Institut
Mines-Telecom

Introduction to compression and quantization

Marco Cagnazzo,
cagnazzo@telecom-paristech.fr

TSIA 207

Plan

Plan

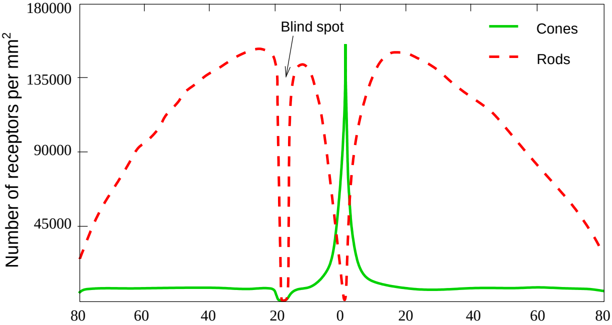
Introduction
- Vision
- Representation

Uniform Quantization

Predictive Quantization

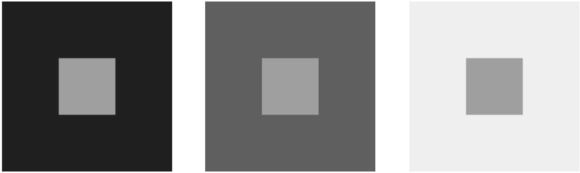
The eye

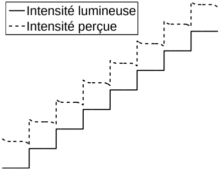
- Light is transformed in neural impulsions by the rods and the cones in the retina
  - **Cones** (6÷7 millions, at the center of the retina): sensitive to colors, good resolution, work in high illumination
  - **Rods** (75÷150 millions) : sensitive to light intensisty, smaller ersolution, work in low illumination



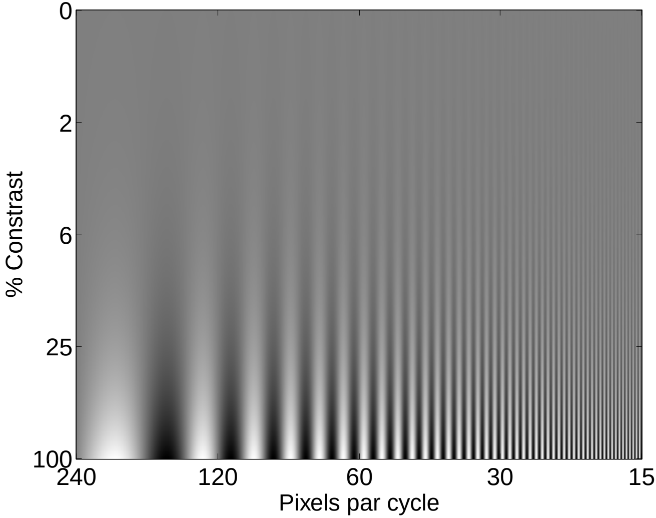
Light Perception

- Perceived luminosity: log function of intensity
- Global dynamic range: $\approx 10^{10}$ (100dB)
- Adaptation to light conditions





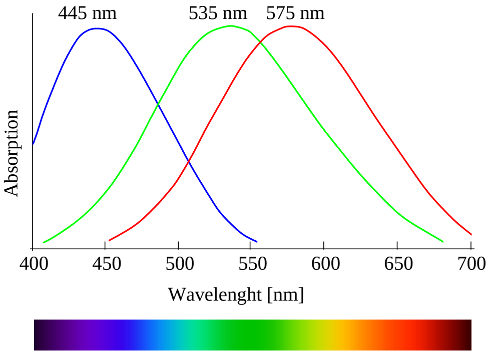
Contrast sensitivity function



- Contrast sensitivity function (CSF)
- Maximum at 2-5 cycles per degree

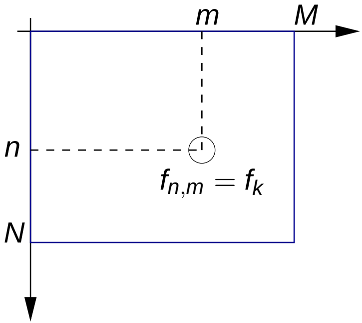
Color perception

- Visible light: 400÷700 nm
- Cones sensitivity depends on the wavelenght
  - 65% red cones
  - 33% green cones
  - 2% blue cones
- The color sensation corresponds to the *tristimulus*
- Combination of *primary colors*



Representation of digital images

- Discrete grid, image $N \times M$ pixels
- Pixel (m, n) is also accessible in "raster scan" in position k
  $k = (n - 1)M + m$
- Notaiton: $f_{n,m}$ or f_k



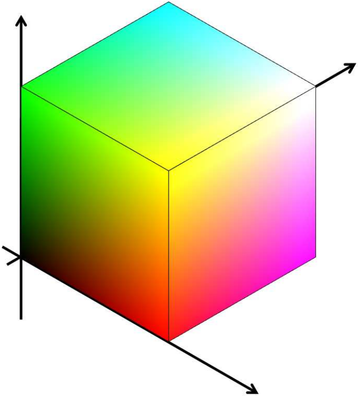
Representation of digital images

Images color : RGB Format

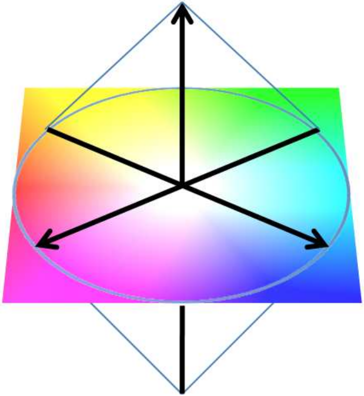
Color images = three components

Representation of digital images

Color spaces



Espace RGB



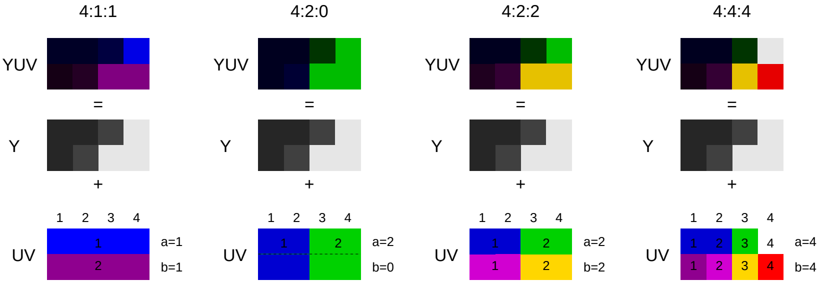
Espace HSV

Representation of digital images

Color images: YUV Format

One luminance component (Y) and two chromainance components (U and V, typically subsampled)

Color space subsampling



4:1:1	4:2:0	4:2:2	4:4:4
1/4 horizontal resolution Full vertical resolution	1/2 horizontal resolution 1/2 vertical resolution	1/2 horizontal resolution Full vertical resolution	Full horizontal resolution Full vertical resolution

Compression: Motivations

- HD DVB System
  1 luminance component 1920×1080
  2 chrominance components 960×540
  8 bits quantization
  25 images per second
  $R \approx 622$ Mbps
- 2-hours movie ≈ 560 GB

Compression fundamentals

Why is it possible to compress?

- Statistical redundancy
  - images are spatially homogeneous
  - successive images are similar one to another
- Psychovisual redundancy
  - Spatial frequency sensitivity
  - Masking effects
  - Contours importance
  - Other limits of the HVS
- A compression algorithm should take into account both kinds of redundancy ot maximize its performance

Lossless and lossy algorithms

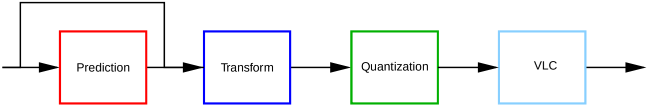
- Lossless algorithms
  - Perfect reconstruction
  - Based on statistics
  - Small compression ratio
- Lossy algorithms
  - Decoded $\neq$ original
  - Based on quantization
  - Psychovisual redundancy: “visually lossless”
  - High compression ratio

Symmetric and asymmetric algorithms (video)

- Symmetric algorithms
  - Same complexity for encoder and decoder
  - No motion estimation/compensation
  - Low compression ratio
  - Possibly real-time
- Asymmetric algorithms
  - Encoder (much) more complex than decoder
  - Motion Estimation/Compensation
  - High compression ratio
  - Typically “off line”, or hardware implementations

Basic tools for compression

- Transform
  - It concentrates information in a few coefficients
- Prediction
  - Alternative (and sometimes additional) method for information concentration
- Quantization
  - Rate reduction: rough representation of less important coefficients
- Lossless coding, or variable length coding (VLC)
  - Residual redundancy reduction



Plan

Definitions

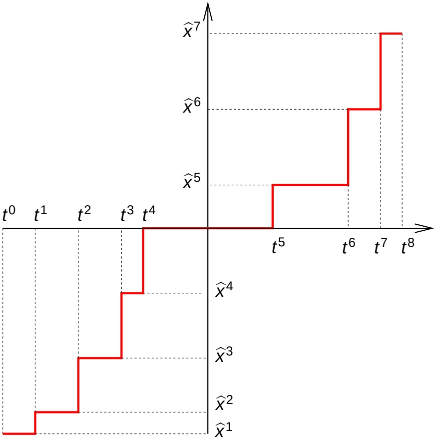
$$Q : x \in \mathbb{R} \rightarrow y \in \mathcal{C} = \{\widehat{x}^1, \widehat{x}^2, \dots \widehat{x}^L\} \subset \mathbb{R}$$

- C : Dictionary, it is a discrete subset of $\mathbb{R}$
- $\widehat{x}^i$: quantization level, codeword
- $e = x - Q(x)$: Quantization noise
- $\Theta^i = \left\{x : Q(x) = \widehat{x}^i\right\}$: Decision regions or cells

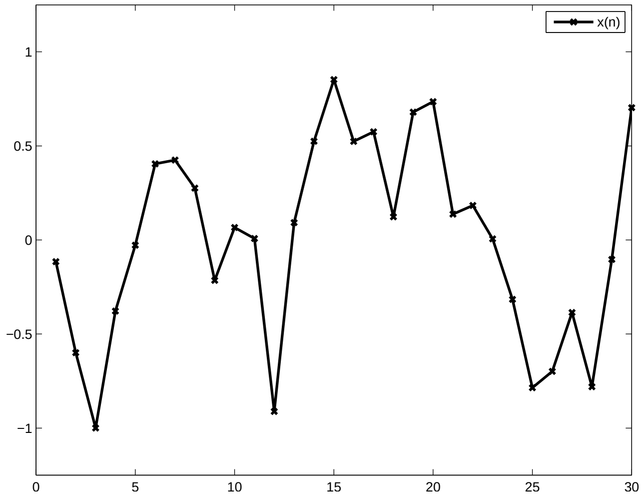
Regions and levels completely define the QS

Definition : scalar quantization (SQ)

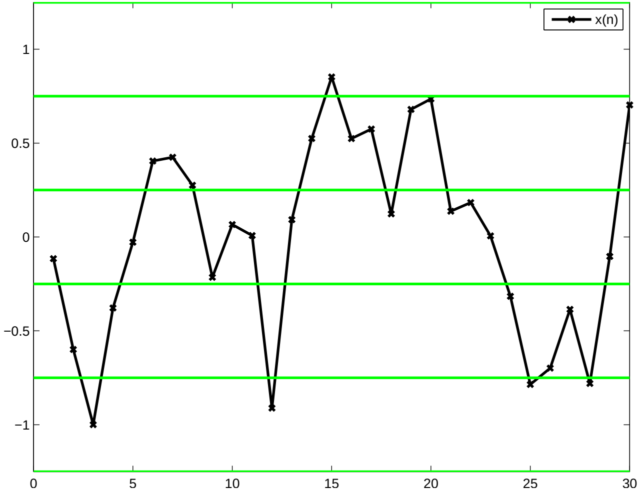
$$Q : x \in \mathbb{R} \rightarrow y \in \mathcal{C} = \{\widehat{x}^1, \widehat{x}^2, \ldots \widehat{x}^L\} \subset \mathbb{R}$$



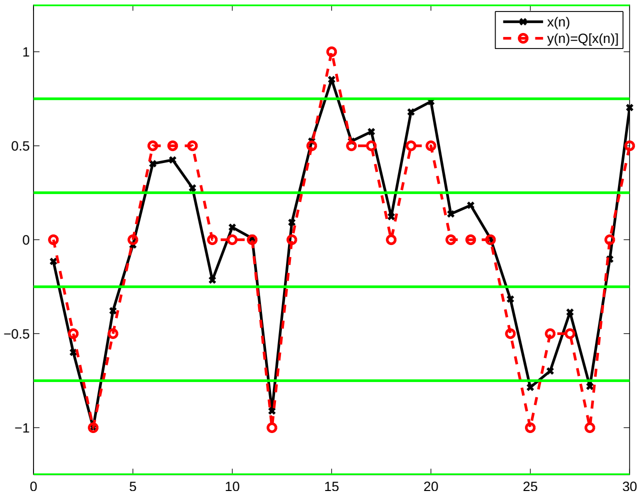
Example 1



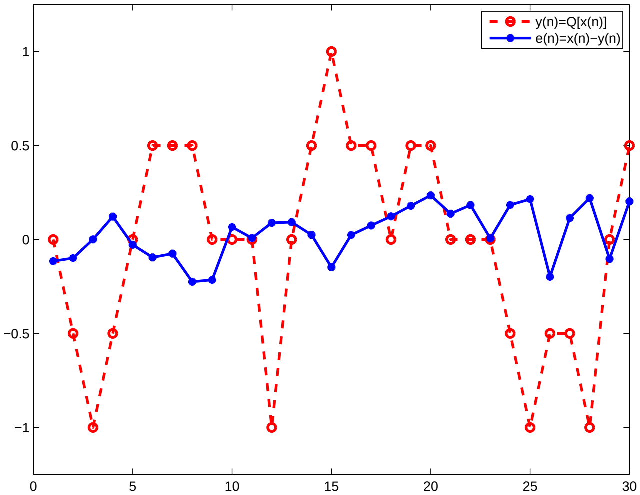
Example 1



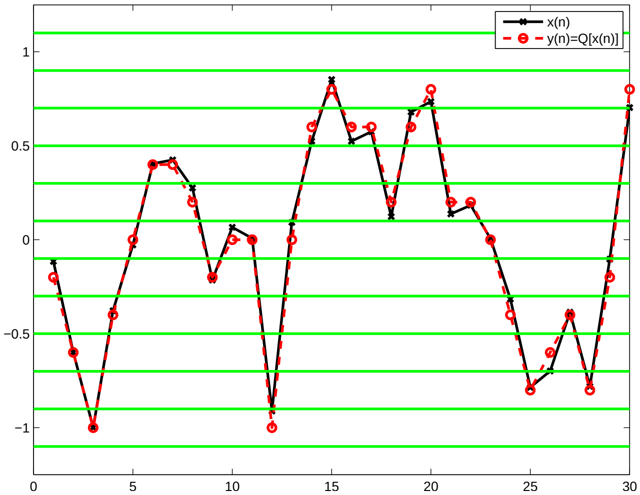
Example 1



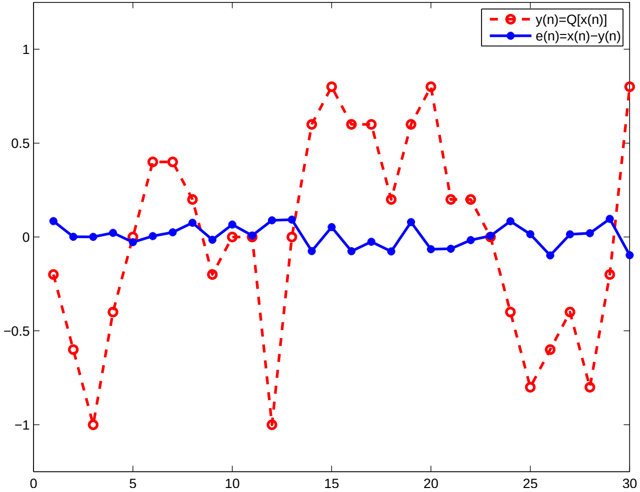
Example 1



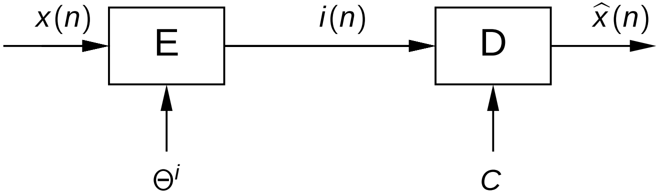
Example 2



Example 2



Quantization as coding / decoding



- The necoder sends $i(n)$
- The decoder associates to $i(n)$ a codeword $\widehat{x}^{i(n)}$
- Terminology: quantization is $x \rightarrow i$ *inverse quantization* is $i \rightarrow \widehat{x}^i$

Rate of a QS

- Number of bits needed to represent $i(n)$
- We assume $R = \log_2 L$
- Good approximention of real-life rates (entropy coders)

Distorsion

- We use the squared error:

$$d[x(n), \widehat{x}(n)] = |e(n)|^2 = \left|x(n) - \widehat{x}(n)\right|^2$$

- For a signal $x(\cdot)$ of duration N, we use the mean square error (MSE):

$$D = \frac{1}{N}\sum_{n=0}^{N-1} d[x(n), \widehat{x}(n)]$$

- For random signals,

$$D = \mathrm{E}\left\{|X(n) - Q(X(n))|^2\right\} = \mathrm{E}\left\{|E(n)|^2\right\}$$

- In this case, distortion is the variance of the random process $E(n) = X(n) - Q(X(n))$, and is indicated as σ_{Q}^2

TELECOM ParisTech

Uniform Quantization

A unifrom SQ (UQ) is characterized by:

- $\forall i, t^i = t^{i-1} + \Delta$
- $\widehat{x}^i = \frac{t^i + t^{i-1}}{2}$

UQ is simple, it minimizes the maximum error and is optimal for uniform RV's (random variables)
Moreover

- $\Delta^i = \Delta = 2A/L$

Uniform Quantization : distortion

Hypothesis : $X \sim \mathcal{U}(-A, A)$. Find $\sigma_Q^2 = \mathrm{E}\left[(X - \widehat{X})^2\right]$

Uniform Quantization : distortion

Hypothesis : $X \sim \mathcal{U}(-A, A)$. Find $\sigma_{\mathrm{Q}}^2 = \mathrm{E}\left[(X - \widehat{X})^2\right]$

$$\sigma_{\mathrm{Q}}^2 = \mathrm{E}\left[(X - \widehat{X})^2\right] \qquad = \int_{-A}^{A} p_X(u)[u - Q(u)]^2 du$$

Uniform Quantization : distortion

Hypothesis : $X \sim \mathcal{U}(-A, A)$. Find $\sigma_{\mathrm{Q}}^2 = \mathrm{E}\left[(X - \widehat{X})^2\right]$

$$\sigma_{\mathrm{Q}}^2 = \mathrm{E}\left[(X - \widehat{X})^2\right] = \int_{-A}^{A} p_X(u)[u - Q(u)]^2 du$$

$$\ldots = \sum_{i=1}^{L} \int_{\Theta^i} \frac{1}{2A}[u - \widehat{x}^i)]^2 du = \frac{1}{2A} \sum_{i=1}^{L} \int_{\widehat{x}^i - \Delta/2}^{\widehat{x}^i + \Delta/2} [u - \widehat{x}^i)]^2 du$$

$$= \frac{1}{2A} \sum_{i=1}^{L} \int_{-\Delta/2}^{\Delta/2} t^2 dt = \frac{1}{2A} L \frac{\Delta^3}{12} = \frac{\Delta^2}{12}$$

Quantization noise is actually a uniform RV in $(-\Delta/2, \Delta/2)$

Uniform Quantization : RD curve

$$D = \frac{\Delta^2}{12} \qquad = \frac{4A^2}{12L^2} \qquad = \frac{A^2}{3 \cdot 2^{2R}} \qquad = \sigma_X^2 2^{-2R}$$

$$\mathrm{SNR} = 10 \log_{10} \frac{\mathrm{E}\left\{X^2\right\}}{D} \qquad = 10 \log_{10} \frac{\sigma_X^2}{\sigma_X^2 2^{-2R}}$$

$$= 10 \log_{10} 2^{2R} \approx 6R$$

High Resolution (HR) Uniform Quantization

- Hypothesis : $L \to +\infty$, X generic RV
- In HR, in any given Θ^i we approximate p_X as a (different) constant.
- Therefore, the qunatization noise in Θ^i is $\mathcal{U}\left(-\frac{\Delta}{2}, \frac{\Delta}{2}\right)$
- From the total probability law, $E \sim \mathcal{U}\left(-\frac{\Delta}{2}, \frac{\Delta}{2}\right)$
- Donc :

$$D = \frac{\Delta^2}{12} = \frac{A^2}{3} 2^{-2R}$$

High Resolution (HR) Uniform Quantization

$$\mathrm{SNR} = 10\log_{10}\frac{\mathrm{E}\left\{X^2\right\}}{D} = 10\log_{10}\frac{\sigma_X^2}{A^2/3}2^{2R} \approx 6R - 10\log_{10}\frac{\gamma^2}{3}$$

where $\gamma^2 = \frac{X_{\mathrm{max}}^2}{\sigma_X^2} = \frac{A^2}{\sigma_X^2}$ is the *load factor*, i.e. the ratio between the peak power and the average power.

$$D = \frac{A^2}{3}2^{-2R} = \frac{\gamma^2}{3}\sigma_X^2 2^{-2R} = K_X\sigma_X^2 2^{-2R}$$

Scalar quantization: example on color image

Image Originale, 24 bpp

Scalar quantization: example on color image

Débit 21 bpp PSNR 47.19 dB TC 1.143

Scalar quantization: example on color image

Débit 18 bpp PSNR 42.38 dB TC 1.333

Scalar quantization: example on color image

Débit 15 bpp PSNR 36.97 dB TC 1.600

Scalar quantization: example on color image

Débit 12 bpp   PSNR 31.40 dB   TC 2.000

Scalar quantization: example on color image

Débit 9 bpp PSNR 29.26 dB TC 2.667

Scalar quantization: example on color image

Débit 6 bpp PSNR 27.83 dB TC 4.000

Scalar quantization: example on color image

Débit 3 bpp PSNR 25.75 dB TC 8.000

Plan

Predictive quantization

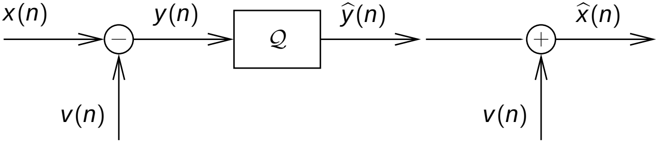
Principles

- Quantizaion alone is not effective for compression
- Too simple underlying model: SQ neglects dependencies among samples
- Idea: exploit sample correlation by prediction
- Goal: reduction of the signal's variance

Coding scheme

Open loop scheme

- $x(n)$ is predicted from the past
- If the prediction is good, $v(n) \approx x(n)$



- How to provide $v(n)$?
- What is the gain?

Prediction gain

Prediction Error = Signal Error :

$$q(n) = y(n) - \widehat{y}(n) = x(n) - v(n) - \widehat{x}(n) + v(n) = \bar{q}(n)$$

Therefore the target of predictive quantization (PQ) is to minimize the distortion of y

Coding gain:

$$\mathrm{SNR}_p = 10\log_{10}\frac{\sigma_X^2}{D} = 10\log_{10}\frac{\sigma_X^2}{\sigma_Y^2} + 10\log_{10}\frac{\sigma_Y^2}{D} = G_P + G_Q$$

Prediction is effective if and only if the prediction error has a smaller variance than the original signal

Example

$$X(n) \sim \mathcal{N}(0, \sigma^2) \qquad \mathrm{E}\left[X(n)X(m)\right] = \sigma^2 \rho^{|n-m|}$$

$$V(n) = X(n-1) \qquad \rho : G_P > 0 \ ?$$

Example

$$X(n) \sim \mathcal{N}(0, \sigma^2) \qquad \mathrm{E}\left[X(n)X(m)\right] = \sigma^2 \rho^{|n-m|}$$

$$V(n) = X(n-1) \qquad \rho : G_P > 0 \text{ ?}$$

$$Y(n) = X(n) - X(n-1) \text{ Zero mean Gaussian RV}$$

$$\sigma_Y^2 = \mathrm{E}\left[(X(n) - X(n-1))^2\right] = 2\sigma^2 - 2\sigma^2\rho$$

$$G_P = 10\log_{10}\frac{\sigma_X^2}{\sigma_Y^2} = 10\log_{10}\frac{\sigma^2}{2(1-\rho)\sigma^2}$$

$$G_P > 0 \Leftrightarrow \rho > \frac{1}{2}$$

Predictors

- Linear Predictors are simple and moreover optimal for Gaussian RV

$$v(n) = -\sum_{i=1}^{P} a_i x_{n-i} \qquad \text{Filter witht } P \text{ parameters}$$

$$y(n) = x(n) - v(n) = \sum_{i=0}^{P} a_i x_{n-i} \qquad \text{Prediction error}$$

- with $a_0 = 1$.
- y is the result of filtering x with

$$A(z) = 1 + a_1 z^{-1} + \ldots + a_P z^{-P}$$

- Optimal filter: minimization of σ_Y^2

AR model

- If $Y(z) = A(z)X(z)$, $X(z) = \frac{Y(z)}{A(z)}$
- If the prediction is optimal, $Y(z)$ is white noise with power σ_Y^2
- Thus, the spectral power density (SPD) of X is $S_X(f) = \frac{\sigma_Y^2}{|A(f)|^2}$
- The underlying model for X is autoregressive (AR):

$$X(z) = \frac{Y(z)}{1 + a_1 z^{-1} + \ldots + a_P z^{-P}} \quad \Leftrightarrow$$
$$x(n) + a_1 x(n-1) + \ldots + a_P x(n-P) = y(n)$$

- $X(n)$ is an AR filtering of white noise $Y(n)$

Predictor selection

Problème :
Find the vector $\underline{a}$ minimizing

$$\sigma_Y^2 = \mathrm{E}\left\{Y^2(n)\right\} = \mathrm{E}\left\{\left[X(n) + \sum_{i=1}^{P} a_i X(n-i)\right]^2\right\}$$

Predictor selection

$$\sigma_Y^2 = \mathrm{E}\left\{X^2(n)\right\} + 2\sum_{i=1}^{P} a_i \mathrm{E}\{X(n)X(n-i)\} + \sum_{i=1}^{P}\sum_{j=1}^{P} a_i a_j \mathrm{E}\{X(n-i)X(n-j)\}$$
$$= \sigma_X^2 + 2\underline{r}^t\underline{a} + \underline{a}^t \mathbf{R}_X \underline{a}$$

with:

$$\underline{r} = [r_X(1)\dots r_X(P)] \quad \mathbf{R}_X = \begin{bmatrix} r_X(0) & r_X(1) & r_X(2) & \dots & r_X(P-1) \\ r_X(1) & r_X(0) & r_X(1) & \dots & r_X(P-2) \\ r_X(2) & r_X(1) & r_X(0) & \dots & r_X(P-3) \\ \dots & \dots & \dots & \dots & \dots \\ r_X(P-2) & r_X(P-3) & r_X(P-4) & \dots & r_X(1) \\ r_X(P-1) & r_X(P-2) & r_X(P-3) & \dots & r_X(0) \end{bmatrix}$$

$$r_X(k) = \mathrm{E}\{X(n)X(n-k)\}$$

Predictor selection

Minimisation of Y variance:

$$\frac{\partial \sigma_Y^2}{\partial \underline{a}} = 2\underline{r} + 2\mathbf{R}_X \underline{a} = 0$$

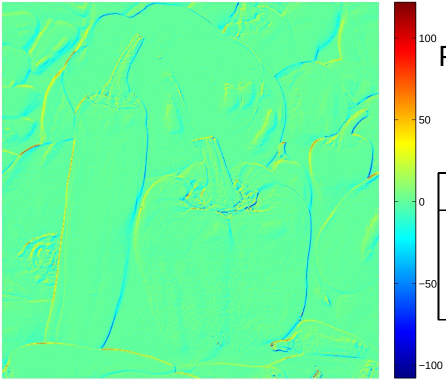
Thus:

$$\underline{a}^{\text{opt}} = -\mathbf{R}_X^{-1}\underline{r} \qquad \sigma_Y^2 = \sigma_X^2 + \underline{r}^t \underline{a}^{\text{opt}}$$

Autocorrelation r_X can b eestimated as

$$\widehat{r}_X(k) = \frac{1}{N} \sum_{n=0}^{N-1-k} X(n)(X(n+k))$$

Predictive quantization: example



Predictor:

$$\hat{f}_{n,m} = af_{n-1,m} + bf_{n,m-1}$$

a	b	σ_Y^2
0	0	2902.7
1/2	1/2	78.7
0.449	0.546	78.4