# Spatial Statistics and Image Analysis

## Lecture 6

Konstantinos Konstantinou

Mathematical sciences
Chalmers University of Technology and University of Gothenburg
Gothenburg, Sweden

# Lecture's content

Todays lecture will cover

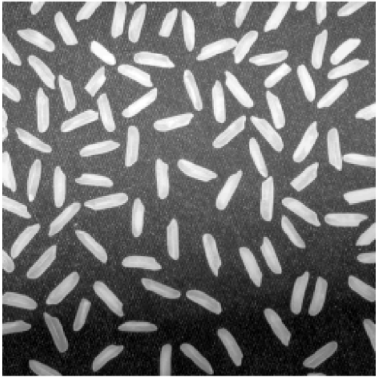
- Unsupervised methods for image segmentation
  1. K-means
  2. Gaussian mixture models
- Morphological operations
- Feature extraction

# Image segmentation

- So far we looked at **supervised methods** for image classification, that is we have access to the labels $Z_1$,..., $Z_N$ for each image in the training set, which we then use to train a classifier.
- Now we will study some **unsupervised methods** for image segmentation into $K$ different classes without having any label information.

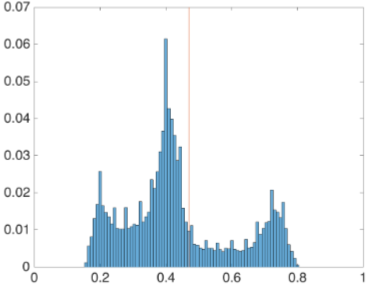
# Image segmentation example

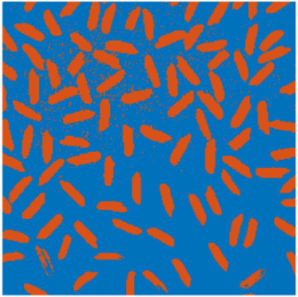
Goal: group the unlabeled data (pixel values) into K different classes (rice or not rice).



# Intensity-based thresholding

- Already in lab 1 we have seen how to segment an image using its histogram and choosing a reasonable threshold.



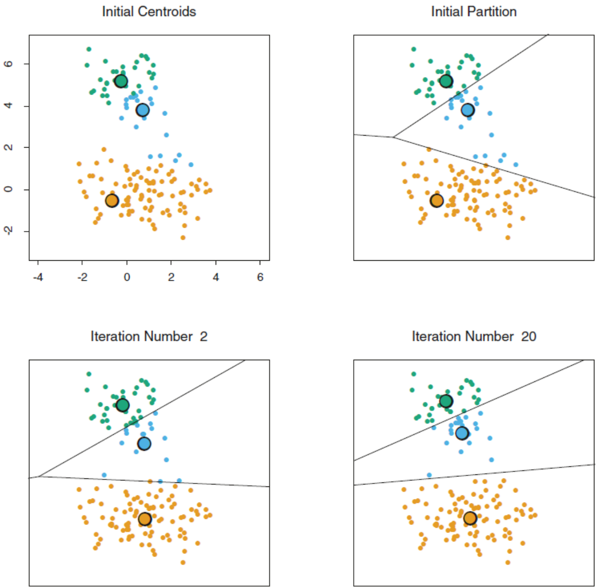



# K-means

- Recall that in a colour image the set of possible pixel values is $V = \{0, ..., 255\}^3$.
- The K-means algorithm:
  1. Randomly select $K$ observations as cluster centres
  2. Assign each observation to the closest cluster centre.
  3. Compute the mean of each cluster and assign these as new cluster centres
  4. Repeat from step 2 until convergence.

  Typically you repeat this procedure a number of times with different starting cluster centres and choose the clustering that has the minimum total variation within the classes.

# Illustration of the K-means



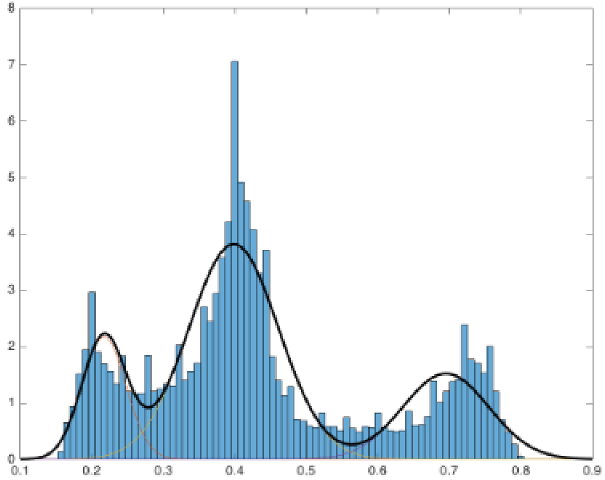
# Gaussian mixture models

- A Gaussian mixture model is based on the assumption that the observations in the data come from different classes, for which the distributions of the observations come from different Gaussians.
- Let K be the number of classes and $Z_i$ denote the class membership of $X_i$. Then density of $X_i$ is given by

$$P(X_i = x) = \sum_{k=1}^{K} P(Z_i = k) P(X_i = x \mid Z_i = k) = \sum_{k=1}^{K} \pi_k N(x; \mu_k, C_k)$$

where $\pi_k = P(Z_i = k)$ is the probability that $X$ belongs to class $k$. As the data are unlabeled $Z_i \in \{1, ..., K\}$ is a latent variable.

# Example of GMM with $K = 3$



# MLE for GMM

Let $\theta = (\{\pi_k\}_{k=1}^K, \{\mu_k\}_{k=1}^K, \{C_k\}_{k=1}^K)$ be the parameters of the GMM. Then the log-likelihood is

$$\ell(\theta \mid x_1, ..., x_n) = \sum_{i=1}^{n} log(\sum_{k=1}^{K} \pi_k N(x_i; \mu_k, C_k))$$

- The summation over k within the logarithm make it impossible to get a closed form solution for $\theta$.
- We will use the Expectation-Maximization(EM) algorithm to estimate the parameters $\theta$ of the GMM.

# EM algorithm for GMM

- Idea: If we knew the latent variables $Z_i$ then estimation of the parameters $\theta$ will be very simple.
- Instead of the likelihood we will use the complete data log likelihood given by

$$logP(X,Z) = \sum_{i=1}^{n} logP(X_i, Z_i) = \sum_{i=1}^{n} logP(X_i \mid Z_i)P(Z_i)$$

$$= \sum_{i=1}^{n} log \prod_{k=1}^{K} [\pi_k N(x_i; \mu_k, C_k)]^{1\{Z_i = k\}}$$

$$= \sum_{i=1}^{n} \sum_{k=1}^{K} 1\{Z_i = k\}[log\pi_k + logN(x_i; \mu_k, C_k)]$$

# EM algorithm

- Unfortunately $Z_i$ are not known BUT we can use posterior expectations for $1\{Z_i = k\}$ using the current estimates for the parameters $\theta^*$
- Using the Bayes rule we get that

$$E_{Z_i|X,\theta^*}(1\{Z_i = k\}) = P(Z_i = k \mid X_i = x_i, \theta^*) = \frac{\pi_k^* N(x_i; \mu_k^*, C_k^*)}{\sum_{j=1}^{K} \pi_j^* N(x_i; \mu_j^*, C_j^*)}$$

- Using the posterior expectations $E_{Z_i|X,\theta^*}(1\{Z_i = k\})$ instead of the unknown values of $1\{Z_i = k\}$ we can estimate the parameters $\theta$ using the expected complete data log likelihood [E-step].

$$Q(\theta \mid \theta^*) = E_{Z|X,\theta^*}(logP(X, Z \mid \theta))$$

$$= \sum_{i=1}^{n} \sum_{k=1}^{K} E_{Z|X,\theta^*}(1\{Z_i = k\})[log\pi_k + logN(x_i; \mu_k, C_k)]$$

- Then we update the parameters $\theta^* = argmax(\theta \mid \theta^*)$ [M-step]
- EM is an iterative algorithm that starts from some initial parameters $\theta^*$ and then iteratively updates $\theta^*$ until convergence ($\ell(\theta \mid \mathbf{x})$ increases at each iteration).

# EM algorithm for GMM

- Each iteration consist of two steps:
  1. E step: Given the current parameters $\theta^*$ we estimate the probabilities that $X_i$ belong in cluster $k$, $p_{ik} = \frac{\pi_k^* N(x_i; \mu_k^*, C_k^*)}{\sum_{j=1}^{K} \pi_j^* N(x_i; \mu_j^*, C_j^*)}$ $\forall i \in \{1, ...N\}$, $\forall k \in \{1, ..., K\}$.
  2. M step: We update the current parameters $\theta^*$ using the values $p_{ik}$.

$$\pi_k = \frac{\sum_i p_{ik}}{N}, \quad k = 1, ..., K$$

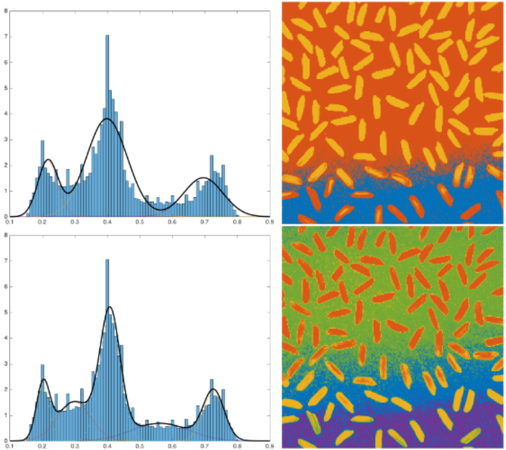
$$\mu_k = \frac{1}{\sum_i p_{ik}} \sum_i p_{ik} x_i, \quad k = 1, ..., K$$

$$C_k = \frac{1}{\sum_i p_{ik}} \sum_i p_{ik} (x_i - \mu_k)^T (x_i - \mu_k), \quad k = 1, ..., K$$

We iterate until convergence.

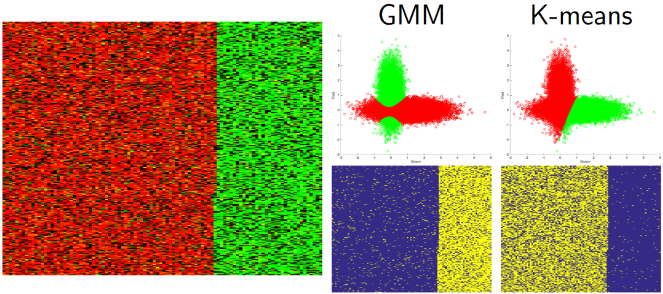
# Example: GMM with K=3 and K=5

Image segmentation using GMM with K=3 (top) and K=5 (bottom).



# Comments

- The K-means clustering procedure is closely related to the EM algorithm for estimating a Gaussian mixture model with $\pi_k = \frac{1}{K}$ and $C_k = \sigma^2\mathcal{I}$. Those assumptions are generally not satisfied.



# Comments

- A GMM does not take spatial dependencies into account.
- The classes may have additional features except for raw pixel values which we may want to use.
- We will extend the mixture model to take into account possible dependencies.
- Markov field mixture model:

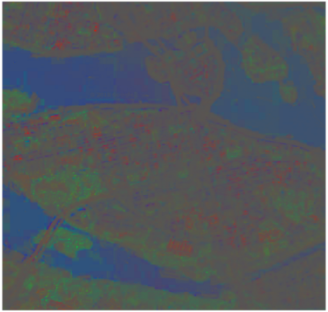
$$P(X_i \mid Z_i = k) \sim N(x; \mu_k, C_k)$$

$$Z \sim P(Z)$$

  where Z is a random field taking the values $\{1, ..., K\}$ with density $P(Z)$.
- We can model spatial dependencies through $P(Z)$
- Undesired features of the image such as shadows or background spatial trend might influence the segmentation.

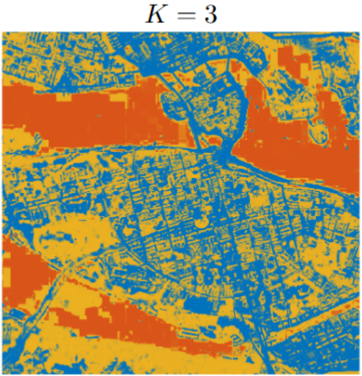
# Relative colours

Shadows might influence the image segmentation. One way to overcome this is transform the colour space. For example do the segmentation using relative colours or LAB colours. The relative amount of green in a pixel is G/(R+B+G).



# Image segmentation using relative colours

Image segmentation using a GMM with $K = 3$ on the original image (left) and using the relative color image (right). The water area (orange) is better classified using the relative color image than the original image.



# Morphological operations on binary images

Morphological operations can be used to regularize or clean binary images. Let $A$ be a set of pixels in an image, and let $S_{ij}$ be a structuring element centered in pixel $ij$.

- Erosion of A: $A \ominus S_{ij} = \{ij : S_{ij} \subset A\}$.
- Dilation of A: $A \oplus S_{ij} = (A^c \ominus S)^c$, where $A^c$ is the complement of $A$.
- Opening of A: $\Psi_S(A) = (A \ominus S) \oplus S'$, where $S'$ is $S$ rotated 180 degrees.
- Closing of A: $\Phi_S(A) = (A \oplus S) \ominus S'$

# Example for image erosion and dilation.

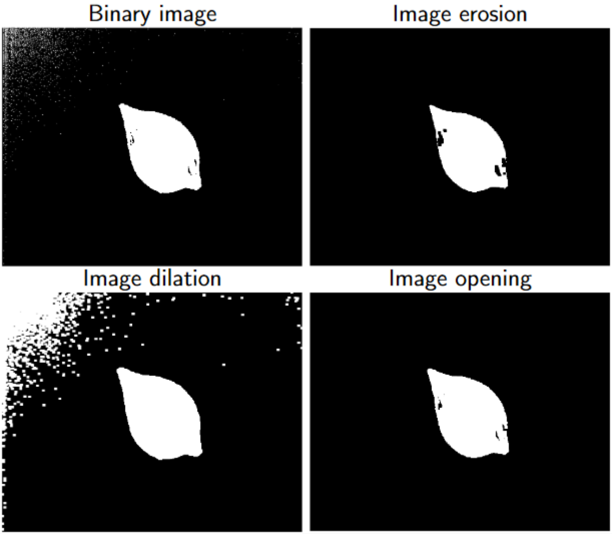
Let S= $\begin{pmatrix} & 1 & \\ 1 & \color{red}{1} & 1 \\ & 1 & \end{pmatrix}$ be the structuring element where red color denotes the origin and a binary image A= $\begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}$. Then

- $A \ominus S = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$
- $A \oplus S = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$

# Morphological operations

- Erosion: decreases the size of an object and removes the objects with a radius smaller than the structuring element.
- Dilation: Increases the size of an object, fills holes and gaps, increase the size of small objects.
- Opening: Removes small white objects.
- Closing: Removes small black objects.

# Morphological operations example



# Morphological operations on grayscale images

Morphological operations can be extended for grayscale images. Let $A$ be a grayscale image, and $S$ a structuring element.Then

- Erosion of A: $(A \ominus S)_{ij} = min(A_{kl} : kl \in S_{ij})$
- Dilation of A: $A \oplus S_{ij} = max(A_{kl} : kl \in S_{ij})$
- Opening of A: $\Psi_S(A) = (A \ominus S) \oplus S'$
- Closing of A: $\Phi_S(A) = (A \oplus S) \ominus S'$

# Morphological operations examples

Binary image

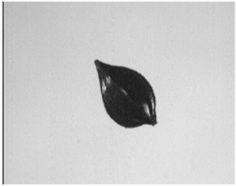


Image erosion

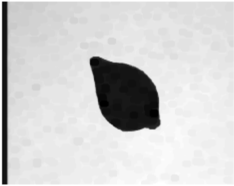


Image dilation

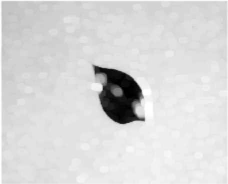
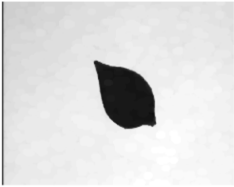


Image opening



# Feature extraction

After the image segmentation part we might be interested to calculate some descriptors of the detected object $A$. For example:

- $Area(A)$= number of pixels in $A$.
- $Perimeter(A)$ = number of pixels in $A$ for which at least one of the eight neighboring pixels is in $A^c$
- $Compactness(A) = 4\pi \frac{Area(A)}{(Perimeter(A))^2}$
- $ConvexArea(A) = Area(B)$, where B is the convex hull of A
- $ConvexPerimeter(A)$=$Perimeter(B)$
- $Convexity(A) = \frac{ConvexPerimeter(A)}{Perimeter(A)}$
- and more.. (see Chapter 6 in Glasbey and Horgan 1995)