EE-451 - Image analysis and pattern recognition – Prof. Jean-Philippe Thiran Spring 2022

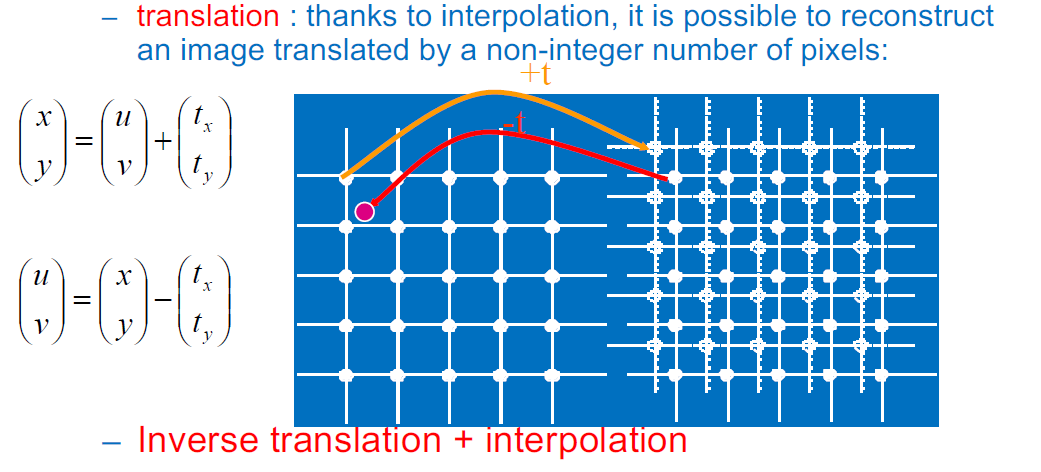
List of questions for the interviews

Part 1 – long answers

1. **Present how to perform geometrical transformations of a digital image? Take the example of a translation of a non-integer number of pixels. L1 s12**

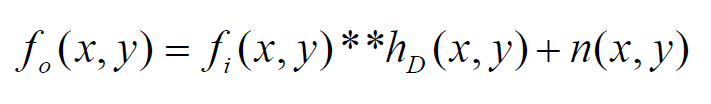
Main idea: leave the grid where it is and find the values of pixels s.t. the image looks transformed.

Example: Translation of a non-integer number of pixels: find place of the red pixel in the original image by inverse transformation and interpolate the value of that pixel to put this value back into the new image (free choice of interpolation depending on computational power, speed needed, accuracy needed. Ideal: sinc). Nonlinear transformation can also be performed with the same procedure such as polynomial transformations.

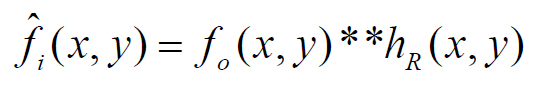


1. **What is image restoration? On this context what is inverse filtering and what is a Wiener filter? L1 s36**

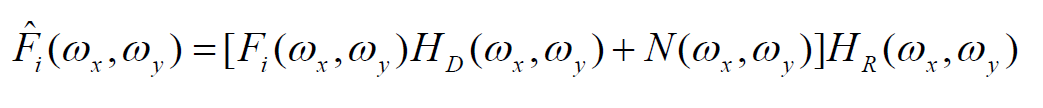
Observed degraded image:



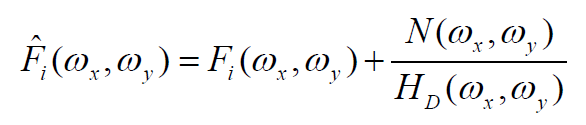
Restored image:



Fourier transform:



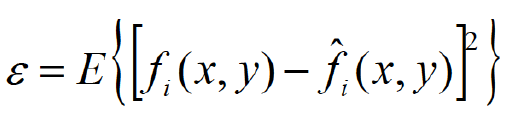
Naive inverse filter:



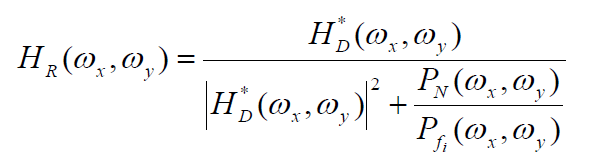
Hd often low-pass (blur) 🡪 1/Hd amplifies the high frequencies of the noise

But if no noise: ideal filter!

Wiener Filter: We want to minimize the expected reconstruction error



After some calculation we get



Pn power spectrum of the noise

Pfi power spectrum of the image

Adaptive band-pass filter:

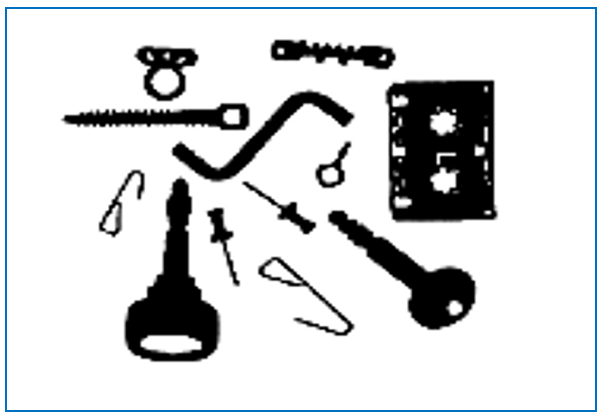
* Large noise 🡪 Hr vanishes (attenuates noisy frequencies)
* Little noise 🡪 1/Hd ideal inverse filter for low-noise frequencies

Estimation of Hd and noise power spectrum Pn

* Pn can be estimated from uniform region of the image, remove mean and compute FT of autocorrelation function
* Hd is the impulse response, can be estimated by a region of the image containing something looking like a dirac (star in the sky) or by taking a region of the image where there is a sharp edge to obtain an estimation of the step response in Fourier domain and derive it to get the impulse response.

1. **Explain what object labeling is and the algorithm to implement it. L2 s15**

Object labeling aims at separating semantically different objects in an image by labeling the pixels of that image. Objects are defined by connex regions that are below a certain predefined threshold.

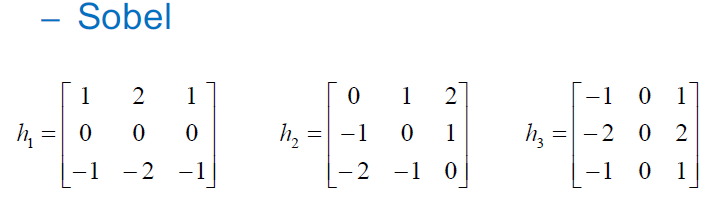


Scan the image from the top left (0,0) until finding a pixel under the threshold. Find all connected pixels (connex region) under threshold with region growing and assign them the same label, then start scanning again unlabeled pixels. Do so until all pixels are all labeled.

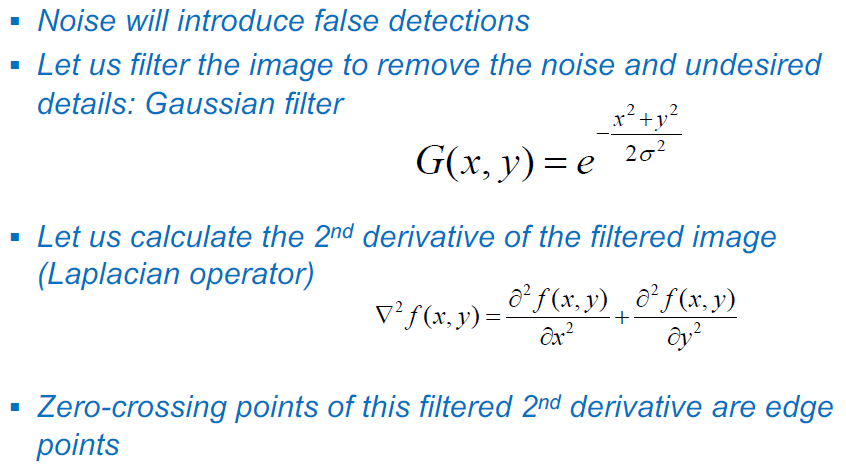
1. **What are the main principles of edge detection, and the two main families of methods to do edge detection? Present typical methods for each family. L2 s20**

Principle: identify edges in an image at which the intensity sharply changes, i.e. when the derivative has a maximum.

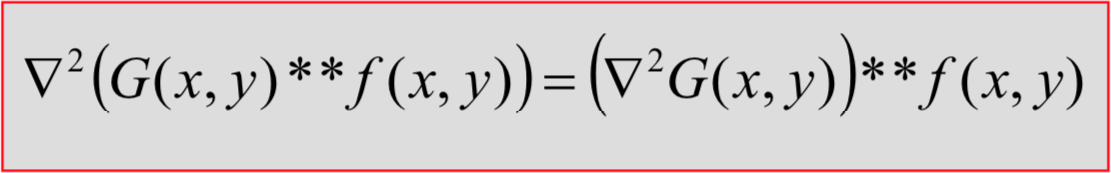
1. Maximum of first derivative



1. Laplacian of Gaussian (LoG):



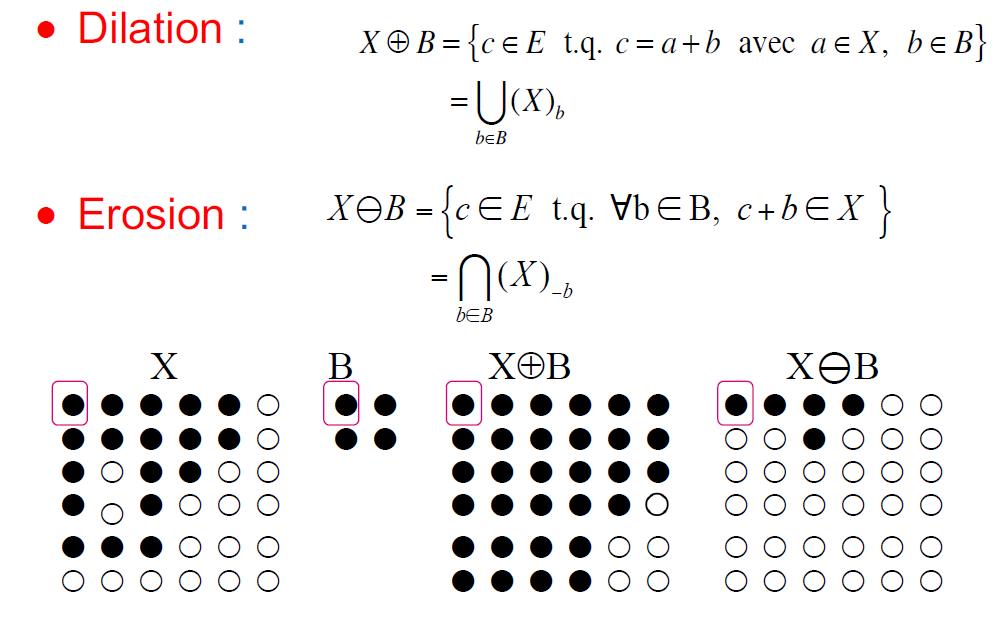
Trick: Exploit property of convolution:



* Convolve image with the second derivative of a Gaussian and find zero-crossing to detect edges

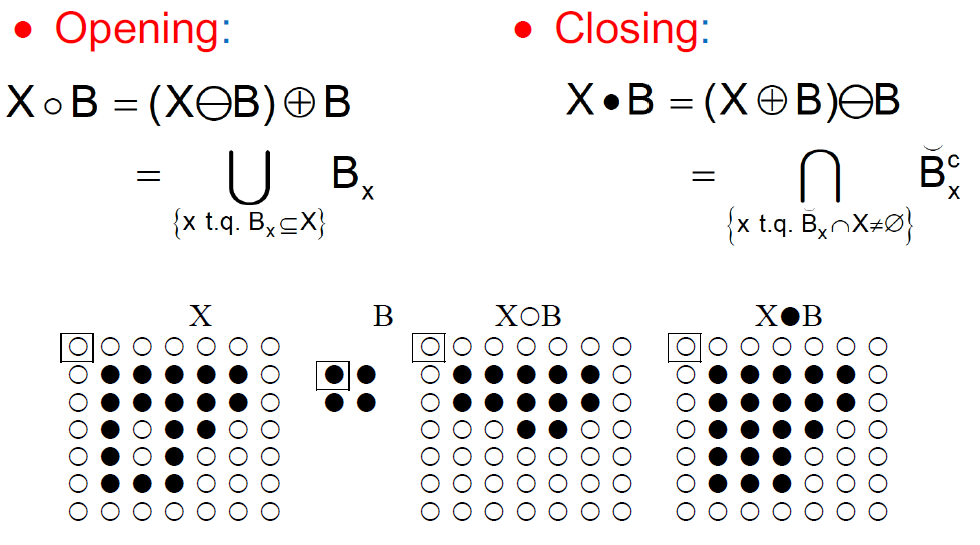
1. **What are the 4 main operators of binary mathematical morphology? Explain each of them. L2 s28-37**

* Dilation & erosion (change the size of the object)
* Opening and closing (fill holes inside or external to the object)



Dilation: Expansion of the object by the structural element. Fills holes that are smaller than the structuring element. Increases the size of the object

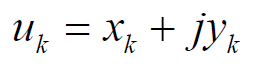
Erosion: Removes parts of the object where the structuring element doesn’t fit entirely. Decreases the size of the object.



Opening: Erosion then Dilation. Remove the external parts that are smaller than the structuring element. Opened object is a subset of the original.

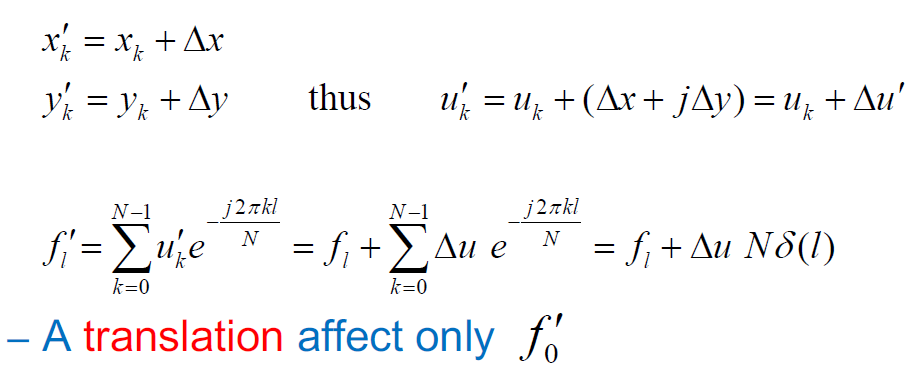
Closing: Dilation then Erosion. Fill holes in the objects that are smaller than the structural element. Closed object is a superset of the original.

1. **What are the Fourier descriptors? L4 s22-30**

Feature based on the Fourier transform of the contour of an objet. Define contours as complex numbers: 

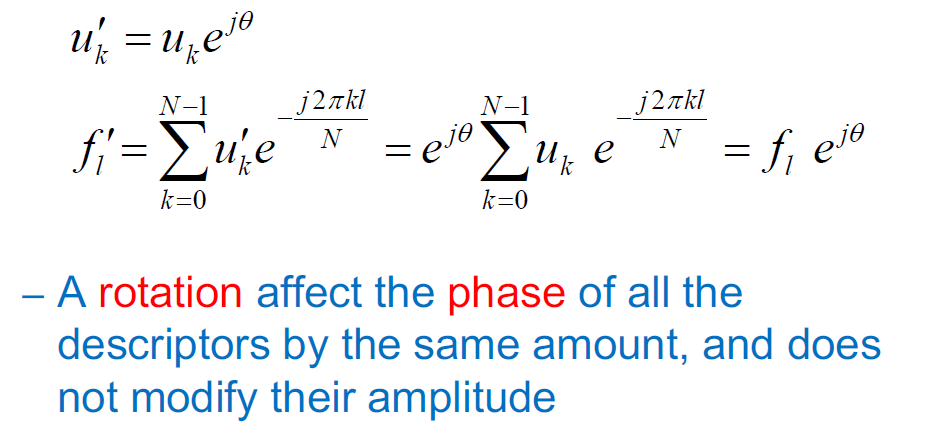
Then compute DFT: The first Fourier descriptors contains most of the information on the shape of the object.

Translation invariance:



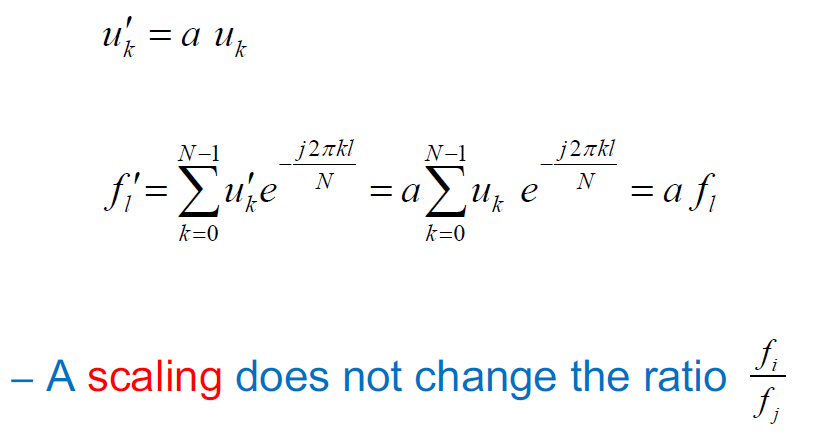
* Don’t take the first coefficient into account

Rotation invariance:



* Ignore the phase of all descriptors

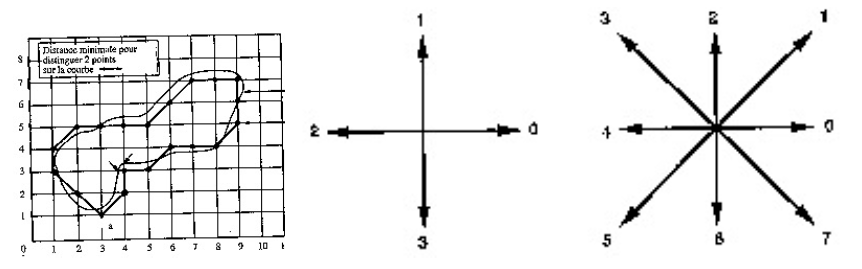
Scale invariance:



* Normalize the Fourier coefficient

1. **What is a Freeman code? L3 s11**

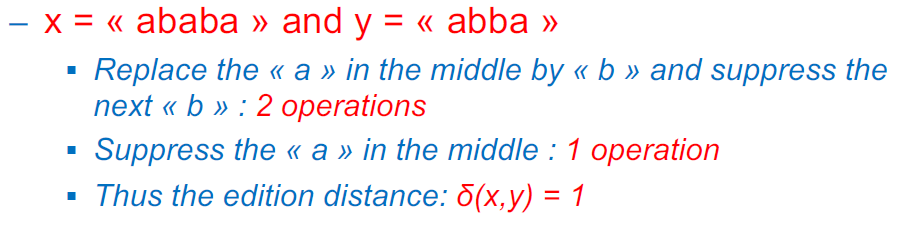
Idea: encode the contour thanks to labeled directions (1,2,3,…)



Compact representation, directions encoded in 3 bits only.

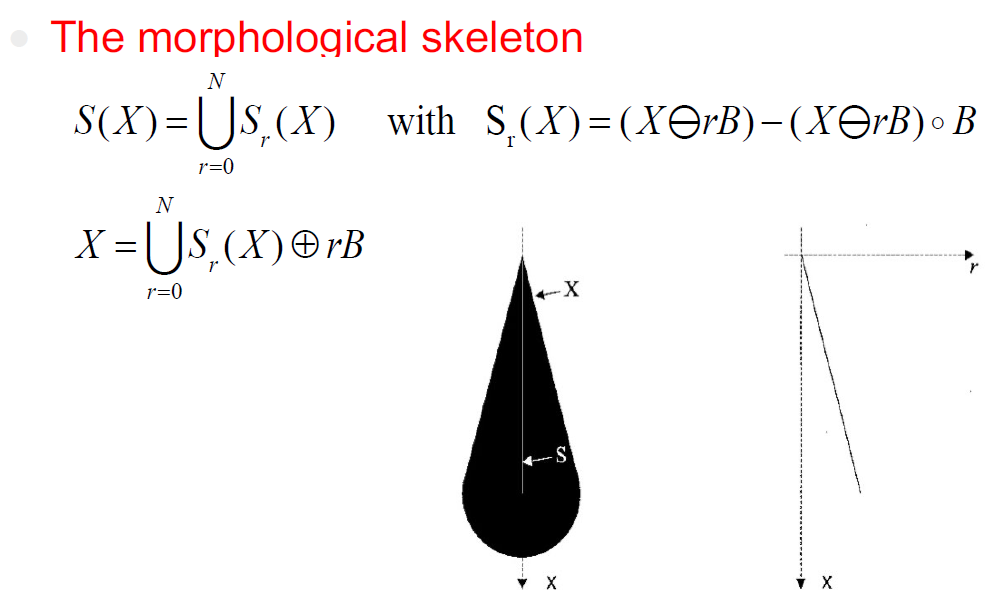
Compare object by computing similarity between codes of 2 different objects:

Edition distance: Minimum number of elementary operations to go from one chain to the other.



Algorithms to compute the edition distance: Fisher-Wagner

1. **What is a morphological skeleton? L3 s18**



S1(X)

S0(X)

Sr(X) = X eroded by rB – (X eroded by rB then opened by B)

B very small structuring element

Object is encoded in the skeleton and can be reconstructed using the same B.

This is a flexible descriptor in the sense that we can ignore the part of the skeleton with small values of r in order to consider the largest part of the object. Doing so for a human shape for example, we can ignore the position of the arms and focus on the body.

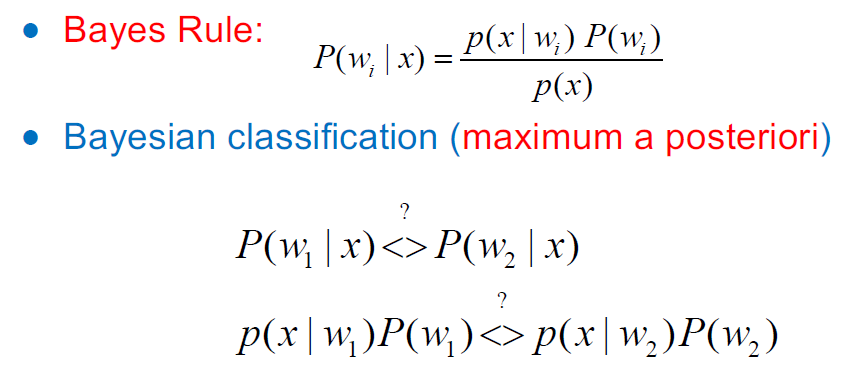
Multi-scale representation of the object, useful to compare object.

1. **What is a Bayesian classifier? (principle, advantages & limitations, application to Gaussian cases) L4 s7**

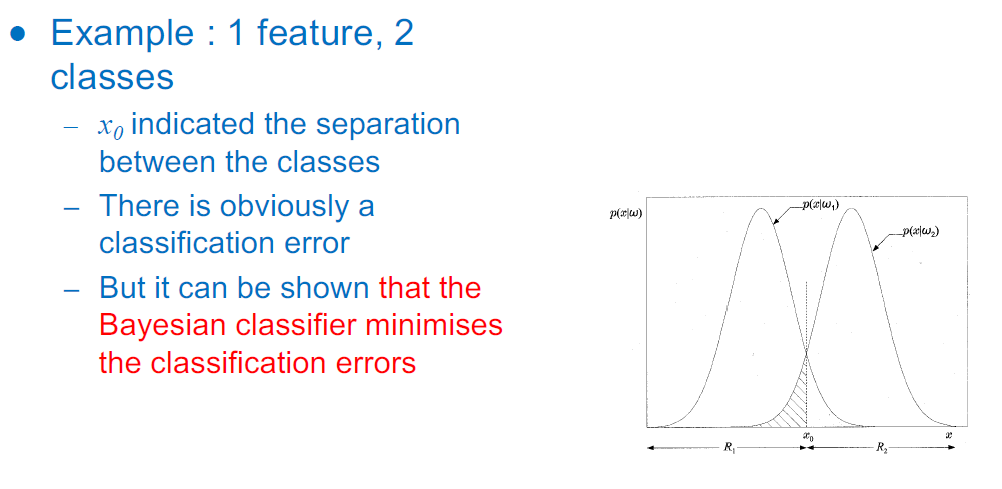
Supervised learning

Datapoint x, classes w1, w2, w3, …:

We want to approximate P(wi|x) to answer the question: To which class the sample x is most likely to belong to?



Because p(x) doesn’t depend on wi 🡪 we can ignore it



Limitation: Distribution assumed to be known!! Can be hard if not Gaussian.

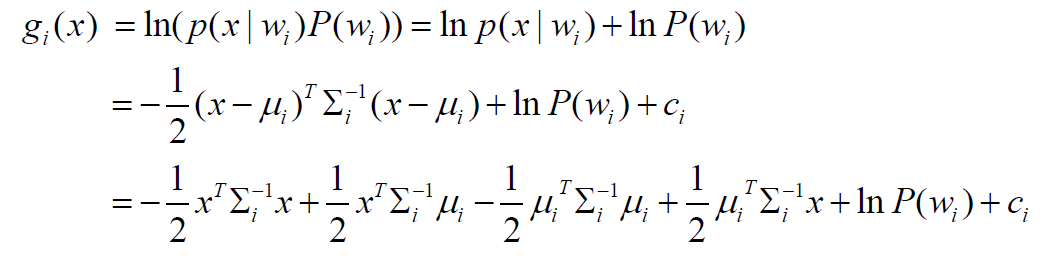
Generalizable to n classes :

Decision surface between class i and j: P(wi|x) – P(wj|x) = 0

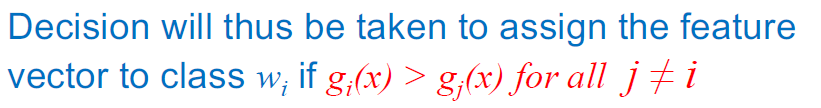
> 0 🡪 more likely to be in class i

< 0 🡪 more likely to be in class j

Gaussian case: if we assume that p(x|wi) are Gaussian, we can take the logarithm

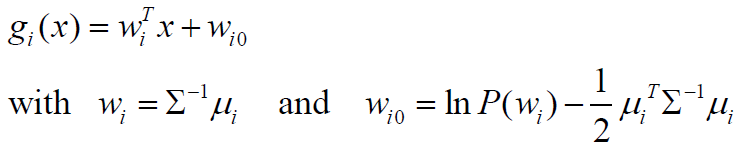
 as discriminant function :

(monotonally increasing)



 Special case:

* Common covariance matrix for all classes

Discriminant becomes

Decision surfaces are

hyperplanes

Part 2 – short answers

1. **What is a Median filter? L1 s29**

For each pixel:

1. Consider nxn neighborhood
2. Sort the pixels value in ascending order
3. Set the median value (middle of the list) to that pixel

Advantages: Supresses small variations and keeps the contours

Very effective for “salt and pepper” kind of noise

1. **What is the Laplacian of Gaussian (LoG) method for edge detection? (this question cannot be taken if question 4 is taken in Part 1) L2 s23**

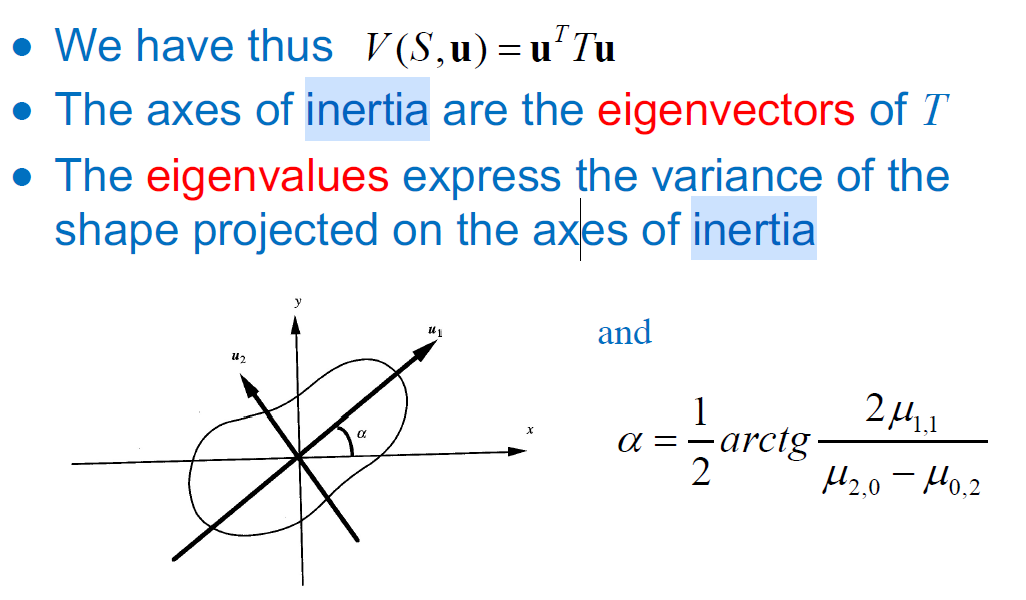
See Q4 in part 1

1. **How do we calculate the axes of inertia of a binary object? L3 s44**

Une image contenant texte

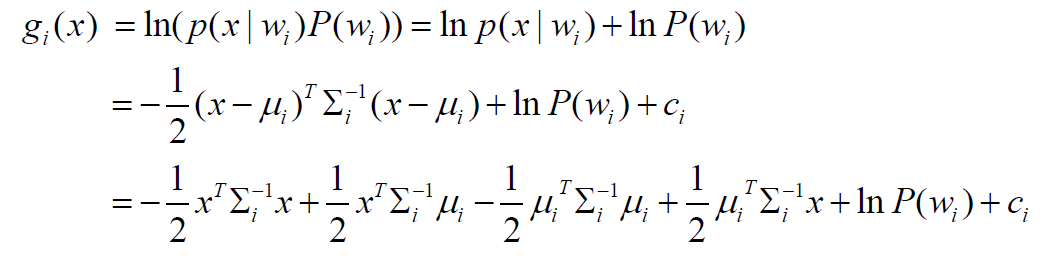
Description générée automatiquementWe are looking for a system of axes that minimizes the variance of the shape projected on those axes.

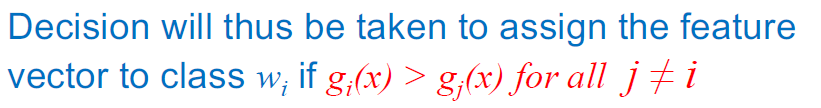
Variance:



1. **What is a Euclidean distance classifier? L4 s18**
2. **What is a Mahalanobis distance classifier? L4 s18**

In a Gaussian case of a Bayesian classifier, the discriminant is





Une image contenant texte

Description générée automatiquement

1. **What is a k-NN classifier? L4 s19**

k-NN is a supervised method that doesn’t make any assumption about the distribution of the data. Very simple algorithm: for a new datapoint, we assign the label of the most represented class around it. In the training set, we take the k-nearest-neighbors of this new point and assign it the dominant label. Various distances can be used (Euclidean, Manhattan, Cosine).

Sensitive to choice of k: too low 🡪 underfitting, too high 🡪 overfitting

Probability of error bounded below by Bayesian classifier

1. **What is a linear perceptron and how can we train it? L4 s22**

In the case of two classes: we want to find the hyperplane that best separates the two classes.

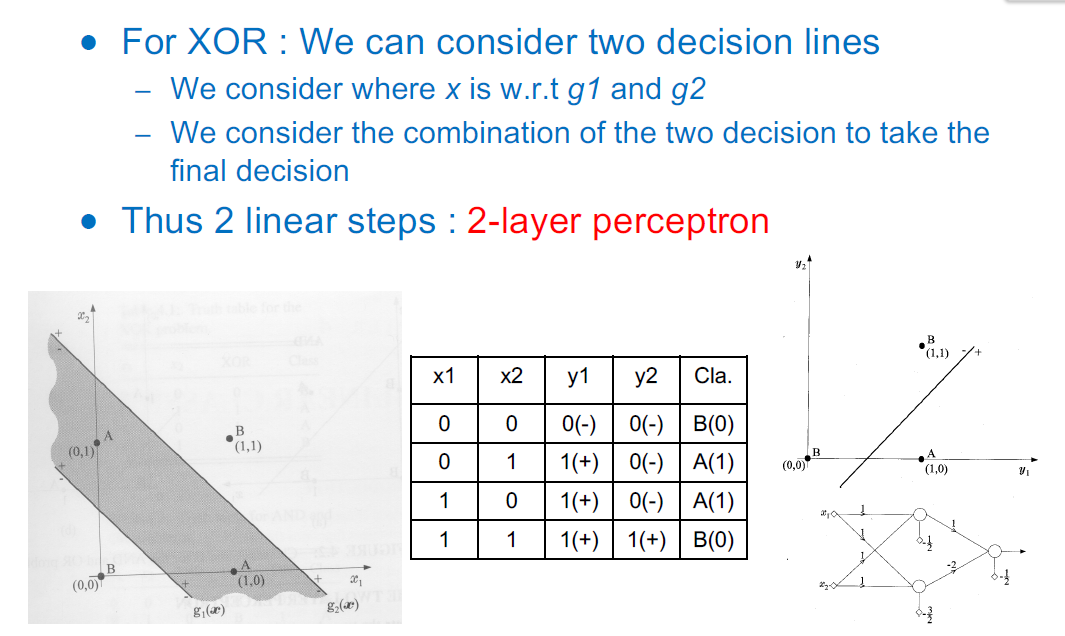
Une image contenant texte

Description générée automatiquementWe define some weights w of dimension of x + 1 and we add a 1 to each feature vector to allow for bias. The goal:

Une image contenant texte

Description générée automatiquement

1. **What is a multi-layer perceptron? L4 s24**

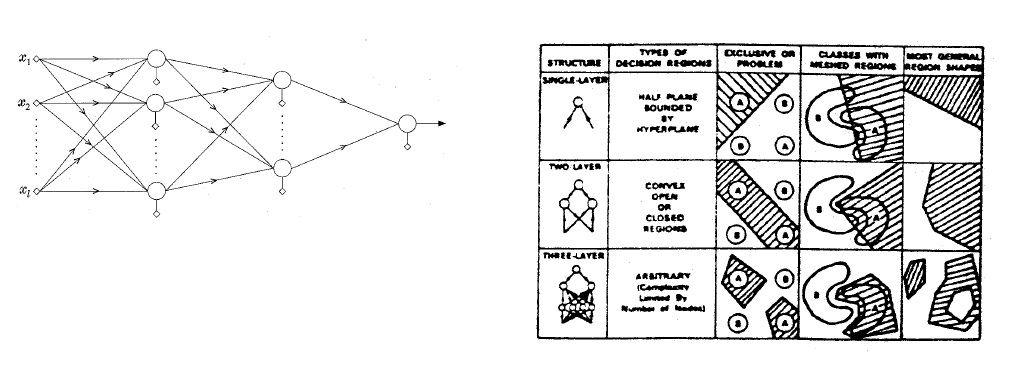


Separation no possible with 1 decision boundary 🡪 Solution: add layers!!

Gives more degrees of freedom to the decision line.

Same as a single linear perceptron but with multiple decision hyperplanes.

Optimization using same idea as for single perceptron.



1. **What is supervised and non-supervised classification?**

Supervised: We have a training set of feature vectors with corresponding labels

Example: Bayesian classifier, linear classifier, MLP, NN…

Unsupervised: We only have features without labels. We need to find similarities among vectors to build clusters

Example: clustering algorithms (k-means)

1. **What is non-supervised classification and describe the k-means algorithm?**

Unsupervised classification: unlabeled feature vectors. We want to classify them into different clusters.

k-means:

1. Chose m centroids randomly
2. Until convergence do:

* Each vector assigned to the closest centroid
* Recalculate the position of the centroids as the means of the vectors of each class