Introduction to Pattern Recognition

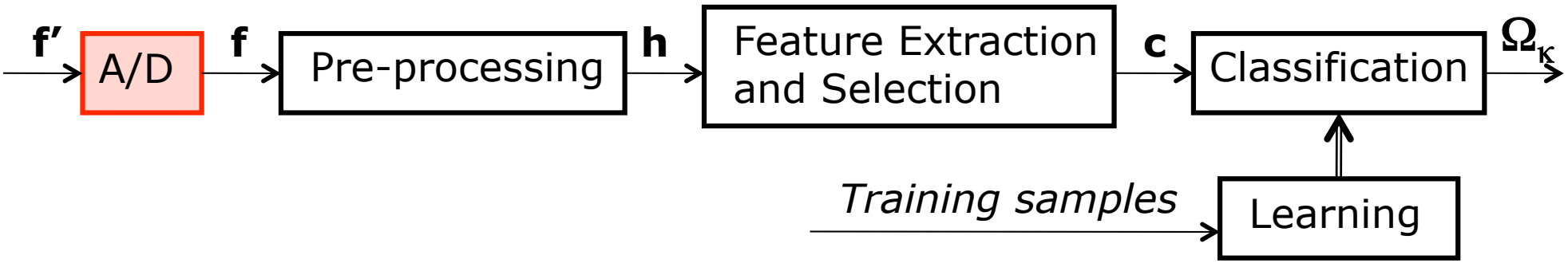
A Review

Dr. Elli Angelopoulou

Lehrstuhl für Mustererkennung (Informatik 5)

Friedrich-Alexander-Universität Erlangen-Nürnberg

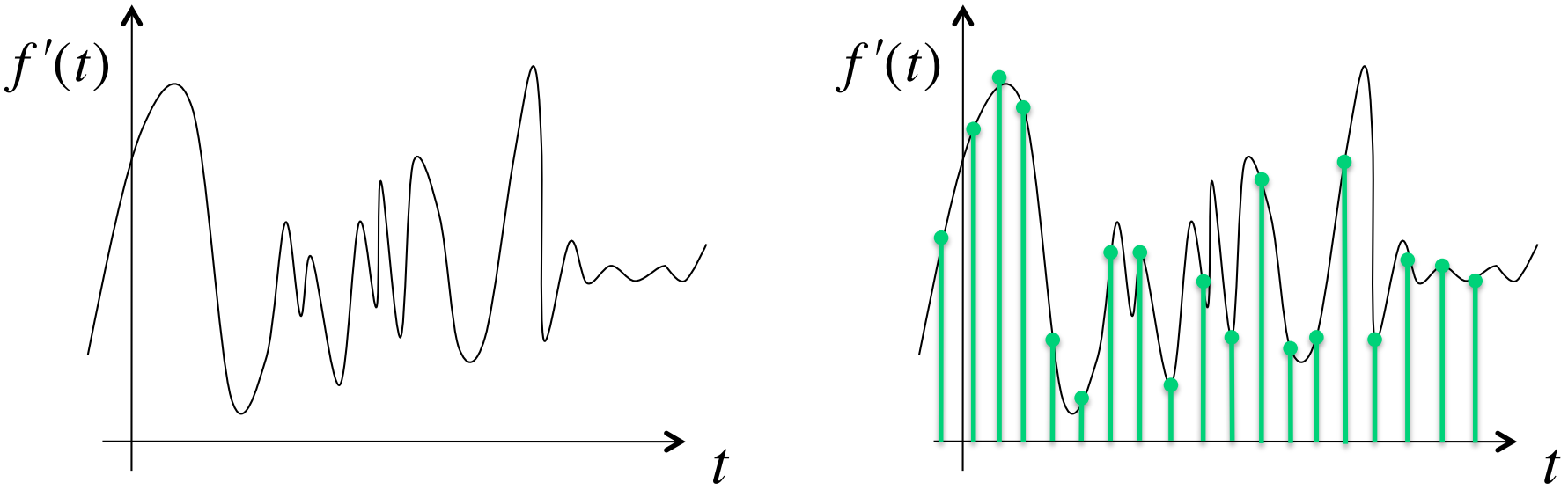
Pattern Recognition Pipeline – Step 1



Analog to Digital Conversion

- The goal of analog to digital conversion is to gather sensed data f' and change it to a representation that is amenable to further digital processing.
- There are two important aspects in the A/D conversion that can impact the PR pipeline:
  - Sampling
  - Quantization
- The overall goal of A/D conversion is to minimize information loss as the signal gets converted from a continuous to a discrete representation.

Sampling



- Sampling is the process of obtaining measurements of the sensed signal at finite positions in time or space.

Nyquist Sampling Theorem

- This theorem provides a theoretical sampling rate at which we will incur no information loss.
- Let $f(x)$ be a band-limited function in the frequency range $(-B_x, B_x)$.
- Then $f(x)$ is completely determined by the samples

$$f_k = f(k\,\Delta x) \quad \text{where } k = 0, \pm 1, \pm 2, \ldots$$

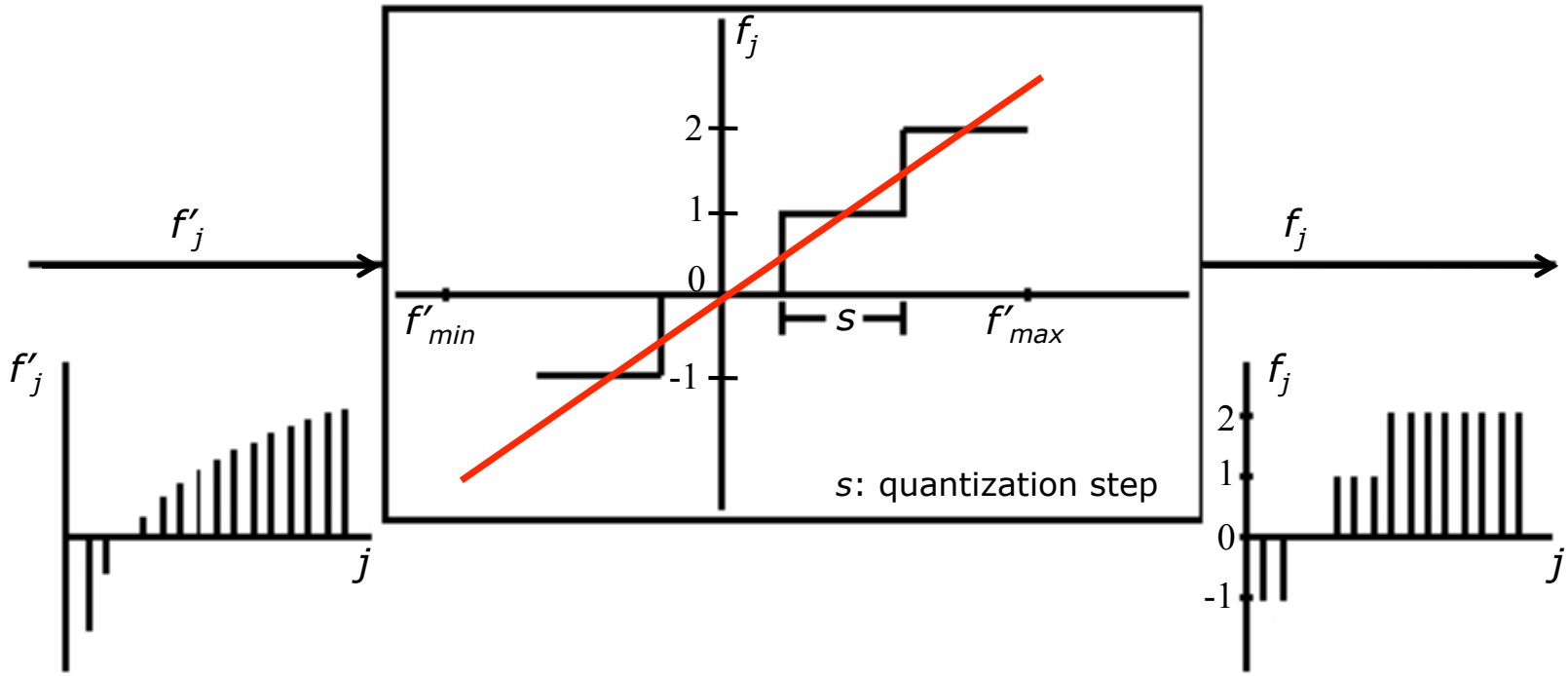
if the sampling interval is chosen as $\Delta x \le \frac{1}{2B_x} = \frac{\pi}{\omega_0}$

- The original signal $f(x)$ can be reconstructed without any error using the following interpolation

$$f(x) = \sum_{k=-\infty}^{\infty} f_k \frac{\sin(2\pi B_x (x - k\Delta x))}{2\pi B_x (x - k\Delta x)} = \sum_{k=-\infty}^{\infty} f_k \,\mathrm{sinc}(2\pi B_x (x - k\Delta x))$$

Quantization

- Once the signal is recorded at discrete locations, it must be stored using a finite number of bits.

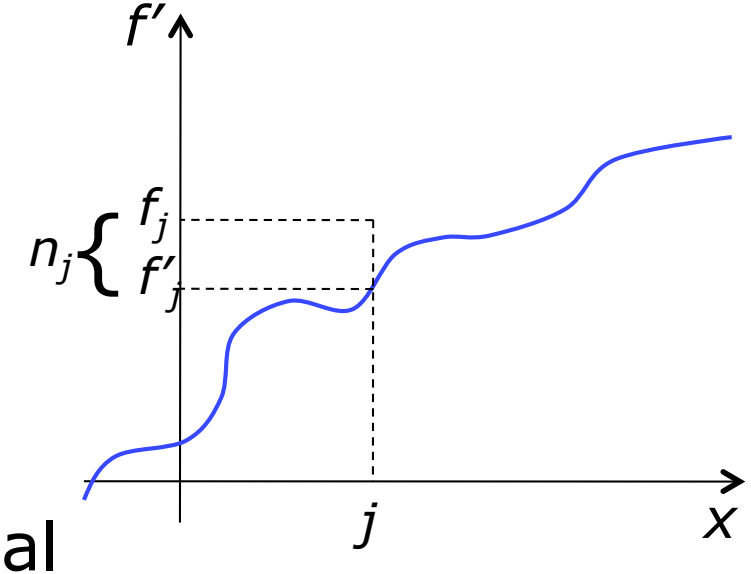


- The number of quantization steps is defined by the number of bits we use to represent the value of the function.

Quantization Error

- Two key questions:
  1. How many bits?
  2. How do we use these bits?
- When we use B bits, we get 2^B quantized levels.
- When we use discrete values to store continuous values we incur information loss, known as quantization error.
- Quantization Error: The error we make when we approximate a real value f'_j by a discrete value f_j:

$$n_j = f'_j - f_j$$



Signal-to-Noise Ratio (SNR)

- There exists a standardized way of expressing the noise in a system or sensor that is associated with quantization. It is called the *Signal-to-Noise Ratio*.
- SNR is a general measure that is used for different types (sources) of noise.
- In Pattern Recognition it is defined as: $SNR = \frac{E\{f'^2\}}{E\{n^2\}}$
- Because input signals can have a wide dynamic range, SNR is usually expressed in terms of the logarithmic decibel scale:

$$SNR_{dB} = r = 10\log_{10}\frac{E\{f'^2\}}{E\{n^2\}} = 10\log_{10}(r')$$

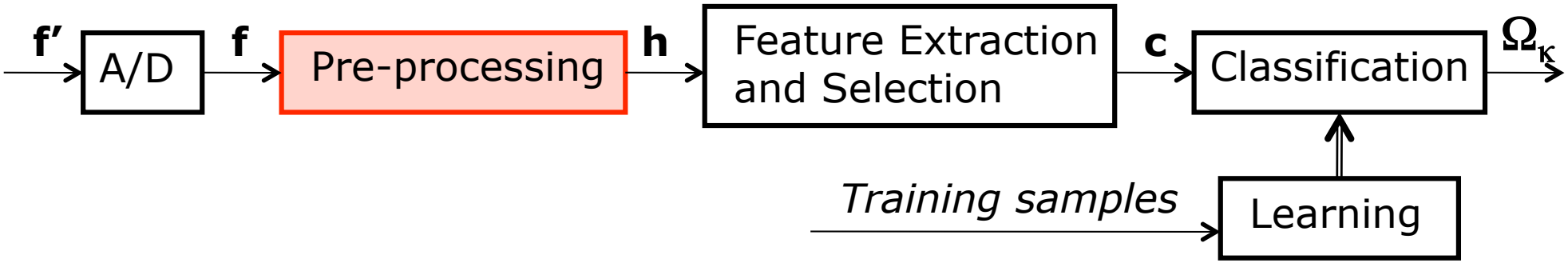
Conclusions on Quantization

- We showed that under certain assumptions, the SNR is directly proportional to the number of bits used for quantization:

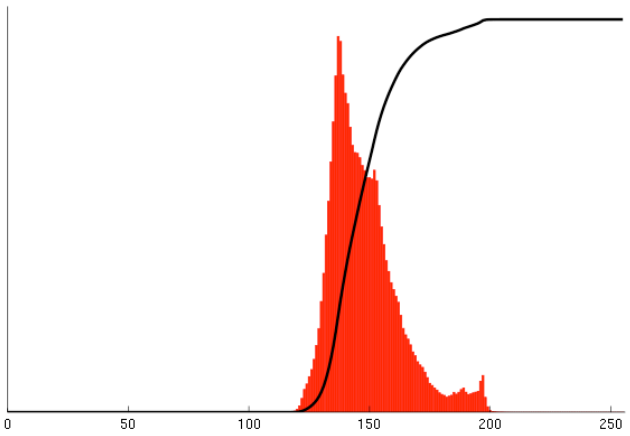
$$SNR_{db} = r = 6B - 7.2$$

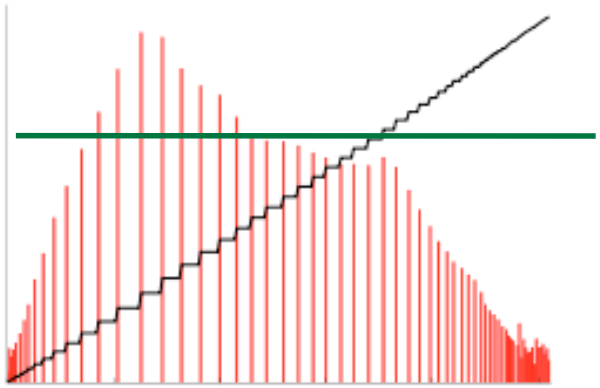
- We also showed that for linear quantization, we get the best results (minimal total quantization error) if the signal amplitudes are equally distributed.
- If the data is high-dimensional then the A/D conversion process involves vector quantization.
- A codebook should then be created (e.g. based on K-means). The signal is stored as an offset to the closest mean (codeword).

Pattern Recognition Pipeline – Step 2



Histogram Equalization





- A histogram plots for each gray level value the frequency with which that value occurs (shown in red)
- The goal of histogram equalization is to have an almost horizontal distribution of values.

Images courtesy of Phillip Capper, http://en.wikipedia.org/

Histogram Equalization Algorithm

1. Compute the histogram of a given image
2. Compute its cumulative distribution function.
3. Break the vertical axis of the cdf plot, into *n* equidistant blocks, where *n* is the number of gray values in the output image.
4. Then all the pixel values (in the horizontal x-axis) in the first block of the cdf get mapped to one gray value. All the pixel values in the 2nd block of the cdf get mapped to the next pixel value etc.
5. In the resulting image each of the *n* intensities has the same probability of occurring. The pixels are spread evenly across the entire range of these *n* pixel values. The image has the highest possible contrast.

Histogram Equalization - Clarifications

- The redistributed values in the tessellation of the vertical axis correspond to the histogram of the equalized image.
- A grey value f is mapped from the cumulative distribution function $D(f)$ to a new "equalized" grey value as follows:

$$f' = \mathrm{g}(f) = \mathrm{round}\left(\frac{\mathrm{D}(f) - D_{\min}}{\#\mathrm{pixels} - D_{\min}} (L-1) \right)$$

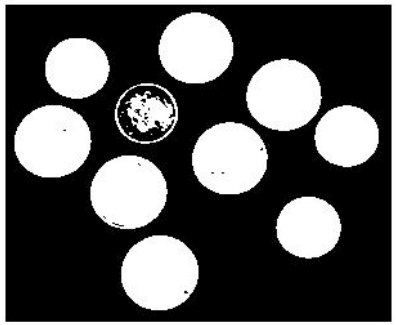
where $D_{\min}$ is the smallest non-zero value in the cdf and L is the number of levels in the new image.

Thresholding

- Thresholding transformation $T(f_{ij})$ for $L_2 = 2$:

$$T(f_{ij}) = \begin{cases} 0 & \text{if } f_{ij} \le \theta \\ 1 & \text{otherwise} \end{cases}$$

- We studied various methods for selecting θ.
  - Intersection of 2 Gaussians
  - Optimal binary thresholding
  - Otsu's thresholding criterion
  - A heuristic approach which is best suited for unimodal distributions.
  - Entropy-based binarization.



Filtering

- A wide range of transformations can be applied to to images in a form of a *filter*.

- Mathematically, a filter H can be treated as a function on an input image I:

$$H(I) = R$$

- There are two main categories of filtering transformations:
  - Linear shift-invariant transformations
  - Non-linear transformations
- Homomorphic mapping allows the transformation of non-linear domains to linear domains (e.g. log, FT, cepstrum).

Linear Shift Invariant (LSI) Transformations

- LSI transformations can be applied to signal using convolution.

$$R(x,y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} H(x-i, y-j) I(i,j)$$

Original image

- We studied a variety of widely used LSI transformations.

1. Smoothing or low-pass filtering.
   - Its goal is to remove noise
   - Mean filtering
   - Gaussian filtering

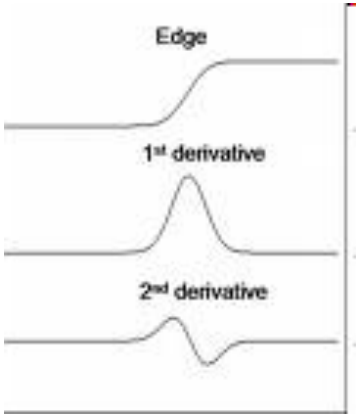
Mean filtering

Gaussian filtering

LSI Transformations - continued

2. Edge Detection or high pass filtering
   - Its goal is to detect pixels where a significant change in intensity occurs.
   - Gradient-based edge detection

$$\mathbf{G}(x,y) = \left(\frac{\partial I(x,y)}{\partial x}, \frac{\partial I(x,y)}{\partial y} \right) = (I_x(x,y), I_y(x,y))$$



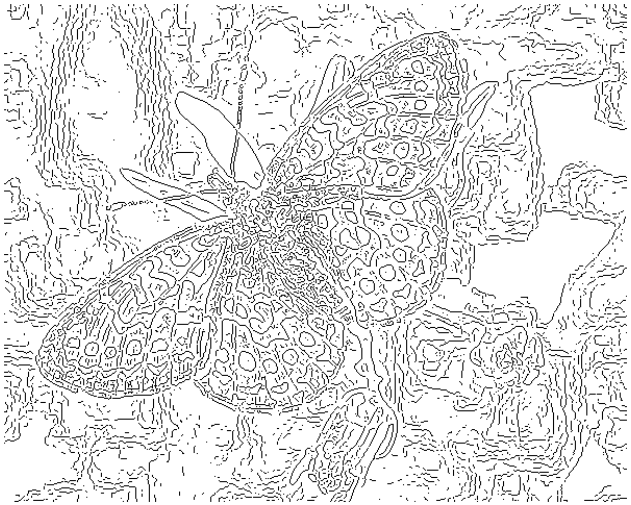
   - Laplacian-based edge detection

$$\nabla^2(I(x,y)) = \left(\frac{\partial^2 I(x,y)}{\partial x^2} + \frac{\partial^2 I(x,y)}{\partial y^2} \right)$$

- Smoothing almost always precedes edge detection
- Low-pass and high-pass filtering can be applied on the same signal at different scales for a multi-resolution analysis.

Different Scales

Original image



Fine scale, high threshold

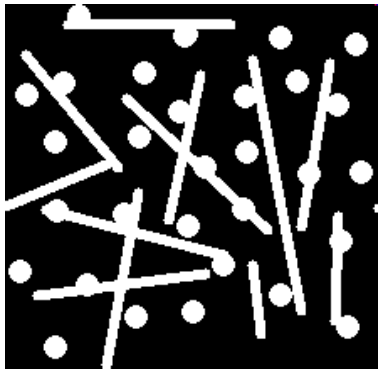
Coarse scale, low threshold

Coarse scale, high threshold

Non-Linear Transformations

- In non-linear transformations we focused on:

1. Mathematical Morphology
   - Images are treated as sets.
   - Different set operations can be defined for both binary and gray-scale images:
     - Erosion
     - Dilation
     - Opening
     - Closing
2. Rank Operations
   - Minimum (maps to erosion)
   - Maximum (maps to dilation)
   - Median



Original image

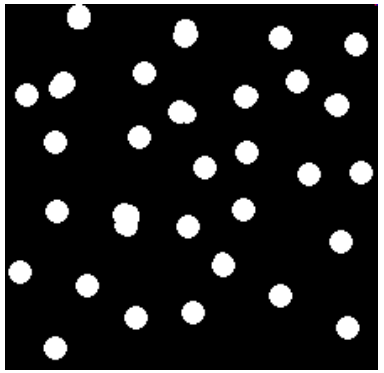
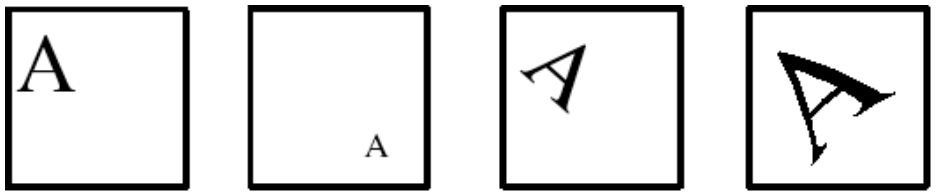


Image after opening

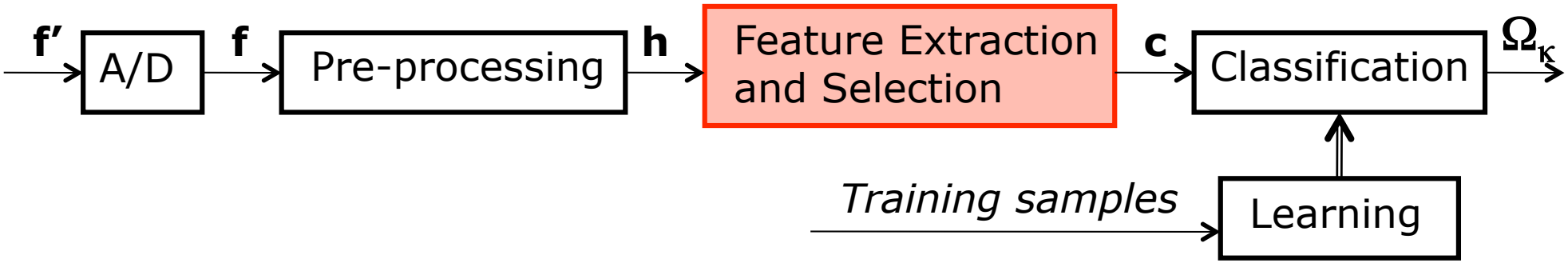
Pattern Normalization

- The goal of normalization is to map the signal to some **normalized representation** with respect to:
  - position -> moments (0^{th} and 1^{st} order)
  - size -> bounding box
  - pose -> moments (0^{th}, 1^{st} and 2^{nd} order)
  - energy level
  - duration
  - Illumination
- Geometric moments are very often used as part of the normalization process:

$$m_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q f(x,y)\,dx\,dy$$



Pattern Recognition Pipeline – Step 3



Feature Extraction

- In feature extraction we compute a numerical characteristic vector $\vec{c} \in R^N$, on which the subsequent classification task is performed.
- There are two distinct methods for extracting features:

1. Heuristic methods

   Typically involve a change in representation via methods like projection to new orthogonal bases.

2. Analytic methods

   The feature vector is derived as part of the solution to an explicit optimization problem.

Heuristic Methods

- The heuristic feature extraction methods that we studied:

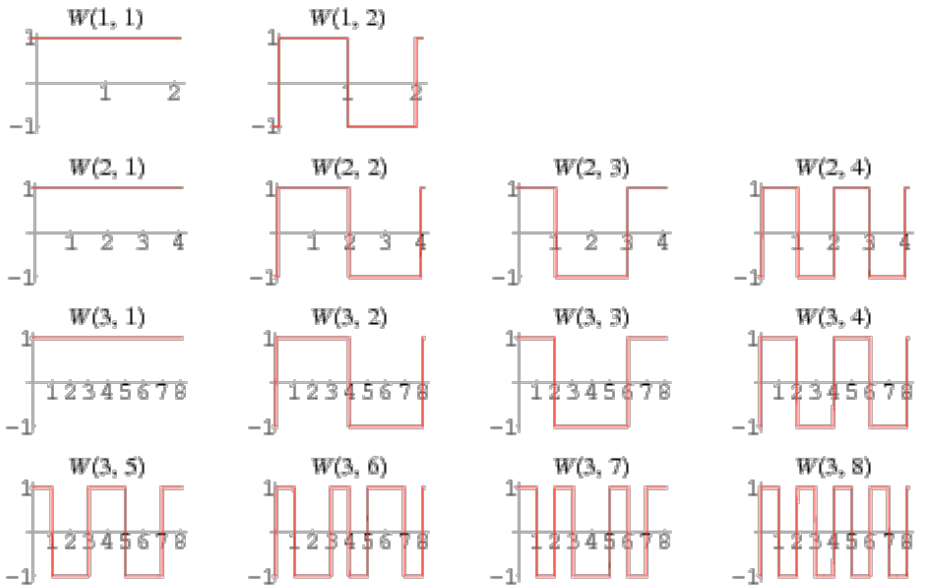
1. Projection to orthogonal bases
   - Fourier Transform
   - Walsh/Hadamard Transform
   - Haar Transform

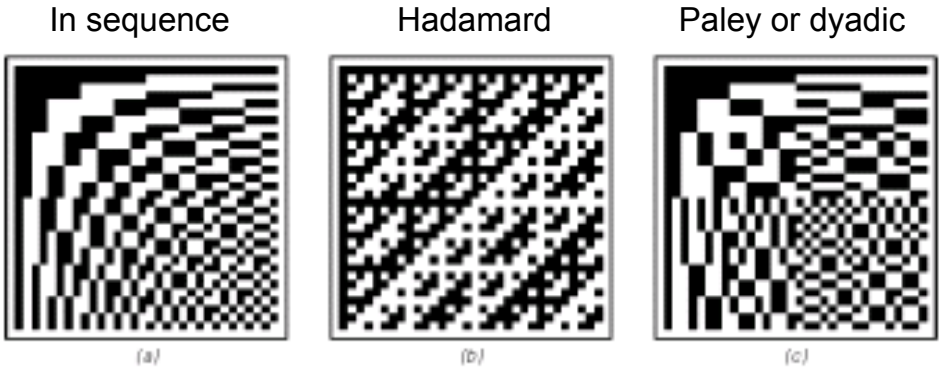
$$\vec{c} = \Phi^T \vec{f}$$

2. Spectrogram
3. Linear Predictive Coding
4. Geometric Moments
5. Feature Extraction via Filtering
6. Wavelets

Walsh-Hadamard Transform

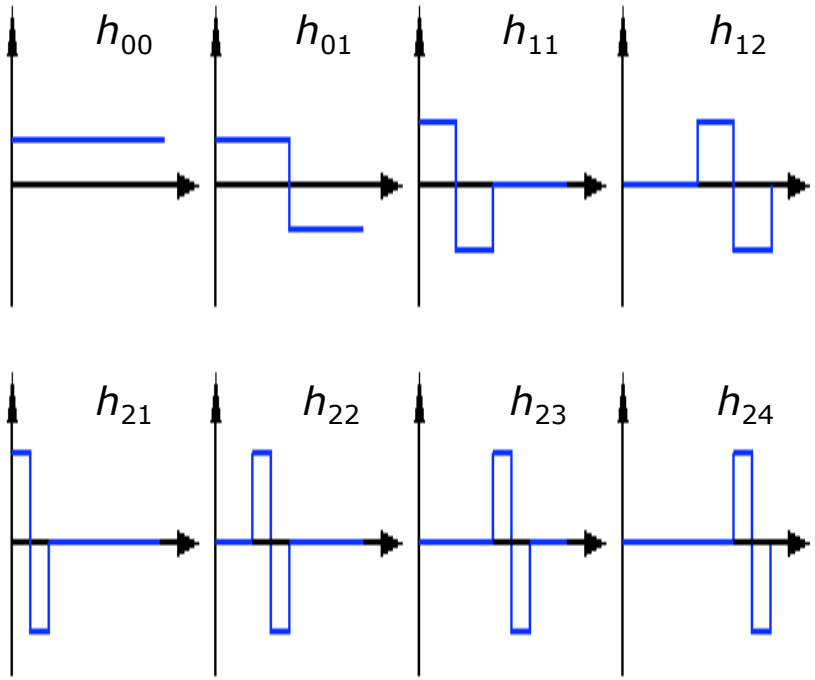
- The Walsh Functions are used as an orthogonal basis.
- There are many different ways of ordering the Walsh functions into a basis.
- One way of arranging the Walsh functions is via the Hadamard matrix.





Haar Transform

- Haar functions: A more intuitive set of orthogonal "square-wave" functions that can be used as a new feature space.
- It is a 2-parameter recursive function, where *p* specifies the magnitude and width of the shape and *q* specifies its position



$$h_{pq}(x) = \frac{1}{\sqrt{M}} \begin{cases} 2^{p/2} & \text{for } \frac{q-1}{2^p} \le x < \frac{q-0.5}{2^p} \\ -2^{p/2} & \text{for } \frac{q-0.5}{2^p} \le x < \frac{q}{2^p} \\ 0 & \text{for other values of } x \text{ in } [0,1] \end{cases}$$

Linear Predictive Coding

- This representation is widely used in sound/speech processing.
- It assumes a buzzer-tube model.
- The glottis (the space between the vocal cords) produces the buzz, which is characterized by its *intensity* (loudness) and *frequency* (pitch).
- The pharynx forms the tube, which is characterized by its *resonances*, which are called *formants*.
- Key idea: The present sample f_n of the speech is predicted by the past *m* speech samples so that:

$$\hat{f}_n = a_1 f_{n-1} + a_2 f_{n-2} + \cdots + a_m f_{n-m} = \sum_{\mu=1}^{m} a_\mu f_{n-\mu}$$

Wavelet

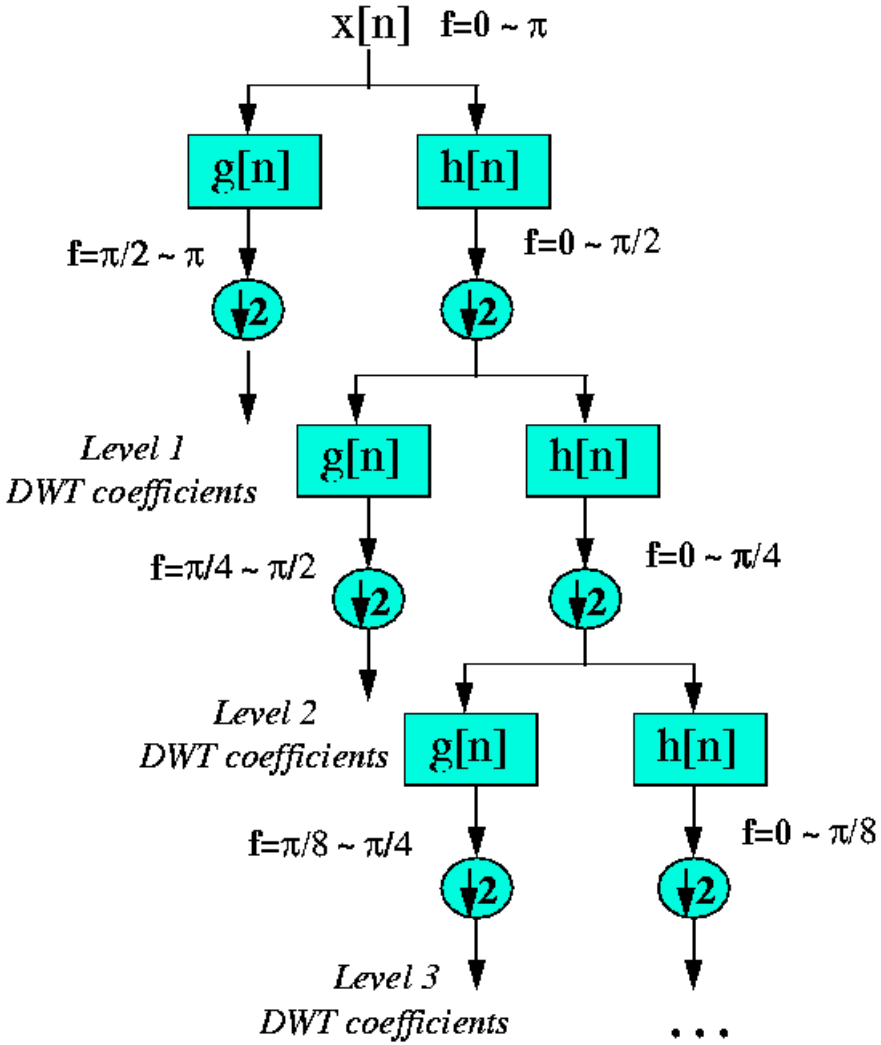
- Used for multi-resolution analysis.
- It uses a sliding scalable window.
- The building block of the wavelet transform, its window, is a small wave, a *wavelet*, which is given by a function $\psi(t)$.

$$\mathrm{CWT}_f^{\psi}(\tau,\alpha) = \frac{1}{\sqrt{|\alpha|}} \int_{-\infty}^{\infty} f(t)\psi^*\left(\frac{t-\tau}{\alpha}\right)dt$$

- A *wavelet transform* is the representation of a signal f(t) by wavelets.

$$f(t) = \frac{1}{c_\psi} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{CWT}_f^{\psi}(\tau,\alpha) \frac{1}{\sqrt{|a|}} \psi^*\left(\frac{t-\tau}{\alpha}\right)dt\,\frac{d\alpha}{\alpha^2} \qquad c_\psi = \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{|\omega|} d\omega$$

Discrete Wavelet Transform

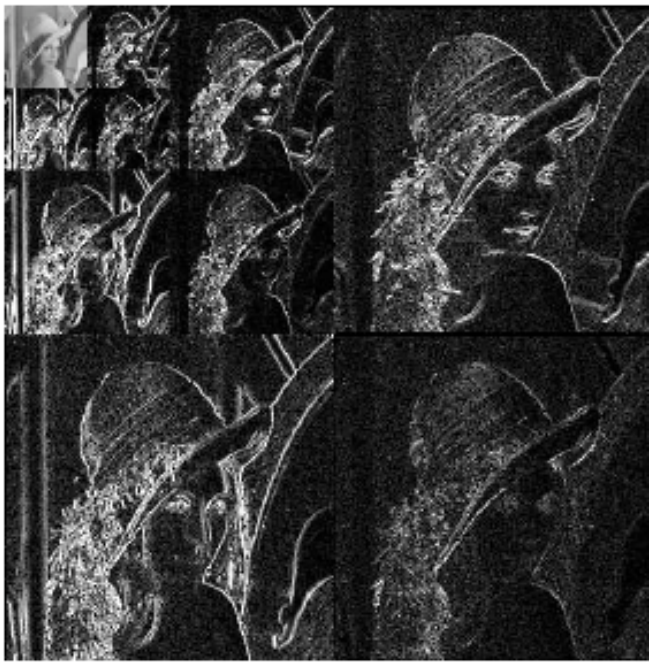


where g[n] is a half-band highpass filter, h[n] is a half-band lowpass filter and x[n] is the input signal with frequency between 0 and π.

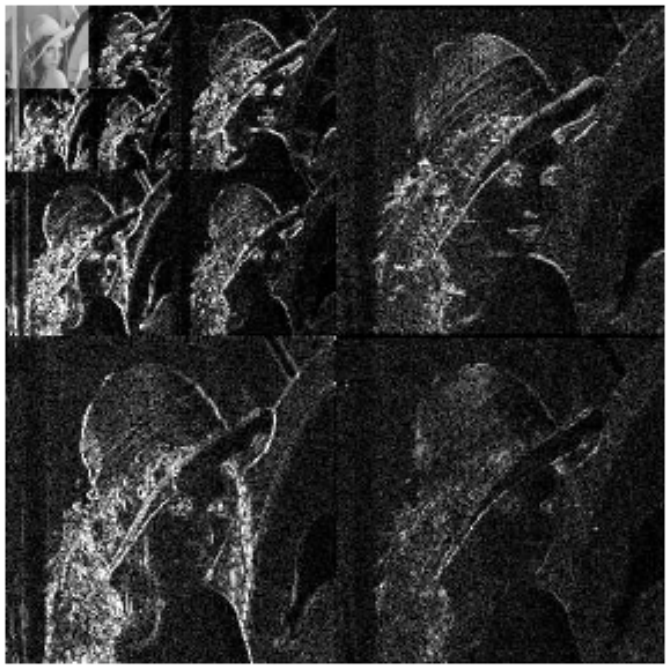
Image courtesy of Robi Polikar http://users.rowan.edu/~polikar/WAVELETS/WTpart4.html

Different Types of Wavelets

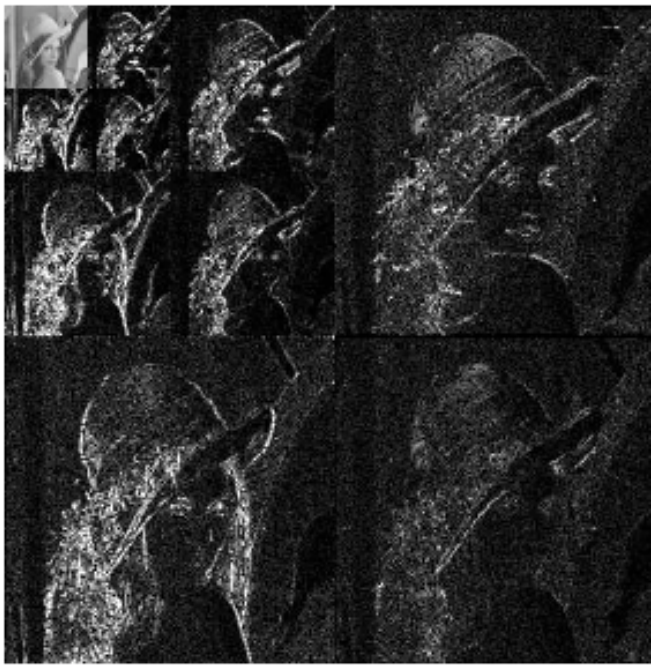
Original image



Haar wavelets



Daubechies wavelets



Biorthogonal wavelets

Analytic Feature Extraction

- Idea: Compute a matrix Φ, so that the resulting features $\vec{c}$ optimize a quality criterion.

$$\vec{c} = \Phi\vec{f}$$

- Depending on the optimization criterion we have different analytic feature extraction methods:
  - PCA - maximizes the spread of features
    Eigenfaces
  - Minimize intraclass distance
  - Maximize interclass distance
  - LDA - mimimize intraclass and maximize interclass distance
    Fisherfaces
  - Optimal feature transform – minimize misclassification rate.

Feature Selection

- Due to the curse of dimensionality we want to select a subset of features from our feature vector that best preserve the discriminating power of the feature vector.

$$\vec{f} \in R^N$$

$$\vec{c} \in R^M \quad \text{, where } M < N$$

$$\vec{c}' \in R^{M'} \quad \text{, where } M' < M$$

$$\{c'_1, c'_2, \ldots c'_{M'}\} \subset \{c_1, c_2, \ldots c_M\}$$

Feature Selection Algorithms

- Algorithms for feature selection are characterized by:

1. The objective function (a.k.a. criterion function) used in evaluating the "goodness" of a subset.
2. The optimization method used in searching the space of possible subsets for the best subset.

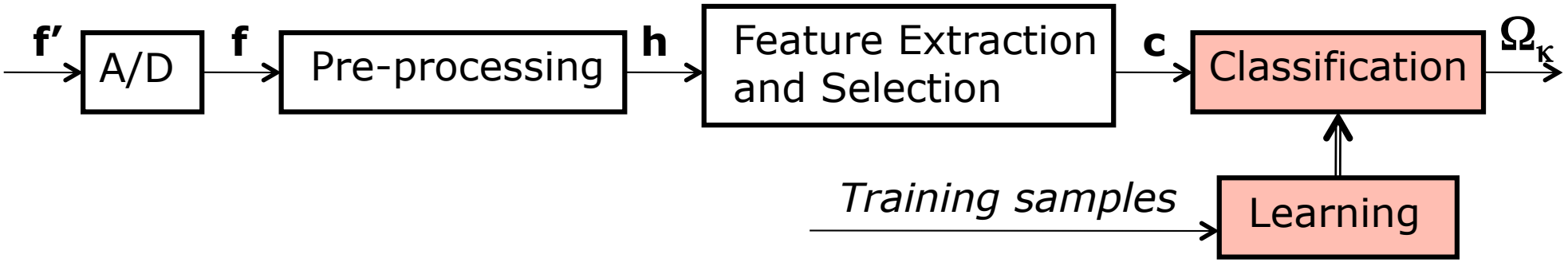
- Widely used criterion functions for feature selection:
  1. Error-rate (minimize)
  2. Bayesian distance (maximize)
  3. Conditional entropy (minimize)
  4. Mutual information (maximize)

Search Strategies for Feature Selection

1. Random Selection.
2. Exhaustive Search.
3. Greedy
4. Hardest Pair
5. (l,r)-search.
6. Branch and Bound.

   Key assumption: Monotonic objective function

Pattern Recognition Pipeline – Step 4



Classification

- Goal of a classifier: Map the computed feature vector $\vec{c}$ to a class Ω_κ.

$$\vec{c} \xrightarrow{\delta\left(\Omega_\kappa \middle| \vec{c}\right)} \Omega_\kappa$$

- The classification task can be viewed as a decision function $\delta()$:

$$\delta\left(\Omega_\kappa \middle| \vec{c}\right) = \begin{cases} 1 & \text{for } \Omega_\kappa, \text{ if it is decided that } \vec{c} \in \Omega_\kappa \\ 0 & \text{for all other classes} \end{cases}$$

- In some classifier the mapping to a class is determined via a discriminant function:

$$d_\kappa\left(\vec{c}\right) = \begin{cases} 1 & \text{if } \vec{c} \in \Omega_\kappa \\ 0 & \text{otherwise} \end{cases}$$

Different Classifiers

- Classification
  - Statistical classifiers
    - Bayesian classifier
    - Gaussian classifier
  - Polynomial classifiers
  - Non-Parametric classifiers
    - k-Nearest-Neighbor density estimation
    - Parzen windows
    - Artificial neural networks
      - Radial basis function networks
      - Multilayer perceptron

Statistical Classifiers

- Bayes classifier

$$\delta(\Omega_\lambda|\vec{c}) = \begin{cases} 1 & \text{if } \lambda = \arg\max_\kappa p(\Omega_\kappa|\vec{c}) \\ 0 & \text{otherwise} \end{cases}$$

A Bayes classifier with a (0,1)-cost function is an optimal classifier.

- Gaussian classifier

It is a Bayesian classifier where we have normally distributed class-conditional feature vectors $p(\vec{c}|\Omega_\kappa)$.

$$p(\vec{c}|\Omega_\kappa) \approx \mathcal{N}(\vec{c}, \vec{\mu}_\kappa, \Sigma_\kappa)$$

Polynomial Classifiers

- The classification decision is based on *K* parametric discriminant functions:

$$d_\kappa(\vec{c}) \in d(\vec{c},\vec{a}_\kappa)$$

- Deriving the discriminant functions is equivalent to deriving the coefficients $\vec{a}_\kappa$.
- Given a labelled training set the coefficients can be derived by solving a system of linear equations.
- Beware of overfitting!!

Non-Parametric Density Estimation

- When we have no information about the model of the underlying probability density function, we can approximate it via non-parametric methods.
- A common framework is the use of relative frequencies:

$$p(\vec{c}) = \frac{K}{NV}$$

- Option 1: Use a fixed value for *K* and find the corresponding V from the data

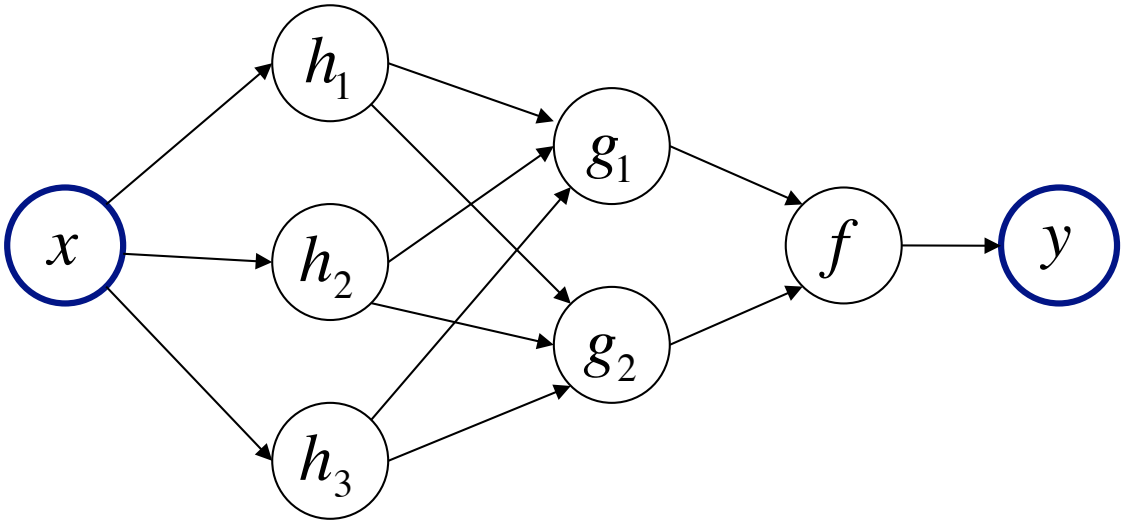
=> **K-nearest-neighbor** (fix *K*, look for a *V*)

- Option 2: Use a fixed volume *V* and find the corresponding value of K from the data

=> **kernel-based density estimation** (fix V, look for a K)

Artificial Neural Networks

- In general an ANN operates as a function $f : x \rightarrow y$.
- There can be multiple layers, some of which may be hidden.
- A widely used form of composition is: $f(x) = \phi\left(\sum_i w_i g_i(x)\right)$
- ϕ is often referred to as an activation function.



Two Types of ANN

- Radial Basis Function Networks

  Each neuron computes a RBF.

- Multilayer Perceptron

  Each node performs a thresholding operation via a sigmoid function.