# Problem 1

#Main

maindir = 'E:\SIT\CS558\cs558s18\_hw4\cs558s18\_hw4\ImClass';

subdir = dir( maindir );

for i = 1 : length( subdir )

subdirpath = fullfile( maindir, subdir( i ).name, '\*.jpg' );

dat = dir( subdirpath ) ;

for j = 1 : length( dat )

datpath = fullfile( maindir, subdir( i ).name, dat( j ).name);

image{j} = double(imread(datpath));

end

end

%%

bin = 8;

image\_rep = rep\_image(image,bin);

train = image\_rep([5:8,13:16,21:24],:);

y = [1;1;1;1;2;2;2;2;3;3;3;3]; %1 is coast, 2 is forest, 3 is incidecity

test = image\_rep([1:4,9:12,17:20],:);

test\_y = [1;1;1;1;2;2;2;2;3;3;3;3];

%%

wrong = 0;

test\_size = size(test,1);

for i = 1: test\_size

k = dsearchn(train, test(i,:));

fprintf('\nTest image %d of class %d has been assigned to class %d.\n', i, test\_y(i),y(k));

if test\_y(i)~=y(k)

wrong = wrong + 1;

end

end

accuracy = (test\_size-wrong)/test\_size;

fprintf('\nAccuracy is %.2f', accuracy);

pause;

%%

bin = 1:64;

accuracy = zeros(length(bin),2);

accuracy(:,1) = bin;

for i = 1:length(bin)

image\_rep = rep\_image(image,bin(i));

train = image\_rep([5:8,13:16,21:24],:);

test = image\_rep([1:4,9:12,17:20],:);

accuracy(i,2) = cal\_accuracy(train,test);

end

plot(accuracy(:,1),accuracy(:,2));

ylim([0.5,1]);

pause;

#hist

function RGB\_vector = hist(X, bin)

RGB\_vector = zeros(3\*bin,1);

for i = 1:3

RGB\_vector((i-1)\*bin+1:i\*bin,:) = histcounts(X(:,:,i),bin);

end

end

#rep\_image

function image\_rep = rep\_image(X,bin)

image\_rep = [];

for i = 1:length(X)

p = cell2mat(X(i));

RGB\_V = hist(p,bin);

image\_rep(i,:) = RGB\_V';

end

end

#cal\_accuracy

function accuracy= cal\_accuracy(train,test)

wrong = 0;

test\_size = size(test,1);

y = [1;1;1;1;2;2;2;2;3;3;3;3];

test\_y = [1;1;1;1;2;2;2;2;3;3;3;3];

for j = 1: test\_size

k = dsearchn(train, test(j,:));

if test\_y(j)~=y(k)

wrong = wrong + 1;

end

end

accuracy = (test\_size-wrong)/test\_size;

end

#Result

Test image 1 of class 1 has been assigned to class 1.

Test image 2 of class 1 has been assigned to class 1.

Test image 3 of class 1 has been assigned to class 1.

Test image 4 of class 1 has been assigned to class 1.

Test image 5 of class 2 has been assigned to class 2.

Test image 6 of class 2 has been assigned to class 2.

Test image 7 of class 2 has been assigned to class 2.

Test image 8 of class 2 has been assigned to class 2.

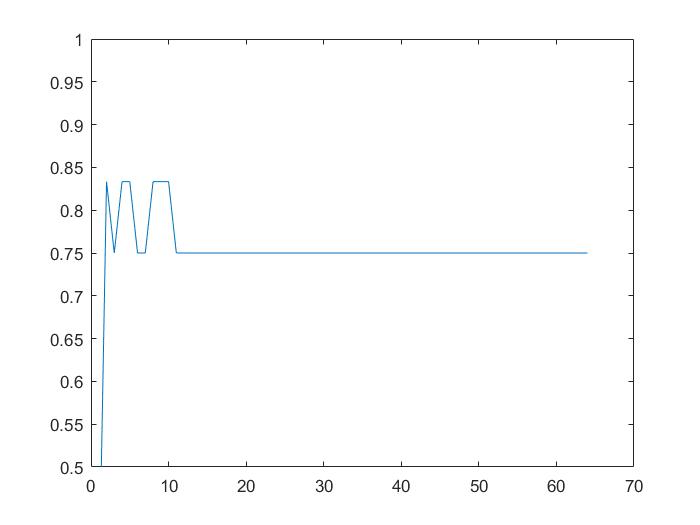
Test image 9 of class 3 has been assigned to class 2.

Test image 10 of class 3 has been assigned to class 1.

Test image 11 of class 3 has been assigned to class 3.

Test image 12 of class 3 has been assigned to class 3.

Accuracy is 0.83



The bin size doesn’t influence accuracy significantly.

# 

# Problem 2

#Main

%% Load training image

sky\_train = double(imread('sky.jpg'));

ground\_train = double(imread('ground.jpg'));

sky\_test1 = double(imread('sky\_test1.jpg'));

sky\_test2 = double(imread('sky\_test2.jpg'));

sky\_test3 = double(imread('sky\_test3.jpg'));

sky\_test4 = double(imread('sky\_test4.jpg'));

%% k-mean

k = 10;

sky\_train\_kmean = reshape(sky\_train, [256\*256,3]);

ground\_train\_kmean = reshape(ground\_train, [256\*256,3]);

[id\_sky,sky\_cent] = kmeans(sky\_train\_kmean, k, 'EmptyAction', 'singleton');

[id\_ground, ground\_cent] = kmeans(ground\_train\_kmean, k, 'EmptyAction', 'singleton');

%% test

[sky1, ground1] = split\_sky\_ground(sky\_test1,sky\_cent,ground\_cent);

[sky2, ground2] = split\_sky\_ground(sky\_test2,sky\_cent,ground\_cent);

[sky3, ground3] = split\_sky\_ground(sky\_test3,sky\_cent,ground\_cent);

[sky4, ground4] = split\_sky\_ground(sky\_test4,sky\_cent,ground\_cent);

%% Show the picture

subplot(1,4,1);

imshow(uint8(ground1));

subplot(1,4,2);

imshow(uint8(ground2));

subplot(1,4,3);

imshow(uint8(ground3));

subplot(1,4,4);

imshow(uint8(ground4));

# split\_sky\_ground

function [sky,ground] = split\_sky\_ground(X,sky\_cent, ground\_cent)

[m,n,k] = size(X);

sky = zeros(size(X));

ground = zeros(size(X));

for i = 1: m

for j = 1:n

p = reshape(X(i,j,:),[1,3]);

d\_sky = mean(sqrt(sum((p - sky\_cent).^2,2)));

d\_ground = mean(sqrt(sum((p - ground\_cent).^2,2)));

if d\_sky < d\_ground

sky(i,j,:) = X(i,j,:);

else

ground(i,j,:) = X(i,j,:);

end

end

end

#Result

