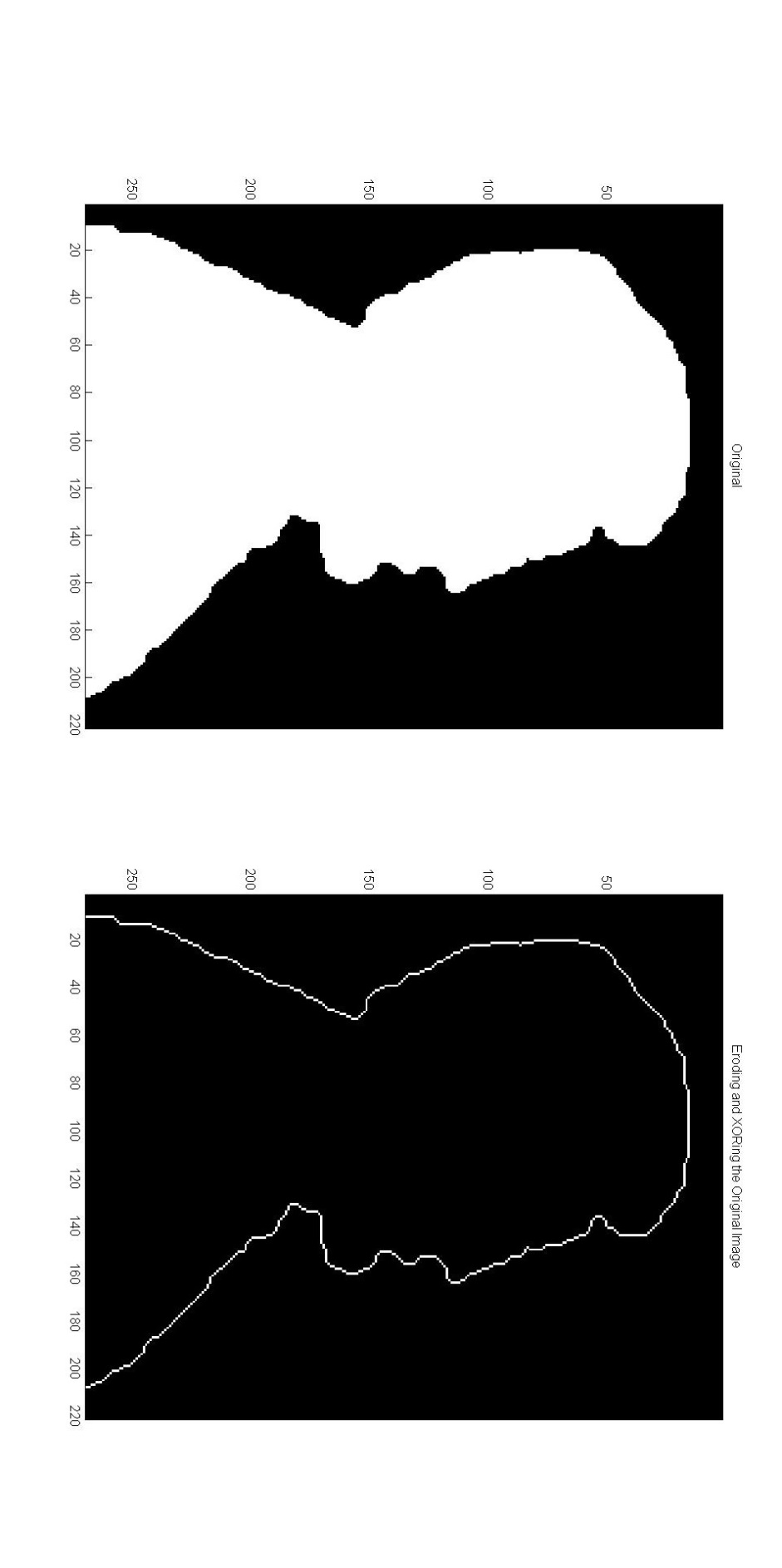
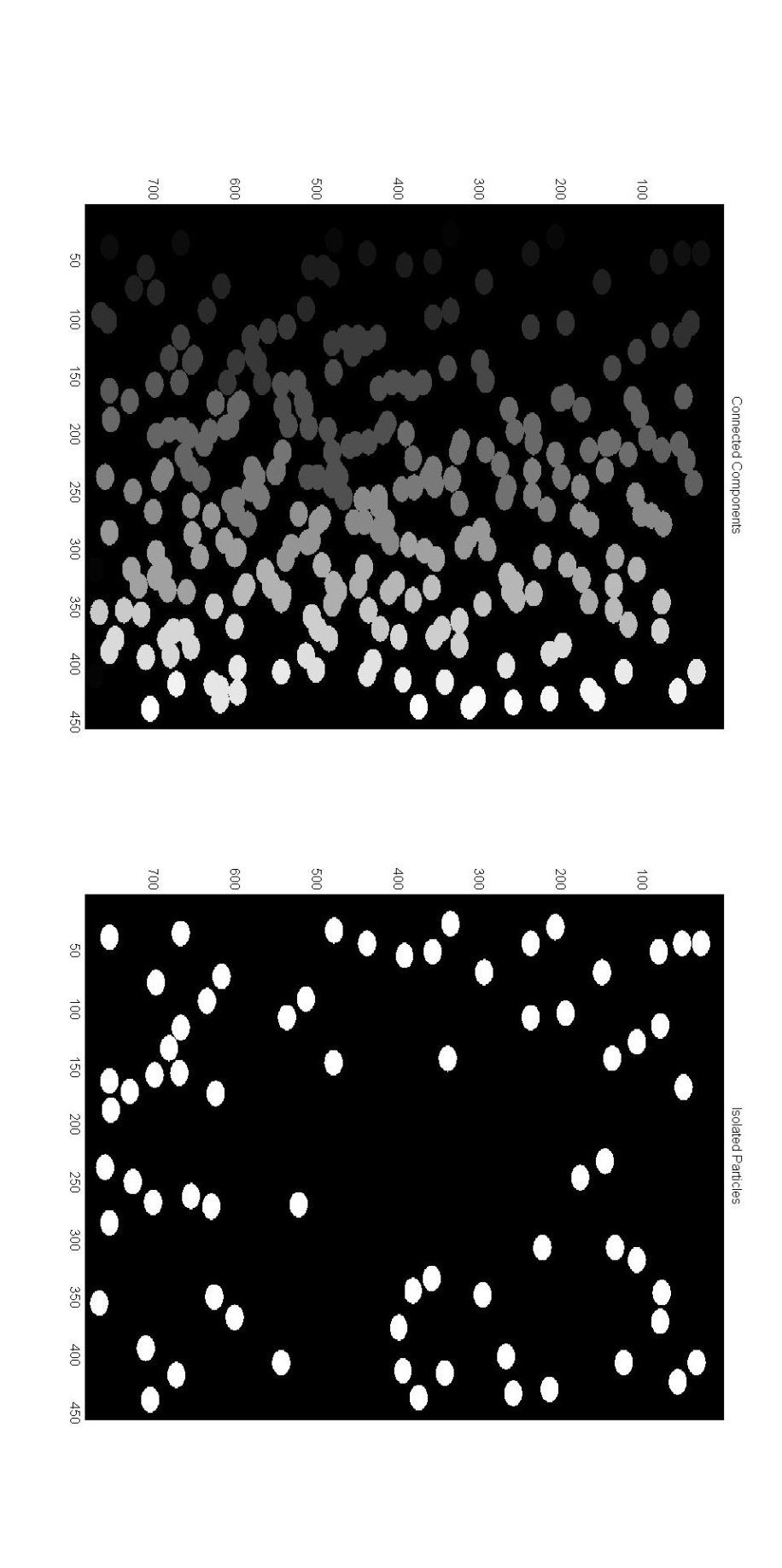
**1a.**



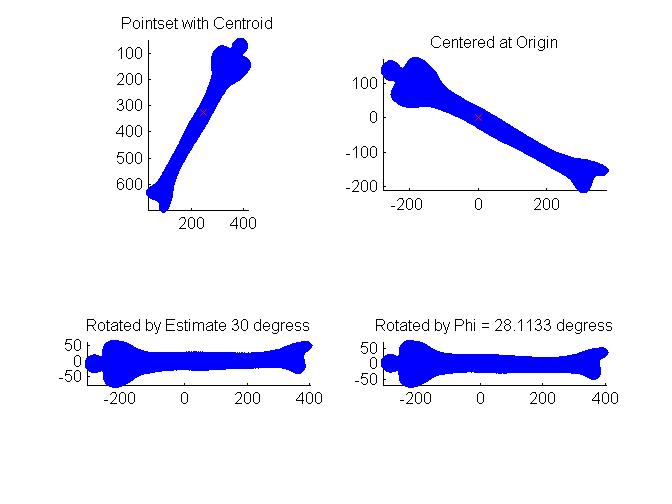
**1b.**



**1c.**



**2.**



**2c.** eigs1 =

1.0e+004 \*

4.5955

0.0559

eigs2 =

1.0e+004 \*

0.0559

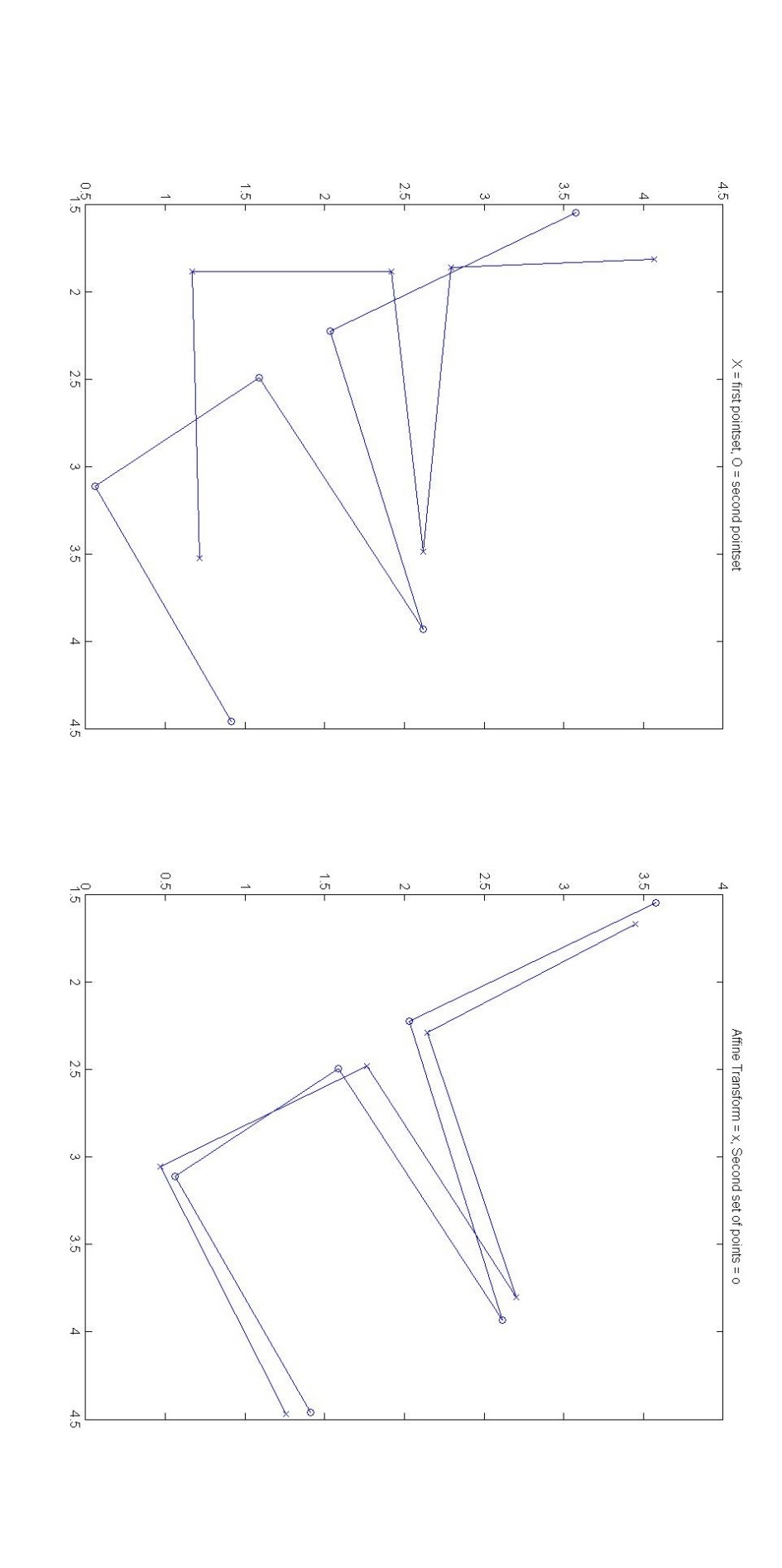
4.5955

**2d.** Aspect ratio = 9.0669

**2e.** Estimated degrees of rotation = 30 degrees.

Phi = 28.1133 degrees.

**3.**



**3b.**

A =

0.8788 -0.4558

0.4532 1.0391

t =

1.9281

-1.6005

**3c.**

Before Alignment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Corresponding Point | 1 | 2 | 3 | 4 | 5 | 6 |
| Distance | 0.5601 | 0.8422 | 0.4444 | 1.0293 | 1.3704 | 0.9566 |

After Alignment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Corresponding Point | 1 | 2 | 3 | 4 | 5 | 6 |
| Distance | 0.1772 | 0.1314 | 0.1555 | 0.1794 | 0.1100 | 0.1557 |

function hw5p1a

im = imread('Fig0907(a)(text\_gaps\_1\_and\_2\_pixels).tif');

%erode = imerode(im, [0 1 0; 1 1 1; 0 1 0]);

subplot(1,2,1);

imagesc(im);

colormap('gray');

title('Eroded (Original)');

dilate = imdilate(im, [0 1 0; 1 1 1; 0 1 0]);

subplot(1,2,2);

imagesc(dilate);

colormap('gray');

title('Dilated');

end

function hw5p1b

im = imread('Fig0914(a)(licoln from penny).tif');

erode = imerode(im, [0 1 0; 1 1 1; 0 1 0]);

edges = xor(im, erode);

subplot(1,2,1);

imagesc(im);

colormap('gray');

title('Original');

subplot(1,2,2);

imagesc(edges);

colormap('gray');

title('Eroding and XORing the Original Image');

end

function hw5p1c

im = imread('FigP0936(bubbles\_on\_black\_background).tif');

[L, NUM] = bwlabel(im, 4);

subplot(1,2,1);

imagesc(L);

colormap('gray');

title('Connected Components');

stats = regionprops(L, 'Area');

stats = cell2mat(struct2cell(stats));

for i=1:size(stats,2)

temp = stats(1,i);

if(temp >= 400 || temp <= 380)

stats(1,i) = 0;

else

stats(1,i) = 1;

end

end

res = zeros(size(L));

for i=1:size(stats,2)

if(stats(1,i) == 1)

[r, c] = find(L == i);

rc = [r c];

for j=1:size(rc,1)

res(rc(j,1), rc(j,2)) = 1;

end

end

end

subplot(1,2,2);

imagesc(res);

colormap('gray');

title('Isolated Particles');

end

function hw5p2a

I = double(imread('Fig1116(leg\_bone).tif'));

[r c] = find(I);

figure;

subplot(2,2,1);

scatter(c, r, 'b.');

axis('image');

axis('ij');

title('Pointset with Centroid');

hold;

[cX, cY] = ait\_centroid(I);

scatter(cX, cY, 'rx');

%move image's centroid to the origin

centered = find(I);

centered(:,1) = c - cX;

centered(:,2) = r - cY;

cmatrix = [0 0; 0 0];

for i=1:size(centered,1)

cmatrix = cmatrix + (transpose(centered(i,:)) \* centered(i,:));

end

cmatrix = cmatrix ./ size(centered,1)

trace = cmatrix(1,1) + cmatrix(2,2);

lambda2 = (trace + sqrt(trace^2 - 4\*det(cmatrix)))/2;

lambda3 = (trace - sqrt(trace^2 - 4\*det(cmatrix)))/2;

eigs1 = [lambda2; lambda3]

eigs2 = eig(cmatrix)

aspectratio = sqrt(4.5955/0.0559)

subplot(2,2,2);

h = scatter(centered(:,2), centered(:,1), 'b.');

axis('image');

title('Centered at Origin');

hold;

scatter(0,0,'rx');

estimate = 30

subplot(2,2,3);

h = scatter(centered(:,2), centered(:,1), 'b.');

h=get(h,'children');

rotate(h, [0 0 1], 30, [0 0 0]);

axis('image');

title('Rotated by Estimate 30 degress');

degree = 2\*cmatrix(1,2)/(cmatrix(1,1) - cmatrix(2,2));

phi = (atan(degree))/2;

phi = phi \* 180 / pi;

subplot(2,2,4);

h = scatter(centered(:,2), centered(:,1), 'b.');

h=get(h,'children');

rotate(h, [0 0 1], phi, [0 0 0]);

axis('image');

title(['Rotated by Phi = ', num2str(phi), ' degress']);

end

function [meanx,meany] = ait\_centroid(pic)

[x,y,z] = size(pic); % Checking whether the picture is colored or monochromatic, if colored then converting to gray.

if(z==1)

;

else

pic = rgb2gray(pic);

end

im = pic;

[rows,cols] = size(im);

x = ones(rows,1)\*[1:cols]; % Matrix with each pixel set to its x coordinate

y = [1:rows]'\*ones(1,cols); % " " " " " " " y "

area = sum(sum(im));

meanx = sum(sum(double(im).\*x))/area;

meany = sum(sum(double(im).\*y))/area;

end

function hw5p3a

coords = load('coords.txt');

ps1x = coords(1, :);

ps1y = coords(2, :);

ps2x = coords(3, :);

ps2y = coords(4, :);

original = [ps1x; ps1y];

transform = [ps2x; ps2y];

figure;

subplot(1,2,1);

plot(ps1x, ps1y, 'x-');

title('X = first pointset, O = second pointset');

hold;

plot(ps2x, ps2y, 'o-');

ctX = mean(ps1x);

ctY = mean(ps1y);

coX = mean(ps2x);

coY = mean(ps2y);

centeredO = zeros(size(original));

centeredO(1,:) = original(1,:) - coY;

centeredO(2,:) = original(2,:) - coX;

centeredT = zeros(size(transform));

centeredT(1,:) = transform(1,:) - ctY;

centeredT(2,:) = transform(2,:) - ctX;

A = centeredT \* pinv(centeredO)

t = transform - A \* original;

tX = mean(t(1,:));

tY = mean(t(2,:));

t = [tX; tY]

translated = A \* original;

for i=1:size(translated,2)

translated(:,i) = translated(:,i) + [tX;tY];

end

subplot(1,2,2);

plot(translated(1,:),translated(2,:), 'x-');

title('Affine Transform = x, Second set of points = o');

hold;

plot(ps2x, ps2y, 'o-');

D1 = [0 0 0 0 0 0];

for i=1:size(D1,2)

D1(1,i) = sqrt((ps2x(1,i)-ps1x(1,i))^2 + (ps2y(1,i)-ps1y(1,i))^2);

end

D1

D2 = [0 0 0 0 0 0];

for i=1:size(D2,2)

D2(1,i) = sqrt((ps2x(1,i)-translated(1,i))^2 + (ps2y(1,i)-translated(2,i))^2);

end

D2

end