Image Compression using SVM

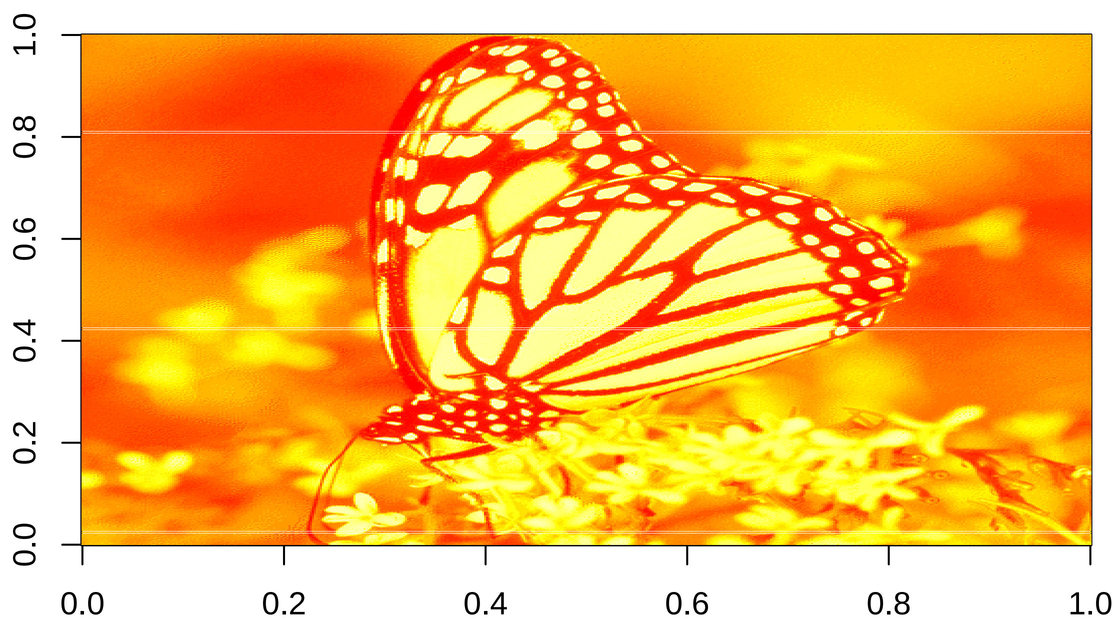
Let's take an RGB image of a butterfly to perform Singular Value Decomposition

```
#install.packages("pixmap")
library(pixmap)
image <- read.pnm("Monarch.512.ppm")
```

```
## Warning in rep(cellres, length = 2): 'x' is NULL so the result will be NULL
```

```
#Splittig image by color and assigning to a variable
red <- matrix(image@red, nrow = image@size[1], ncol = image@size[2])
green <- matrix(image@green, nrow = image@size[1], ncol = image@size[2])
blue <- matrix(image@blue, nrow = image@size[1], ncol = image@size[2])
```

```
image(red, col = heat.colors(255)) #Image Before decomposition
```

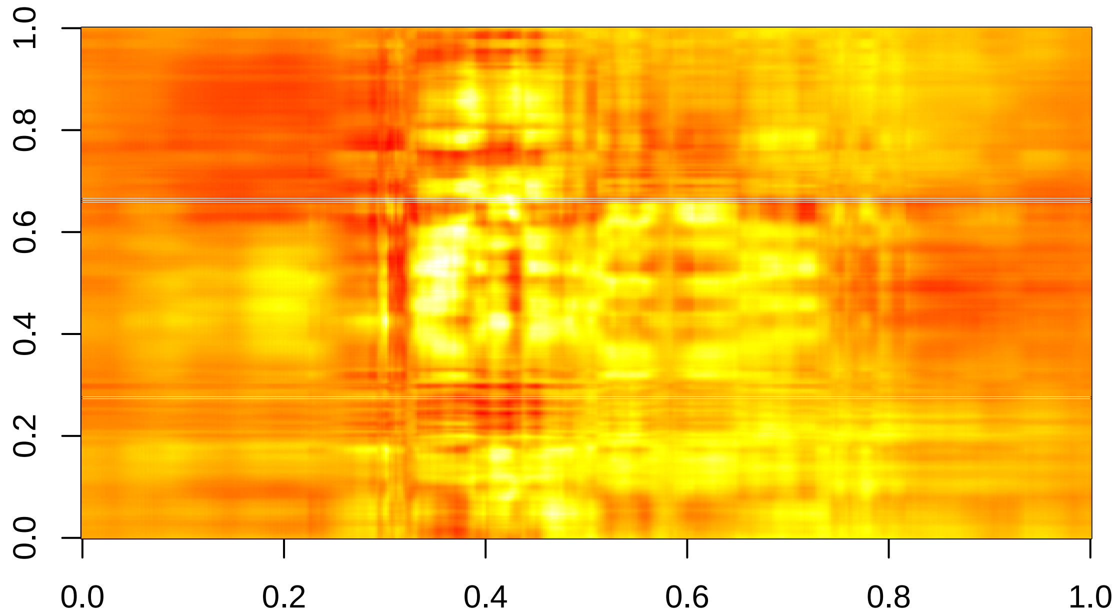


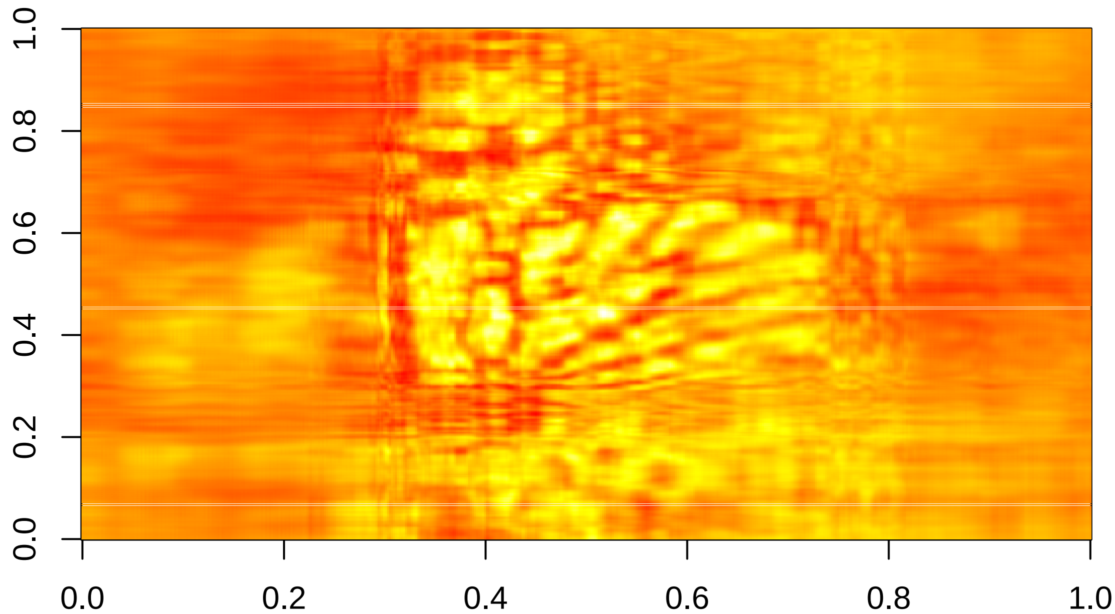

```
#by applying SVD to red matrix, it yields one diagonal and two orthogonal matrices
red_svd <- svd(red)
d <- red_svd$d
u <- red_svd$u
v <- red_svd$v

Decomposed_image <- u %*% diag(d) %*% t(v)
```

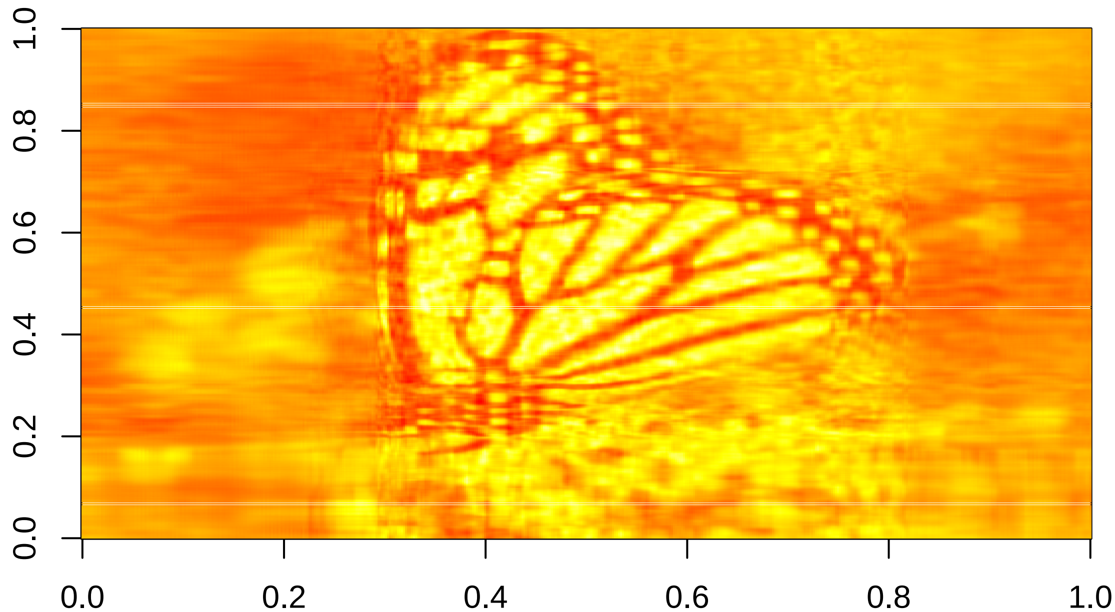
We can compare the images for singular column values.

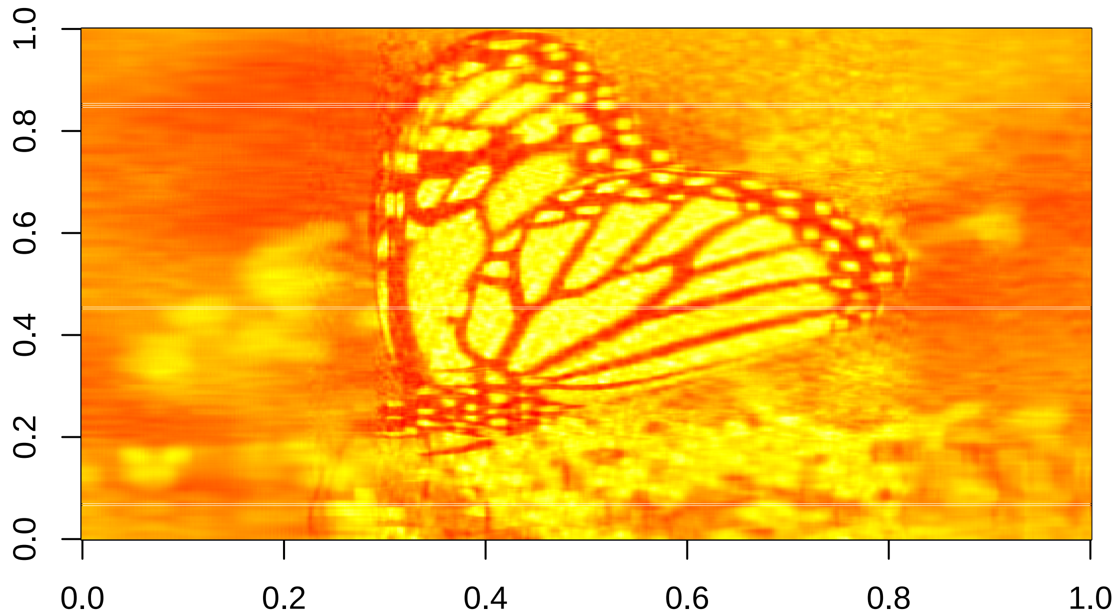
```
for (i in c(5, 10, 20, 30, 50))
{
  compressed_red_svd <- u[,1:i] %*% diag(d[1:i]) %*% t(v[,1:i])
  image(compressed_red_svd, col = heat.colors(255))
}
```

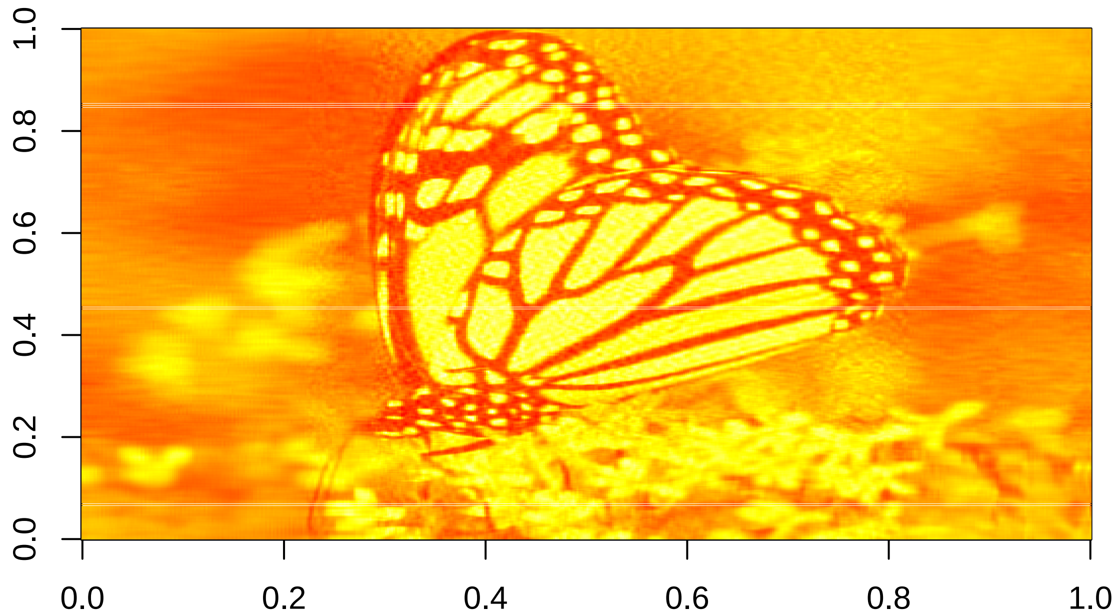




1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0


0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0


1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0



Conclusion:

By observing the above 5 images, we can say the fifth image is visibly much more clear,detailed and easily over the other. which means Most significant data is captured with less than 10% of image matrix data after decomposition.