# A report on image clustering

(Henry Tu, 04-oct-2015)

## Q1. Find the feature selection function to convert an image into a vector

In this project we choose to use spatial color-averaging as the feature selection function to transform one image into vector.

We use 3 layers. For layer 1, we have one vector of 3 dimensions which is the average colors in the channels of the image. For layer 2, we have 4 vectors of 3 dimensions from 4 sub images of the original image. For layer 3, we have 16 vectors of 3 dimensions from 16 sub images.

In code, we write

|  |
| --- |
| colorAvg <- function(img)  {  r <- mean(img[,,1]);  g <- mean(img[,,2]);  b <- mean(img[,,3]);  return( c(r, g, b) );  }  colorAvg2 <- function(img)  {  mr = nrow(img);  mc = ncol(img);  mr <- mr / 2;  mc <- mc / 2;  a1 <- img[1:mr, 1:mc, ];  a2 <- img[1:mr + mr, 1:mc, ];  a3 <- img[1:mr, 1:mc + mc, ];  a4 <- img[1:mr + mr, 1:mc + mc, ];    v1 <- colorAvg(a1);  v2 <- colorAvg(a2);  v3 <- colorAvg(a3);  v4 <- colorAvg(a4);    m <- rbind(v1, v2, v3, v4);  return(m);  }  colorAvg3 <- function(img)  {  mr = nrow(img);  mc = ncol(img);  mr <- mr / 2;  mc <- mc / 2;  a1 <- img[1:mr, 1:mc, ];  a2 <- img[1:mr + mr, 1:mc, ];  a3 <- img[1:mr, 1:mc + mc, ];  a4 <- img[1:mr + mr, 1:mc + mc, ];    v1 <- colorAvg2(a1);  v2 <- colorAvg2(a2);  v3 <- colorAvg2(a3);  v4 <- colorAvg2(a4);    m <- rbind(v1, v2, v3, v4);  return(m);  }  colorAvg123 <- function(img)  {  img <- readJPEG(img);  m1 <- colorAvg(img);  m2 <- colorAvg2(img);  m3 <- colorAvg3(img);  m <- rbind(m1, m2, m3);  return( as.vector(m) );  } |

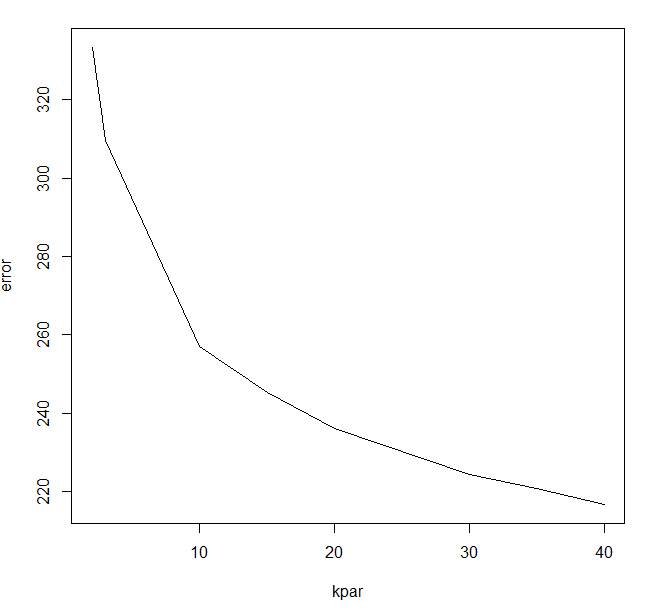
To load the folder of images, we apply the spatial color averaging function to each image.

In code, we write

|  |
| --- |
| load1 <- function(path, num=2688)  {  files <- list.files(path, full.name=TRUE);  num <- min(num, length(files));  df <- sapply(files[1:num], colorAvg123);  return( t(df) );  } |

## Q2. Divide the images into k clusters using kmeans

We use kmeans procedure to explore the dataset with different number of k and we arrive at the following table. We saw that when the number of clusters goes up, the error goes down.



We choose to use k=15 and k=20 to produce the output

## Q3. Rendering the result in HTML page

Please see the output15.html and output20.html to have more details

