# Image Segmentation

Chao Ni

December 16, 2020

In this exercise we will implement image segmentation with the mean-shift and Expectation-Maximization approaches.

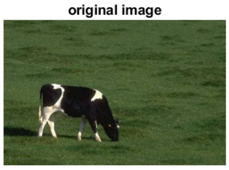
## 1 Image Preprocessing

### 1.1 Image smoothing

In order to avoid the jumpy pixels of the image and allow the clustering to be smooth, we need to smooth the picture with a Gaussian filter. The smoothed image as well as the original images are shown in Fig. 1.

### 1.2 Color space

We will use L*a*b* color space to apply in the image segmentation. L* represents lightness of the color, and $a*$ represents the color position between red and green while b* between blue and yellow. We prefer this over RGB because human eyes tend to cluster pixels with similar color but different lightness together, as is the case in using L*a*b* color space. There is a paper comparing performance in image segmentation for all color spaces, the difference is minor but L*a*b* is almost always the best option. [1]




(a) Original image


(b) Gaussian kernel(window: 5, $\sigma$: 5)


(c) L*a*b* color space

Figure 1: Image Preprocessing: smoothing and transforming color space

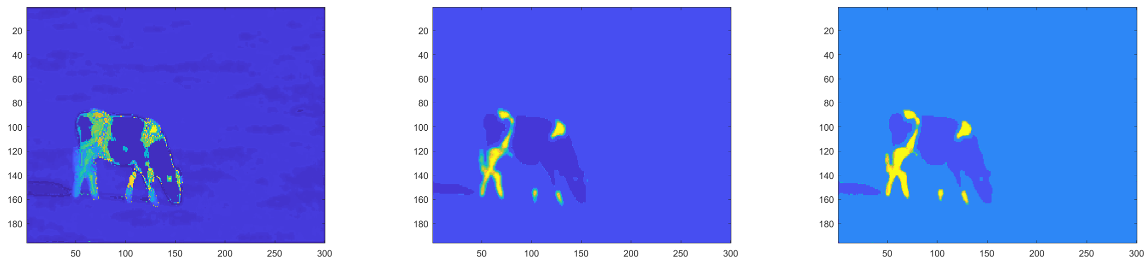
## 2 Mean-Shift Segmentation

The idea of Mean-Shift segmentation is to assign each pixel a mode that it belongs to. All similar pixels that should be clustered together should finally reach locally to the same mode that they belong to. The difference is color-wise instead of pixel position-wise. Therefore it allows for arbitrary shape of segmentation. We can control the radius within which the mean will be computed and be guiding the shift direction to control the cluster number. Therefore this radius has physical meaning in some sense. With different radius, we have the results shown in Fig. 2,3.

## 3 EM Segmentation

The idea of EM algorithm is to assign each pixel a probability to all possible clusters. We assume a Gaussian distribution and update the mean and covariance matrix in an iterative fashion until conver-

[1] https://www.cv.tu-berlin.de/fileadmin/fg140/VISAPP_2014_127_CR.pdf




(a) radius: 5, clusters: 258 (b) radius: 15, clusters: 15 (c) radius: 25, clusters: 7

Figure 2: Mean-Shift: Segmentation result with color assignment

(a) radius: 5, clusters: 258 (b) radius: 15, clusters: 15 (c) radius: 25, clusters: 7

Figure 3: Mean-Shift: Image reconstruction

gence. Number of clusters has to be assigned manually. We set the convergence criteria as that when the norm of mean update is below 0.1, then the optimization stops. We compare the results with different number of clusters and show them below.

## 3.1 Parameters

### 3.1.1 $K = 3$, $tol = 0.1$, 21 iterations

- $\mu = \begin{bmatrix} 48.9383 & 121.3023 & 139.5648 \\ 89.4219 & 114.4443 & 149.0918 \\ 133.2018 & 124.8240 & 140.9921 \end{bmatrix}$, $\mu(k,:) = \mu_k$

- $\Sigma_1 = \begin{bmatrix} 727.1606 & -140.2448 & 217.6935 \\ -140.2448 & 36.2716 & -48.5120 \\ 217.6935 & -48.5120 & 73.4516 \end{bmatrix}$, $\Sigma_2 = \begin{bmatrix} 53.0642 & 0.0465 & 0.5164 \\ 0.0465 & 0.7787 & -0.1617 \\ 0.5164 & -0.1617 & 1.5173 \end{bmatrix}$,

  $\Sigma_3 = 10^3 \begin{bmatrix} 2.4143 & 0.0836 & 0.0201 \\ 0.0836 & 0.0134 & -0.0097 \\ 0.0201 & -0.0097 & 0.0271 \end{bmatrix}$

- $\alpha = [0.1514, 0.8076, 0.0410]$.

### 3.1.2 $K = 4$, $tol = 0.1$, 97 iterations

- $\mu = \begin{bmatrix} 41.9683 & 122.9127 & 136.9726 \\ 92.7809 & 114.5319 & 148.4501 \\ 85.6858 & 114.3939 & 149.6783 \\ 130.8314 & 124.5585 & 141.3042 \end{bmatrix}$, $\mu(k,:) = \mu_k$

- $\Sigma_1 = \begin{bmatrix} 648.3679 & -121.6964 & 180.5023 \\ -121.6964 & 31.4746 & -40.6672 \\ 180.5023 & -40.6672 & 58.7971 \end{bmatrix}$, $\Sigma_2 = \begin{bmatrix} 17.4452 & 0.1532 & -0.0247 \\ 0.1532 & 0.3940 & -0.0279 \\ -0.0247 & -0.0279 & 0.4884 \end{bmatrix}$,

$$\Sigma_3 = \begin{bmatrix} 74.4036 & -0.7392 & 5.6439 \\ -0.7392 & 1.3728 & -0.1578 \\ 5.6439 & -0.1578 & 1.8471 \end{bmatrix}, \Sigma_4 = 10^3 \begin{bmatrix} 2.4506 & 0.0929 & 0.0044 \\ 0.0929 & 0.0146 & -0.0109 \\ 0.0044 & -0.0109 & 0.0278 \end{bmatrix}$$

- $\alpha = [0.1204, 0.3848, 0.4520, 0.0428]$.

#### 3.1.3 $K = 5$, $tol = 0.1$, 75 iterations

- $\mu = \begin{bmatrix} 15.1448 & 128.6472 & 128.5771 \\ 92.7333 & 114.5387 & 148.4385 \\ 85.8079 & 114.3463 & 149.7269 \\ 64.9654 & 118.0145 & 143.8554 \\ 113.9177 & 125.9896 & 138.9904 \end{bmatrix}$, $\mu(k,:) = \mu_k$

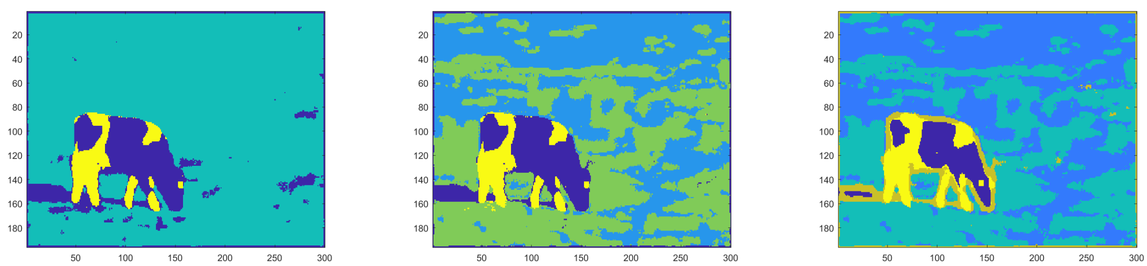
- $\Sigma_1 = \begin{bmatrix} 15.6271 & 3.0910 & -0.6288 \\ 3.0910 & 2.4988 & -0.8932 \\ -0.6288 & -0.8932 & 2.9353 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 17.5842 & 0.1210 & -0.0383 \\ 0.1210 & 0.3859 & -0.0206 \\ -0.0383 & -0.0206 & 0.4670 \end{bmatrix},$

$$\Sigma_3 = \begin{bmatrix} 73.6864 & -0.4052 & 5.3702 \\ -0.4052 & 1.3623 & -0.0490 \\ 5.3702 & -0.0490 & 1.75611 \end{bmatrix}, \Sigma_4 = \begin{bmatrix} 380.4677 & -32.8830 & 66.8160 \\ -32.8830 & 6.3929 & -9.0226 \\ 66.8160 & -9.0226 & 17.6776 \end{bmatrix},$$

$$\Sigma_5 = 10^3 \begin{bmatrix} 3.6734 & 0.0020 & 0.1606 \\ 0.0020 & 0.0099 & -0.0097 \\ 0.1606 & -0.0097 & 0.0314 \end{bmatrix}$$

- $\alpha = [0.0434, 0.3813, 0.4457, 0.0804, 0.0491]$.

## 3.2 Plotting




(a) clusters: 3 (b) clusters: 4 (c) clusters: 5

Figure 4: EM: Segmentation result with color assignment

(a) clusters: 3 (b) clusters: 4 (c) clusters: 5

Figure 5: EM: Image reconstruction

### 3.3 Discussion

We will discuss the results in term of some observations:

- Basically, the weight coincides with the area of the corresponding color;
- The performance is much better than Mean-shift approach at the cost of higher burden of computation;
- If we compare $K = 3$ and $K = 4$, we will see the background is split into two colors. It is corresponding to the fact that the first mean and last mean stays roughly the same, while the middle mean is split into two, and also the weight is split into two smaller weights.
- From $K = 4$ to $K = 5$, the edge of cow is further segmented out and given a weight 0.0434.