Appendices

Appendix I

MATLAB commands used:

```
K = imread('kj10.jpg');
figure, imshow(K), title('original image');
I2 = imresize(K,[600 600]);
I = rgb2gray(I2);
J = imadjust(I);
L = medfilt2(J, [3 3]);
figure, imshow(L)
figure, imshow(J)
thresh =  .4;
BWs = edge(L,'canny',thresh);
figure, imshow(BWs), title('Edge Detector');
%B = imresize(BWs, [400 600])
se = strel('disk',10);
IM2 = imclose(BWs,se);
figure, imshow(IM2), title('Close Image');
%BW2 = bwareaopen(IM2,50);
se90 = strel('line', 3, 90);
se0 = strel('line', 3, 0);
BWsdil = imdilate(IM2, [se90 se0]);
figure, imshow(BWsdil), title('dilated gradient mask');
BWdfill = imfill(BWsdil, 'holes');
figure, imshow(BWdfill), title('fill image');
B = imresize(BWdfill, [300 300]);
BW3 = bwareaopen(B,600);
figure, imshow(BW3), title('remove small area image');
%Img1 = imresize(BWfinal,[300 300],'bicubic');
 B2 = imresize(K, [300 300]);
BWoutline = bwperim(BW3,8);
figure, imshow(BWoutline), title('outline image');
Segout = B2;
Segout(BWoutline) = 255;
figure, imshow(Segout), title('original outline image');
%seD = strel('disk',6);
%BWfinal = imerode(BWdfill,seD);
%figure, imshow(BWfinal), title('segmented image');
%BWfinal = imerode(BWfinal,seD);
%figure, imshow(BWfinal), title('segmented image   two');
%----------------------------------  feature extracted
%Entrop = entropy(Segout(BWoutline))
IM3 = imresize(BW3,[300 300]);
IM4 = bwlabel(IM3);
%[B,L] = bwboundaries(IM4,'noholes');
%imshow(label2rgb(L, @jet, [.5 .5 .5]))
%hold on
%for k = 1:length(B)
    %boundary = B{k};
   % sc{k} = imArea(boundary);
   % plot(boundary(:,2), boundary(:,1), 'w', 'LineWidth', 2)
%end
%IM4 = max(sc(1),sc(2),sc(3));
s9  = regionprops(IM4, 'Orientation')
 IM5 = imrotate(IM4,180-s9.Orientation(1));
figure, imshow(IM5), title('Allign image to horizontal axis');
 IM6 = bwlabel(IM5);
 s0  = regionprops(IM6, 'Area');
 s01  = regionprops(IM6, 'perimeter');
% perim = imPerimeter(IM6);
```

```
 s1  = regionprops(IM6, 'centroid');
 s2  = regionprops(IM6, 'Extent');
 s3  = regionprops(IM6, 'EquivDiameter');
 s4  = regionprops(IM6, 'BoundingBox');
 s5  = regionprops(IM6, 'Eccentricity')
 s6  = regionprops(IM6, 'Extrema');
 s7  = regionprops(IM6, 'MajorAxisLength');
 s8  = regionprops(IM6, 'MinorAxisLength');
 s10 = regionprops(IM6, 'Orientation');
 s11 = regionprops(IM6,  'ConvexArea' );
% relation features
permi = cat(1, s01.Perimeter)
ari = cat(1, s0.Area)
object_length = cat(1, s7.MajorAxisLength)
Length_obj = object_length(1)
object_width = cat(1, s8.MinorAxisLength)
width_obj = object_width(1)
object_extent = cat(1, s2.Extent)
extnt = object_extent(1)
Ecc = s5.Eccentricity(1)
object_diameter = cat(1, s3.EquivDiameter)
eq_diam = object_diameter(1)
per1 = permi(1)
are1 = ari(1)
Rel1 = per1/are1
Rel2 = per1/ Length_obj
Rel3 =  per1/width_obj
Rel4 = width_obj/Length_obj
Rel5 = ari/Length_obj;
Rel6 = ari/width_obj;
% shape index feature extractor
%Entrop = entropy(Segout)
% shape circulity (shape index)
% Normalization Process with initialize the feature vectores
%Feature_vector(16)= 0 ;
%f1 = roundTo(f1,5);
%f2 = roundTo(f2,5);
f1 = Length_obj / 255;
%f3 = roundTo(f3,5);
f2 = width_obj / 255;
%f4 = roundTo(f4,5);
f3 = extnt;
%f5 = roundTo(f5,5);
%f6 = roundTo(f6,5);
f4 = eq_diam/255;
%f7 = roundTo(f7,5);
f5 = per1/1000;
%f8 = roundTo(f8,5);
f6 = are1/1000;
%f9 = roundTo(f9,5);
f7 = Rel1;
%f10 = roundTo(f10,5);
f8 = Rel2;
%f11 = roundTo(f11,5);
f9 = Rel3;
%f12 = roundTo(f12,5);
f10 = Rel4;
%f13 = roundTo(f13,5);
f11 = Rel5; %f14 = roundTo(f14,5);
f12 = Rel5; %f18 = roundTo(f18,5); Feature_vector = [f1 f2 f3 f4 f5 f6
f7 f8 f9 f10 f11 f12]
```

Appendix II

Screen Snapshot for Developed Systems