**3(b)**

uclabruin = imread('UCLA\_Bruin.jpg');

imshow(uclabruin);

%reformat image

img = zeros ( 120000, 5 );

for i = 1 : 300

for j = 1 : 400

img(j+400\*i-400, 1) = i;

img(j+400\*i-400, 2) = j;

img(j+400\*i-400, 3) = uclabruin(i, j, 1);

img(j+400\*i-400, 4) = uclabruin(i, j, 2);

img(j+400\*i-400, 5) = uclabruin(i, j, 3);

end

end

max\_clst = 4;

u = img(1, 3:5);

for clus = 2 : max\_clst

curr\_u = u(1, :);

for i = 1 : 120000

s = size(u);

nClst = s(1);

d1 = zeros(1, nClst);

d2 = zeros(1, nClst);

for j = 1 : s(1)

d1(j) = norm(img(i, 3:5) - u(j, :));

d2(j) = norm(curr\_u - u(j, :));

end

if (min(d1) > min(d2))

curr\_u = img(i, 3:5);

end

end

u = [u ; curr\_u];

end

J = zeros(1, 10);

for iter = 1:10

r = zeros(120000,max\_clst); % intialize r\_nk

total = zeros(max\_clst, 3);

nPix = zeros(max\_clst, 1);

clst = 0;

J\_iter = 0;

for i = 1:120000

d = intmax;

for k = 1:max\_clst % find the closest u

curr\_d = norm(img(i, 3:5) - u(k, :));

if (curr\_d < d)

d = curr\_d;

clst = k;

end

end

r(i,clst) = 1; % set the data point to the found cluster

total(clst, :) = total(clst, :) + img(i, 3:5); % re-estimate u

nPix(clst) = nPix(clst)+1;

end

for i = 1:120000

for k = 1:max\_clst

u(k, :) = total(k,:) / nPix(k);

J\_iter = J\_iter + r(i, k)\*norm(img(i, 3:5) - u(k,:)).^2;

end

end

J(iter) = J\_iter;

end

figure;

scatter([1:10],J);

new = uclabruin;

for i = 1 : 300

for j = 1 : 400

current = zeros(1, 3);

current = transpose(u) \* transpose(r(j+400\*i-400, :));

new(i, j, 1) = current(1);

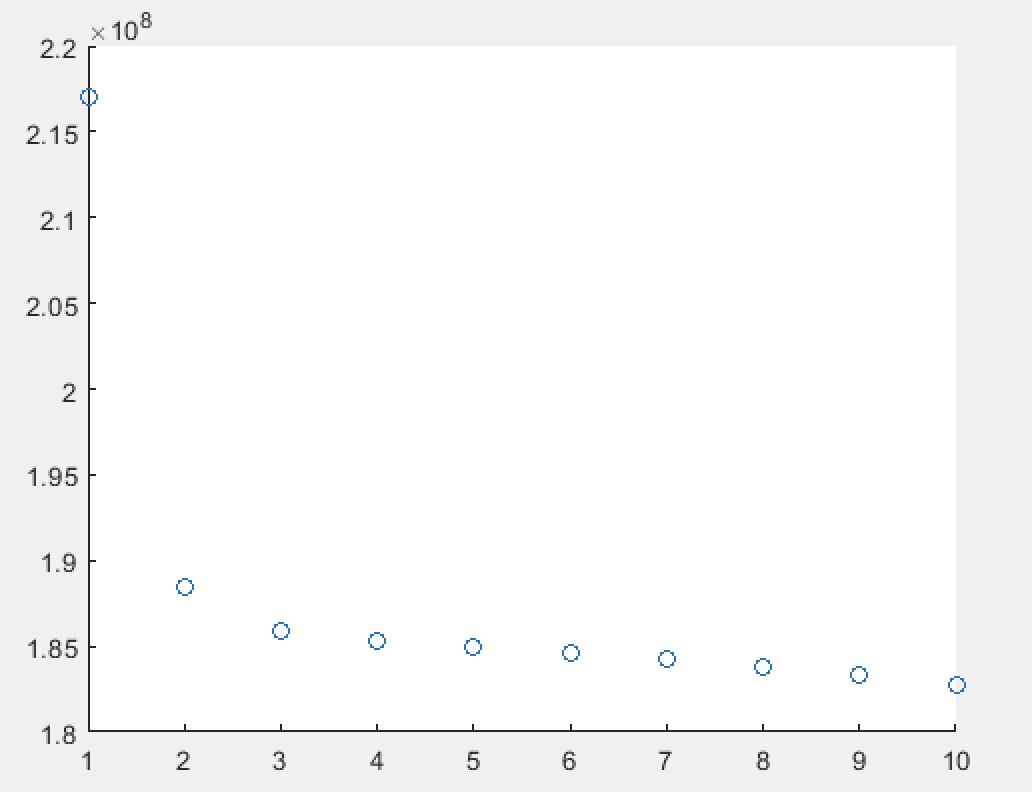
new(i, j, 2) = current(2);

new(i, j, 3) = current(3);

end

end

imshow(new);



The K-means algorithm seems to converge to 1.8E+8 after 10 iterations.

**3(c)**

K = 4



J\_10 = 1.8272E+8

K = 8



J\_10 = 5.5495E+7

K = 16



J\_10 = 3.1021E+7

Increasing number of clusters results in more distinguished and colorful picture.

**3(d).**

120000\*24=2880000bits

N\*log(K) + 24\*K

K = 4 120000\*2+24\*4 = 240096 Compression ratio = 8.34%

K = 8 120000\*3+24\*8 = 360192 Compression ratio = 12.51%

K = 16 120000\*4+24\*16 = 480384 Compression ratio = 16.68%