Digital Image Processing - CS 663

Avyakta Wrat - 180070010 Nayan Barhate - 180070037
Meeta Malviya - 180070034 Varad Mane - 18D070034

Mean Shift Segmentation

Mean shift is a procedure for locating the maxima—the modes—of a density function given discrete data sampled from that function.[1] This is an iterative method, and we start with an initial estimate x. A function determines the weight of nearby points for re-estimation of the mean.

Formula used in segmentation is-
$\vec{r_i}$ is symbolic for all 3-D vector that of RGB space. $\vec{r}$ represents the 3-D RGB vector for which mean shift is being calculated.
$\vec{z_i}$ is symbolic for all 2-D vector that of the pair(x,y) of img. $\vec{Z}$ represents the 3-D RGB vector for which mean shift is being calculated.

$$f(x) = \alpha \sum_i K\left(\frac{||\vec{z_i} - \vec{z}||^2}{2h_s^2}\right) K\left(\frac{||\vec{r_i} - \vec{r}||^2}{2h_r^2}\right)$$

$$K(x^2) = \beta exp(-x^2/2)$$

$$f(x) = \alpha\beta^2 \sum_i exp\left(\frac{||\vec{z_i} - \vec{z}||^2}{2h_s^2}\right) exp\left(\frac{||\vec{r_i} - \vec{r}||^2}{2h_r^2}\right)$$

differentiating wrt to $\vec{r}$ and $\vec{z}$

$$\Delta f = \frac{\alpha\beta^2}{h_r^2} \sum_i (\vec{r_i} - \vec{r})\, exp\left(\frac{||\vec{z_i} - \vec{z}||^2}{2h_s^2} + \frac{||\vec{r_i} - \vec{r}||^2}{2h_r^2}\right)$$

$$\Delta log(f) = \frac{\Delta f}{f} = \frac{1}{h_r^2} \frac{\sum_i (\vec{r_i} - \vec{r}) exp\left(\frac{||\vec{r_i}-\vec{r}||^2}{2h_r^2} + \frac{||\vec{z_i}-\vec{z}||^2}{2h_s^2}\right)}{\sum_i exp\left(\frac{||\vec{r_i}-\vec{r}||^2}{2h_r^2} + \frac{||\vec{z_i}-\vec{z}||^2}{2h_s^2}\right)}$$

$$\Delta f = \frac{\alpha\beta^2}{h_s^2} \sum_i (\vec{z_i} - \vec{z})\, exp\left(\frac{||\vec{z_i} - \vec{z}||^2}{2h_s^2} + \frac{||\vec{r_i} - \vec{r}||^2}{2h_r^2}\right)$$

$$\Delta log(f) = \frac{\Delta f}{f} = \frac{1}{h_r^2} \frac{\sum_i (\vec{z_i} - \vec{z}) exp\left(\frac{||\vec{r_i}-\vec{r}||^2}{2h_r^2} + \frac{||\vec{z_i}-\vec{z}||^2}{2h_s^2}\right)}{\sum_i exp\left(\frac{||\vec{r_i}-\vec{r}||^2}{2h_r^2} + \frac{||\vec{z_i}-\vec{z}||^2}{2h_s^2}\right)}$$

hence the mean is shift to

$$\vec{m_r} = \frac{\sum_i \vec{r_i} exp\left(\frac{||\vec{r_i}-\vec{r}||^2}{2h_r^2}\right) exp\left(\frac{||\vec{z_i}-\vec{z}||^2}{2h_s^2}\right)}{\sum_i exp\left(\frac{||\vec{r_i}-\vec{r}||^2}{2h_r^2}\right) exp\left(\frac{||\vec{z_i}-\vec{z}||^2}{2h_s^2}\right)} - \vec{r}$$

$$\vec{m_s} = \frac{\sum_i \vec{z_i} exp\left(\frac{||\vec{r_i}-\vec{r}||^2}{2h_r^2}\right) exp\left(\frac{||\vec{z_i}-\vec{z}||^2}{2h_s^2}\right)}{\sum_i exp\left(\frac{||\vec{r_i}-\vec{r}||^2}{2h_r^2}\right) exp\left(\frac{||\vec{z_i}-\vec{z}||^2}{2h_s^2}\right)} - \vec{z}$$

$$\vec{z} = \vec{z} + k_r m_r(\vec{x})$$

$$\vec{r} = \vec{r} + k_s m_r(\vec{r})$$

Where k_s and k_r are scaling factor,
Main code-snippet is given below (when only spatial part is updated):-

```
for k=x_min_range : x_max_range
    for l=y_min_range : y_max_range
        %rgb space difference norm
        X = double((output_img(k,l,1) - output_img(i,j,1)) ^ 2 + (output_img(k,l,2) - output_img(i,j,2)) ^ 2 + (output_img(k,l,3) - output_img(i,j,3)) ^ 2);
        %spatial gaussian
        a = exp( - ((k - i) ^ 2 + (l - j) ^ 2) / (2 * (h_s)^2));
         %colour space gaussian
        b = exp(-(X/(2 * h_r^2)));
        %sum of weighted average in all color space
        for m=1:cha
            weighted_factor(m) = weighted_factor(m) + a * b * double((output_img(k,l,m)));
        end
          %sum (G_s * G_c)
        factor = factor + a * b;
    end
```

Steps Taken

1. Down sample the input image and apply Gaussian blurring
2. For each pixel $\vec{z}$ in the image we make a region of $4h_s$ around it (as any point greater than this window will have negligible weight)
3. Apply the above formula for each $\vec{z_i}$ & $\vec{r_i}$ in the window and calculate mean shift $\vec{m}$
4. Shift $\vec{r}$ by $\vec{m_r}$ & $\vec{z}$ by $\vec{m_s}$
5. Repeat last three steps for few iterations

Without spatial update

Flower

1. Bandwidth for the color feature = 10
2. Bandwidth for the spatial feature = 6
3. number of iterations = 15

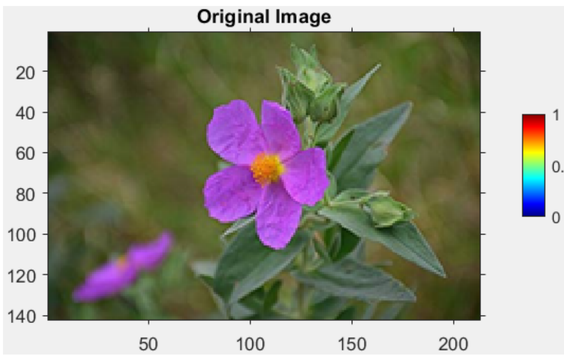

Figure 1: original image

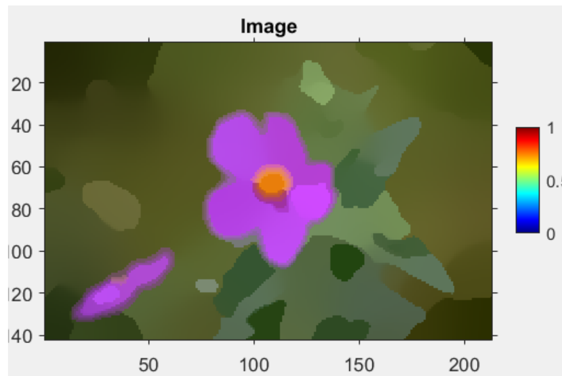

Figure 2: segmented image

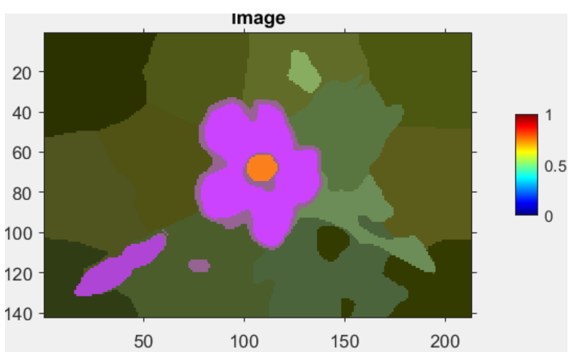

Figure 3: clustered image

Bird

1. Bandwidth for the color feature = 30
2. Bandwidth for the spatial feature = 20
3. number of iterations = 15

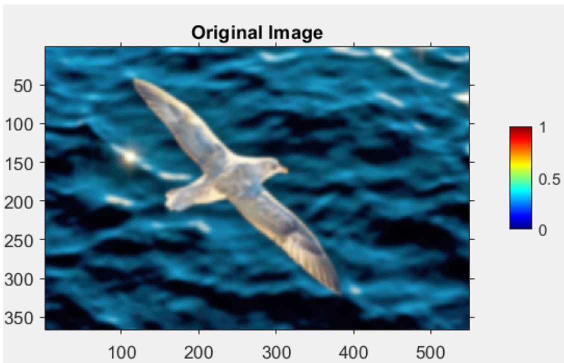

Figure 4: original image

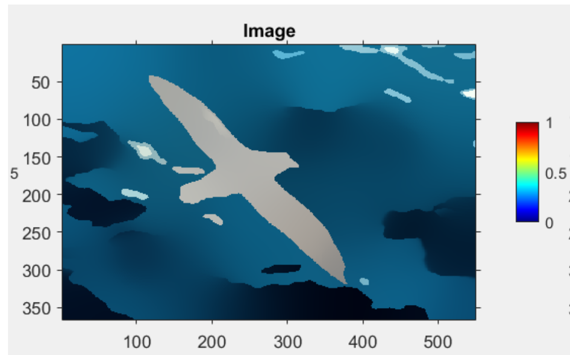

Figure 5: segmented image

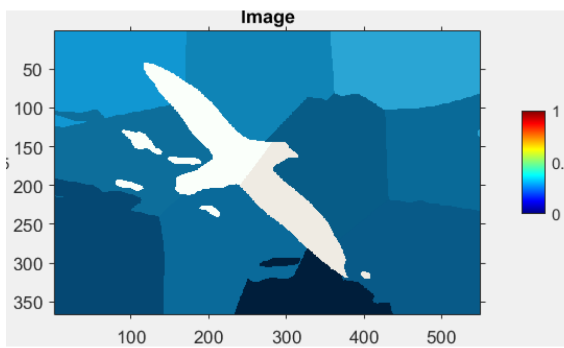

Figure 6: clustered image

Baboon

1. Bandwidth for the color feature = 16
2. Bandwidth for the spatial feature = 8
3. number of iterations = 15

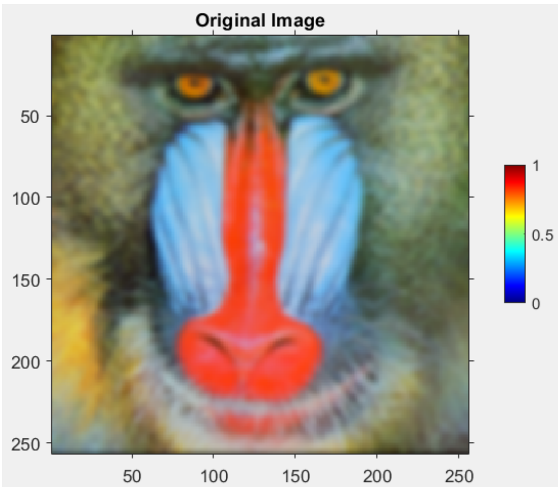

Figure 7: original image

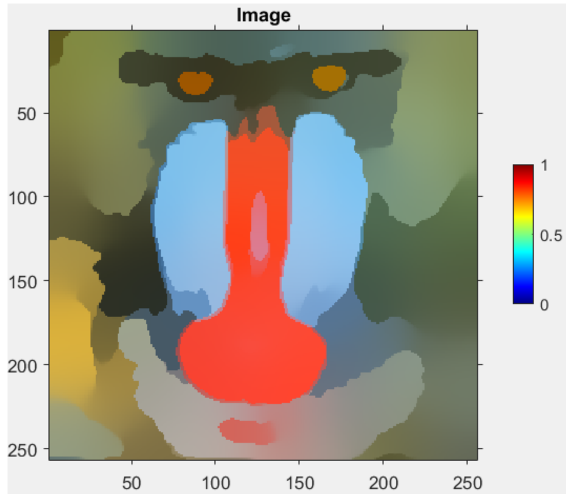

Figure 8: segmented image

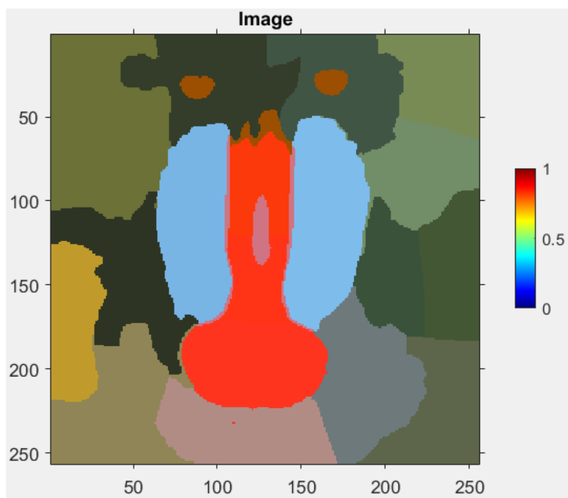

Figure 9: clustered image

With spatial update

Flower

1. Bandwidth for the color feature = 10
2. Bandwidth for the spatial feature = 6
3. number of iterations = 15

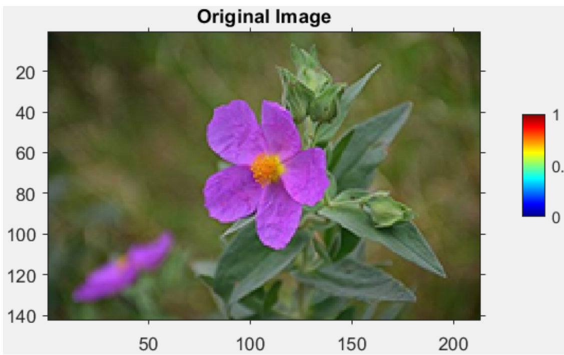

Figure 10: original image

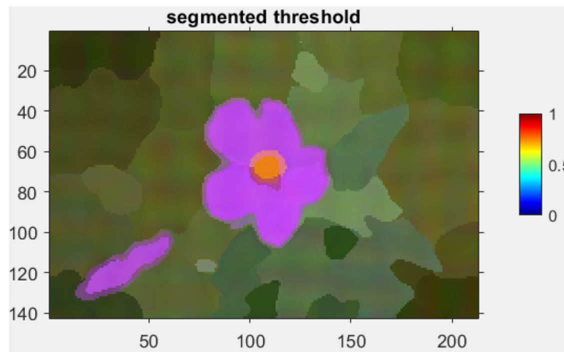

Figure 11: segmented image

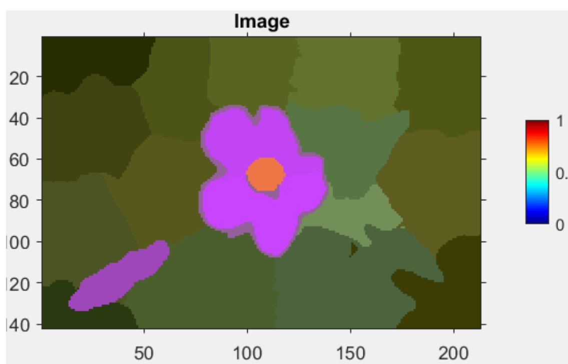

Figure 12: clustered image

Bird

1. Bandwidth for the color feature = 30
2. Bandwidth for the spatial feature = 20
3. number of iterations = 15

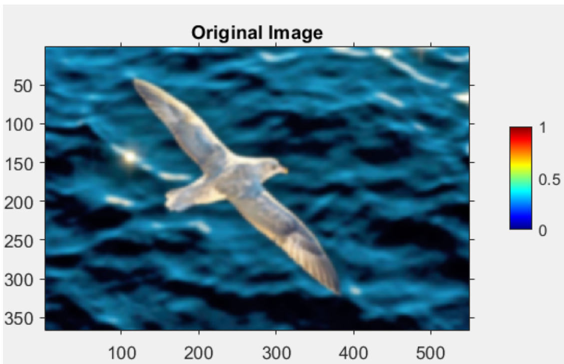

Figure 13: original image

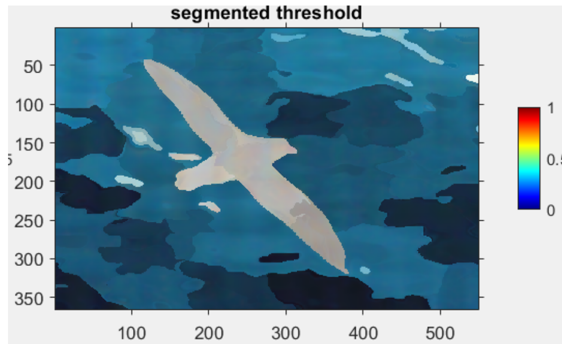

Figure 14: segmented image

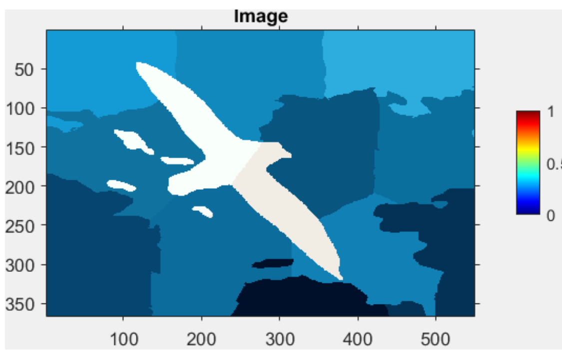

Figure 15: clustered image

Baboon

1. Bandwidth for the color feature = 15
2. Bandwidth for the spatial feature = 15
3. number of iterations = 15

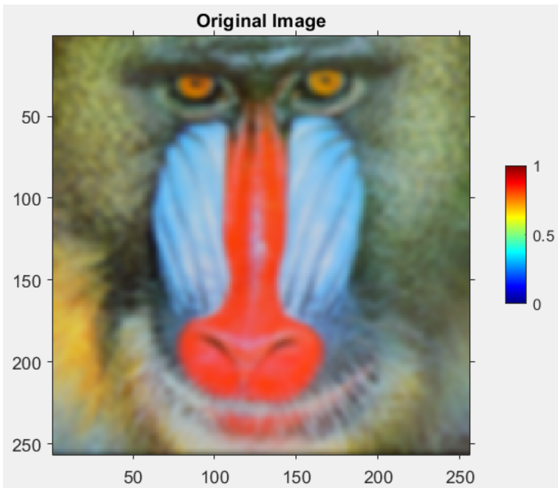

Figure 16: original image

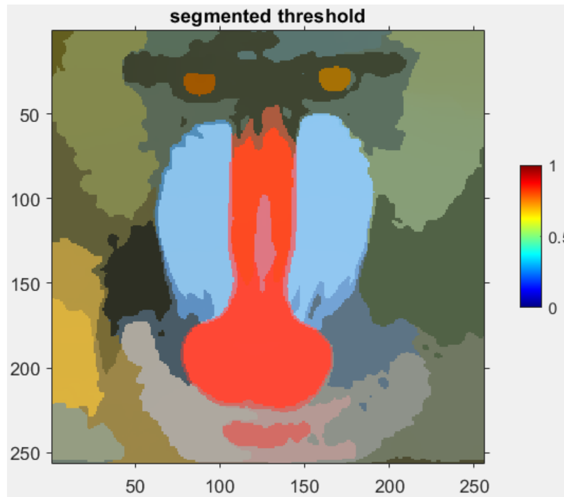

Figure 17: segmented image

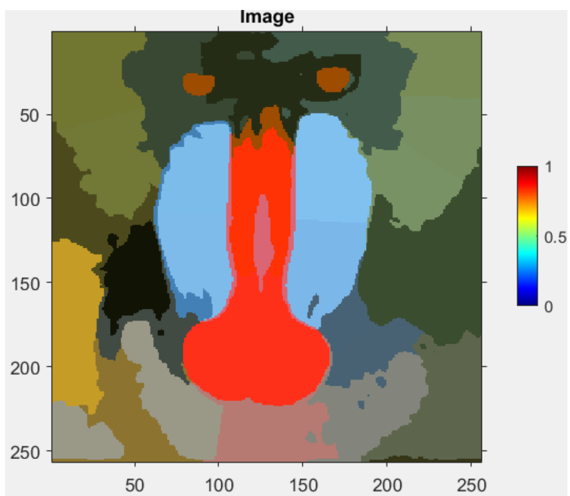

Figure 18: Clustered Image

Remarks

1. The results are better in case of 'without spatial coordinates update'
2. Spatial coordinates update introduces isometric patches.
3. Clustering is done by using kmeans and the colour of patch is the colour of centroid.
4. Introducing learning rate (scaling factor) help us to achieve convergence faster.