

```

%===== segmentK =====
%
% script segment1
%
%
% This is a Matlab script that will run the simple k-means algorithm
% for segmenting an image. You should select two images from the
% homework Matlab file and perform segmentation on them. The output will
% be used by another script of mine for visualization.
%
%
% This code uses some Matlab tricks to be somewhat generic. First,
% all arguments are encapsulated into a cell array. This works as
% follows. The cell array belows consists of two arguments:
%
% >> sampleCellArray = {40, 34};
%
% that when expanded as an argument to a function, provides two
% inputs to the function,
%
% >> plus( sampleCellArray{:} )
%
% The output should be the addition of the two arguments:
%
% ans =
%
%      74
%
% Anyhow, this function expectes the same, but the arguments are
% consistent with what the segKmeans function expects.
%
%===== segmentK =====
close all
clear
load('segment.mat');
picks = [1 2];

for i = 1:length(picks)
    switch (picks(i))
        case 1,
            images{i} = westin;
            iparms{i} = { 5 , [3, 10, 60, 140, 150] };
        case 2,
            images{i} = fish04;
            iparms{i} = { 5 , [1, 5 ,40, 80, 120] };
    end
end

% Add more cases if you want to do all of the images, and change
% "picks" accordingly.

for i = 1:length(images)
    [segimg{i}, K, nmeans{i}] = segKmeans(double(images{i}), iparms{i}{:});
    figure(i);

    figure(3 * i);
    imagesc(images{i});
    axis image;
    colormap('gray');
end

```

```
figure(3 * i + 1);
imagesc(uint8(segimg{i}));
axis image;
colormap('default');

figure(3 * i + 2);
imagesc(uint8(K));
axis image;
colormap('gray');

nmeans{i};
end

%
%===== segmentK =====
```

---







