Basics of Watermarking

Ton Kalker
Philips Research & University of Eindhoven
Ton.Kalker@ieee.org

Overview

- Definition
- Why watermarking?
- Example
- Spread-Spectrum
- Matched Filtering
- Watermark parameters
- Attacks

Overview

- Definition
- Why watermarking?
- Example
- Spread-Spectrum
- Matched Filtering
- Watermark parameters
- Attacks

Watermarking =

- The art of actively modifying audio-visual content such that the modifications
  - Are imperceptible (who is the listener?),
  - Carry retrievable information,
  - That survives under degradations of the content,
  - *And is difficult to remove & change by unauthorized users (cryptography)*.
- Watermarking is not adding meta-data to header fields!

Overview

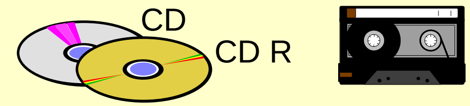
- Definition
- Why watermarking?
- Example
- Spread-Spectrum
- Matched Filtering
- Watermark parameters
- Attacks

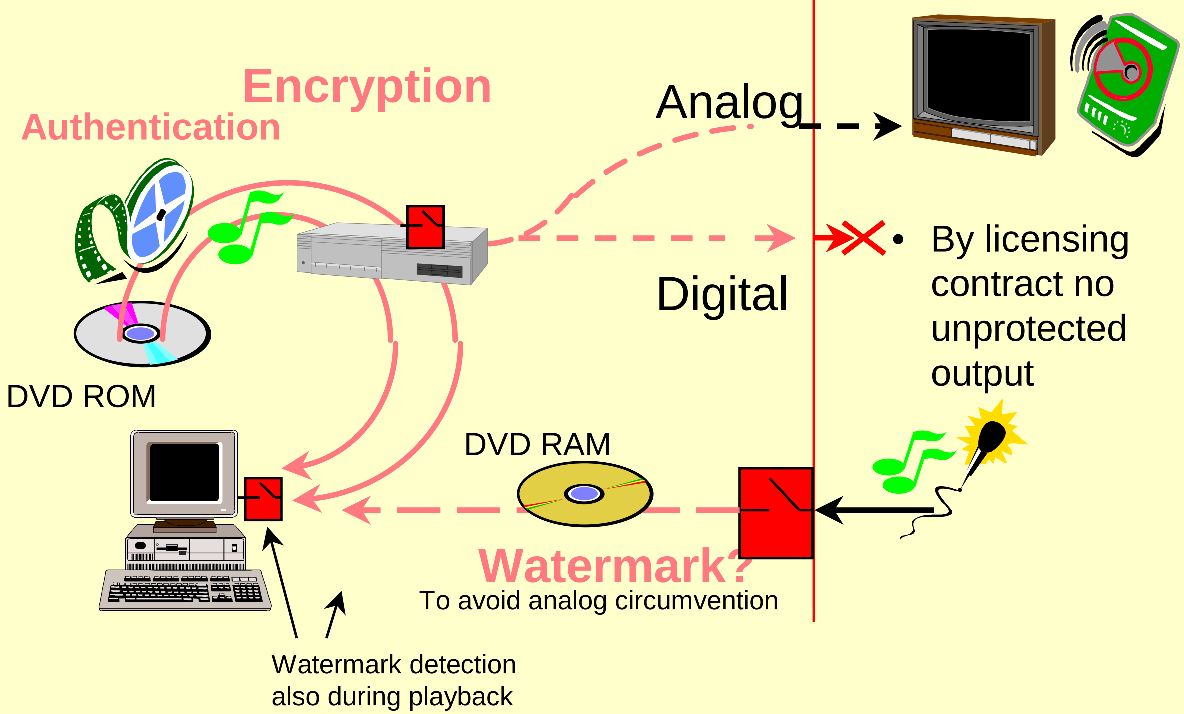
Compliant World

- All content is encrypted on all digital interfaces
- Link-by-link encryption; devices internally process clear content
- Controlled by CSS, 5C, 4C, ...
- Includes DVD players, DVD RAM, SDMI audio, DVD audio, PC's

Non-Compliant World

- All analog devices, some digital
- Marginalized by standardization efforts



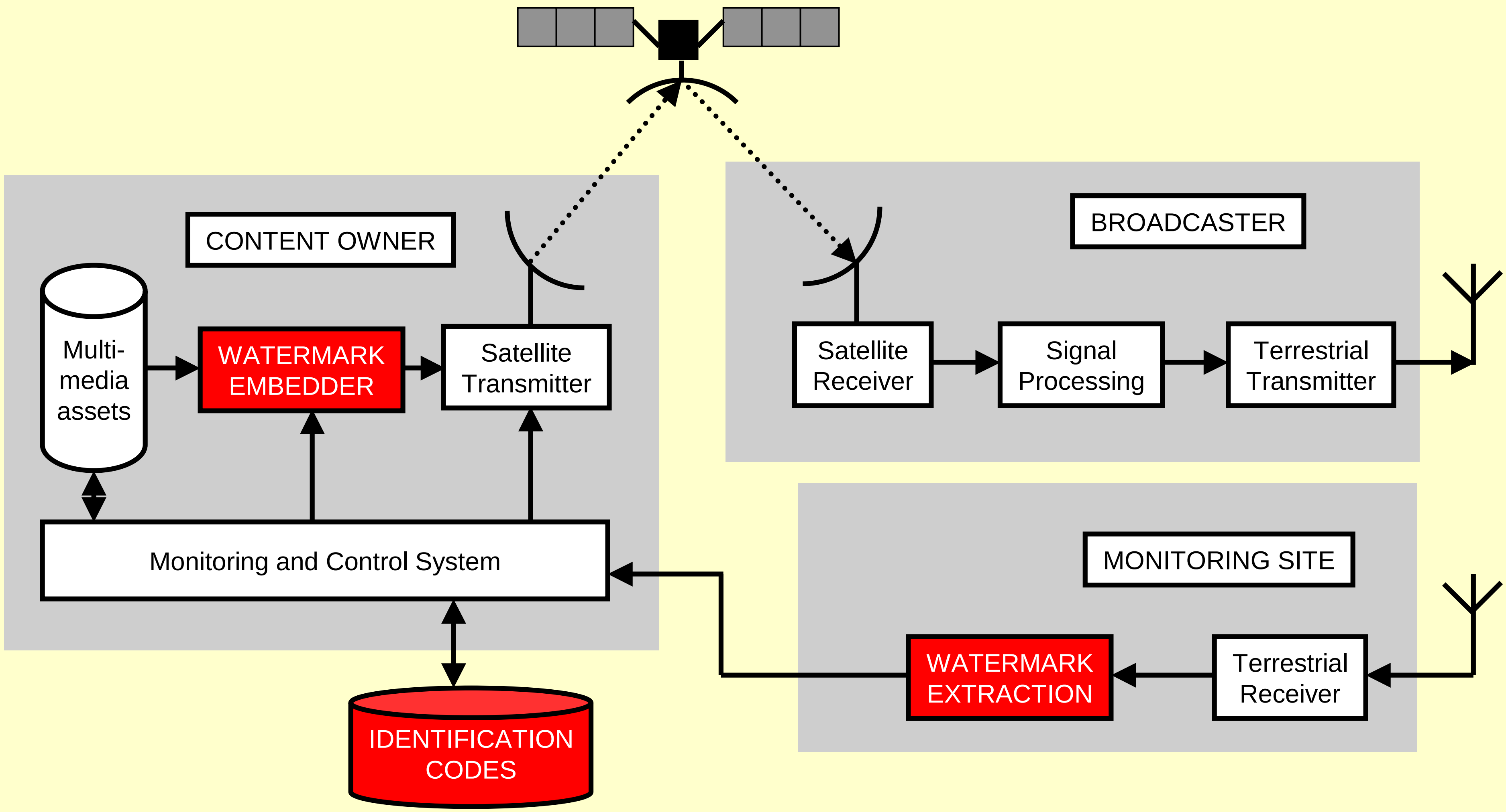


- Macrovision spoilers
- Watermarks

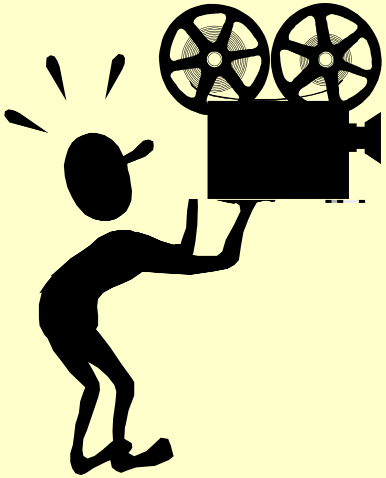
- By licensing contract no unprotected output

- New laws in US and EU

Broadcast Monitoring

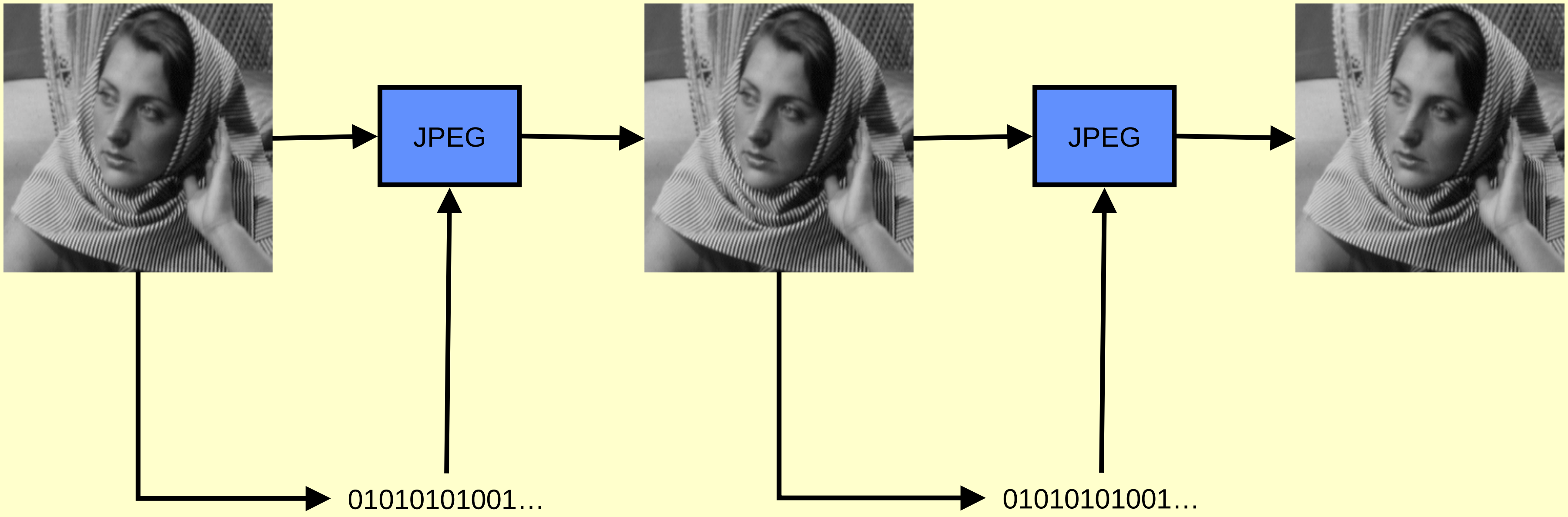


Digital Cinema

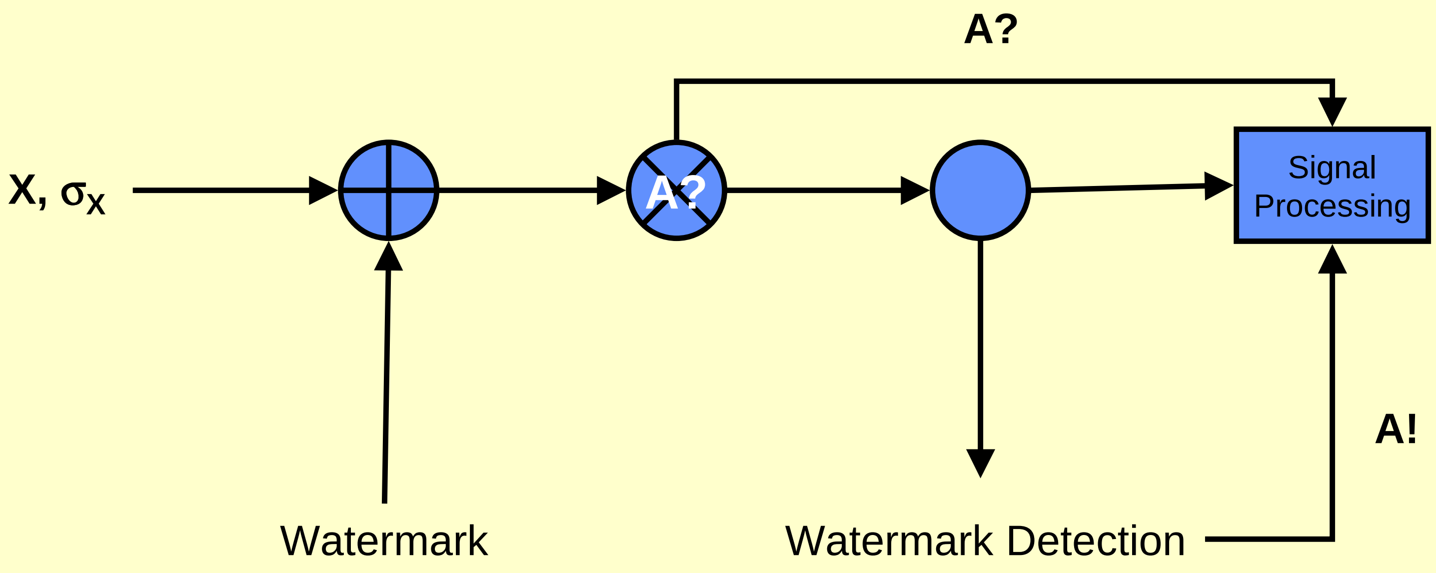


Name That Tune

Helper Data for Processing



Helper Data for Calibration



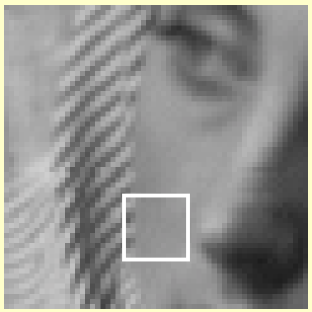
Overview

- Definition
- Why watermarking?
- Example
- Spread-Spectrum
- Matched Filtering
- Watermark parameters
- Attacks

Low-bit Modulation

- Early scheme: alter LSB or low-order bits

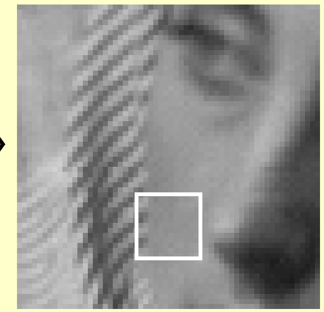
Original



```
130=1000 0010
123=0111 1011
117=0111 0101
 :     :     :
```

Embed 101... →

After embedding



```
131=1000 0011
122=0111 1010
117=0111 0101
 :     :     :
```

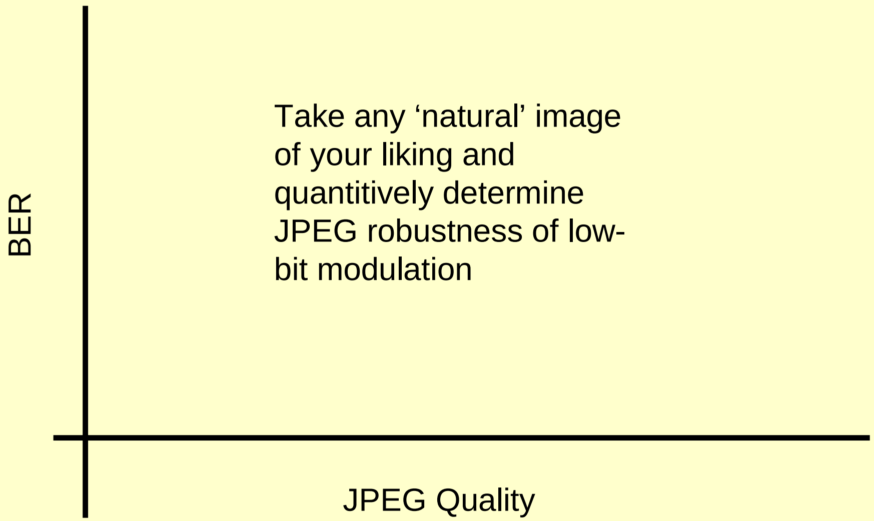
→ imperceptible (modify only LSBs)

→ secure (encrypt embedded information)

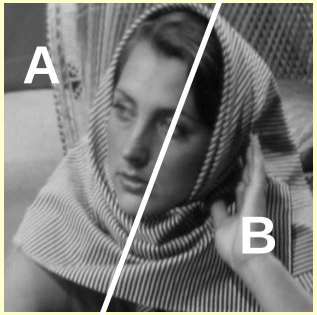
→ not robust (e.g., randomly set LSBs to 0 or 1)

- More accurate: secure info-hiding method

Low Bit Modulation



Patchwork



- 2 disjoint sets, *A* and *B*, of *N/2* pixels each
  - pixels in each set ("patch") chosen randomly
  - assumption:

$$S = \left(\sum_i A_i - \sum_i B_i\right) \big/ N \approx 0$$

  - embedding bit b ={-1,+1}: $A'_i \leftarrow A_i$+b*1, $B'_i \leftarrow B_i$-b*1

$$S' = \left(\sum_i A_i' - \sum_i B_i'\right) \big/ N = \\ \left(\sum_i A_i - \sum_i B_i\right)/N + \\ + (N/2 - (-N/2))/N \approx b$$

  - if $|S'| \approx 1$, watermark present with value *sign(S')*
- Prototypical spread-spectrum watermarking
  - communicate information via many small changes

Overview

- Definition
- Why watermarking?
- Example
- Spread-Spectrum
- Matched Filtering
- Watermark parameters
- Attacks

Spread-Spectrum Watermarking

- Original Signal x[i] (Gaussian, iid, σ_X,…)
- Watermark w[i] (Gaussian, iid, σ_W,…)
- Watermarked Signal
  - (1/2)-bit version (*copy protection*)
    - H0: Y[i] = X[i]
    - H1: Y[i] = X[i] + W[i]
  - 1-bit version (*helper data*)
    - H0: Y[i] = X[i] – W[i]
    - H1: Y[i] = X[i] + W[i]

Spread-Spectrum Watermarking

- Received Signal Z[i]
  - Distinguish between two hypotheses H0 and H1.
- Maximum likelihood testing
  - (Gaussian, iid) optimal tests statistic given by correlation
  - $D = (\Sigma_i Z[i] W[i]) / N$
- Not Marked : Z = X

  - $E[D] = (\Sigma_i E[X[i]] E[W[i]]) / N = 0$

  - $E[D^2] = E[(\Sigma_i X[i] W[i])^{2]} / N^2 =$
    $= (\Sigma_i E[X[i]^2] E[W[i]^2]) / N^2 =$
    $= \sigma_X^2 \sigma_W^2 / N$

Spread-Spectrum Watermarking

- Marked : $Z = X + W$
  - $E[D] = \sigma_W^2$
  - $\sigma_D^2 = \sigma_X^2 \sigma_W^2 / N$

- For N large D is approximately Gaussian distributed
- Error rate determined by $Q(D / \sigma_D)$
- Marked : $E[D] / \sigma_D = \text{Sqrt}(N) (\sigma_W / \sigma_X)$

- Robustness increases with
  - More samples
  - More watermark energy
  - Less host interference

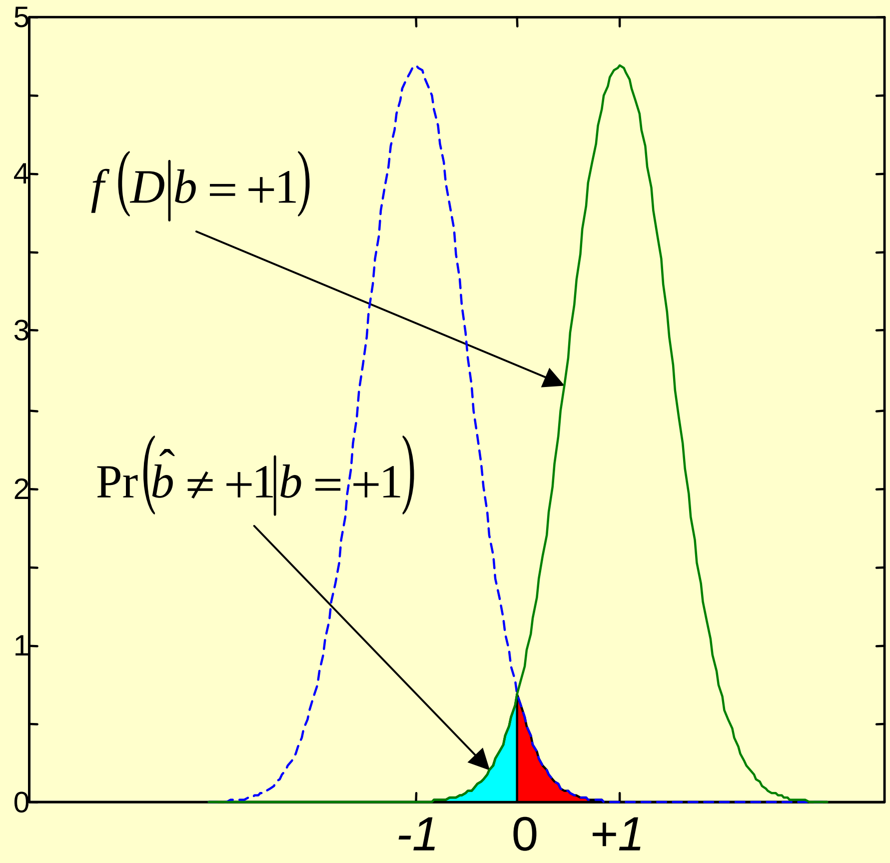
Detection (effectiveness)

- Correlation sum D
  - assumed Gaussian
  - $\sigma_W = 1$
  - variance $\sigma_X^2/(N)$
- Decision rule becomes

$$\hat{b} = \begin{cases} +1, & \text{if } D > 0; \\ -1 & \text{if } D < 0. \end{cases}$$

- Probability of error
  - Q function

$$Q\left(\frac{\sqrt{N}}{\sigma}\right)$$



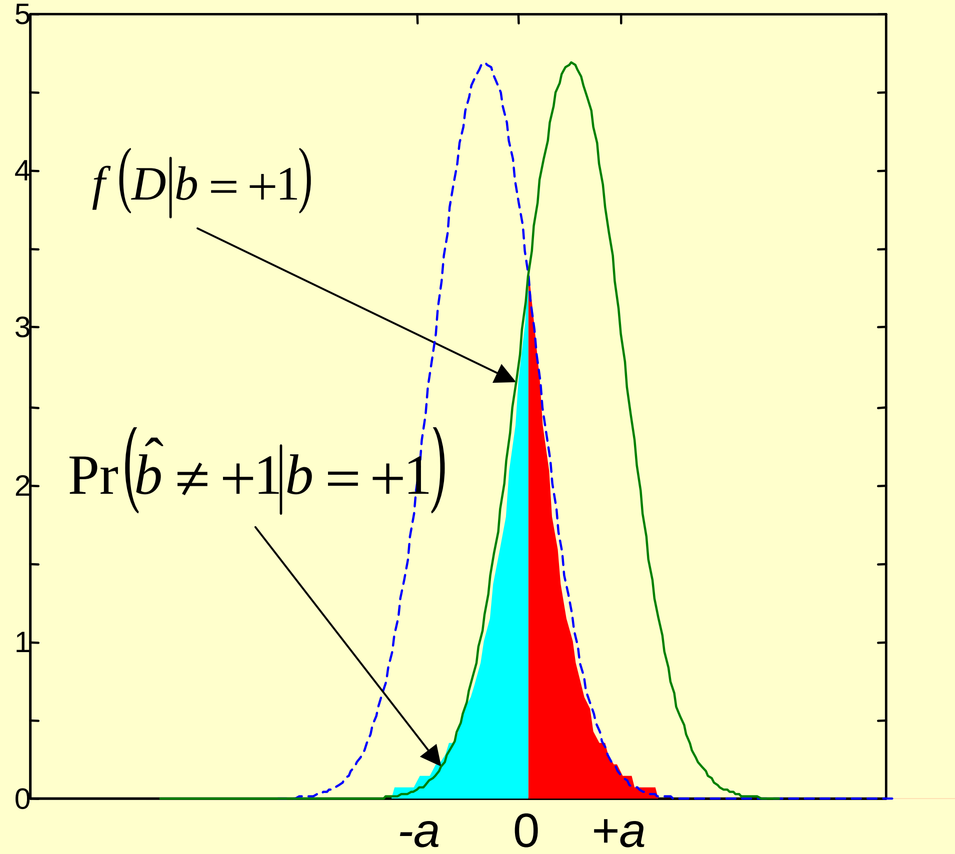
Detection (robustness)

- Correlation sum D
  - assumed Gaussian
  - mean $-a, +a$
  - variance $\sigma_X^2/(N)$
- Decision rule becomes

$$\hat{b} = \begin{cases} +1, & \text{if } D > 0; \\ -1 & \text{if } D < 0. \end{cases}$$

- Probability of error
  - Q function

$$Q\left(a\frac{\sqrt{N}}{\sigma}\right)$$



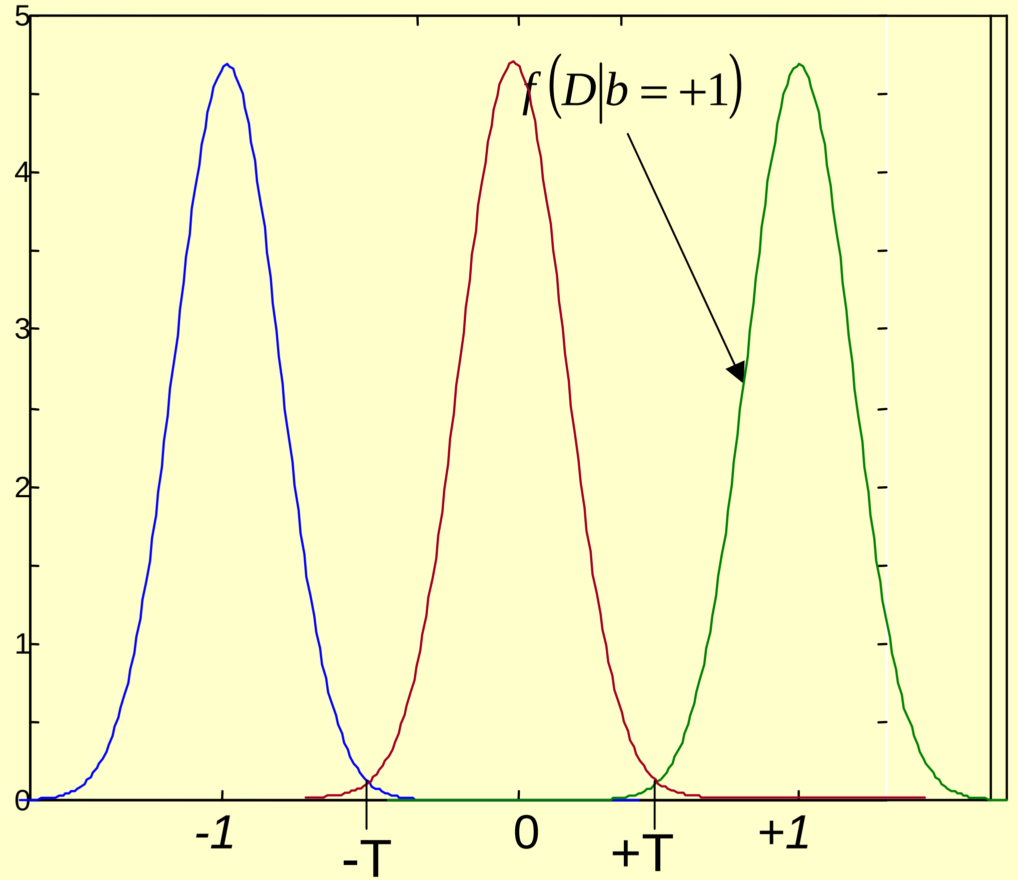
Detection (false positives)

- Correlation sum D
  - assumed Gaussian
  - mean $-1, 0, +1$
  - variance $\sigma_X^2/(N)$
- Decision rule becomes

$$\hat{b} = \begin{cases} +1, & \text{if } D > +T; \\ -1, & \text{if } D < -T; \\ 0, & \text{if } |D| \le T. \end{cases}$$

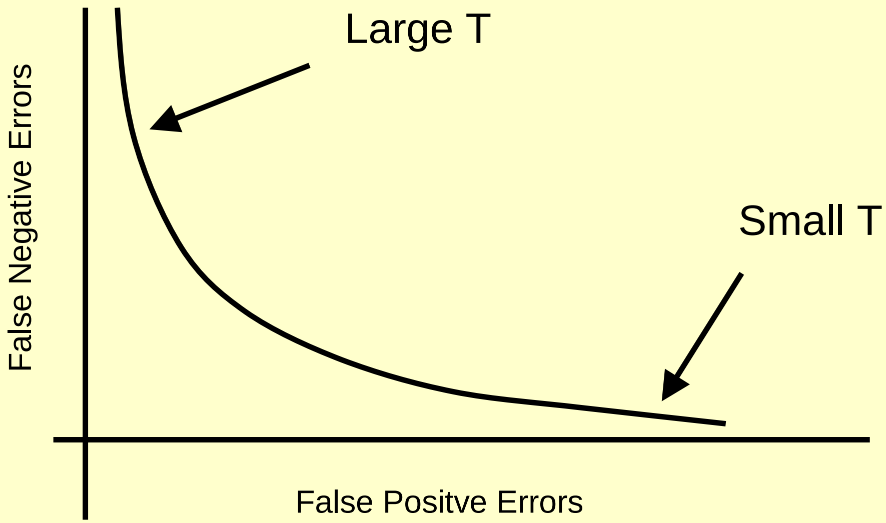
- Probability of false positive

$$2Q\left(T\frac{\sqrt{N}}{\sigma}\right)$$

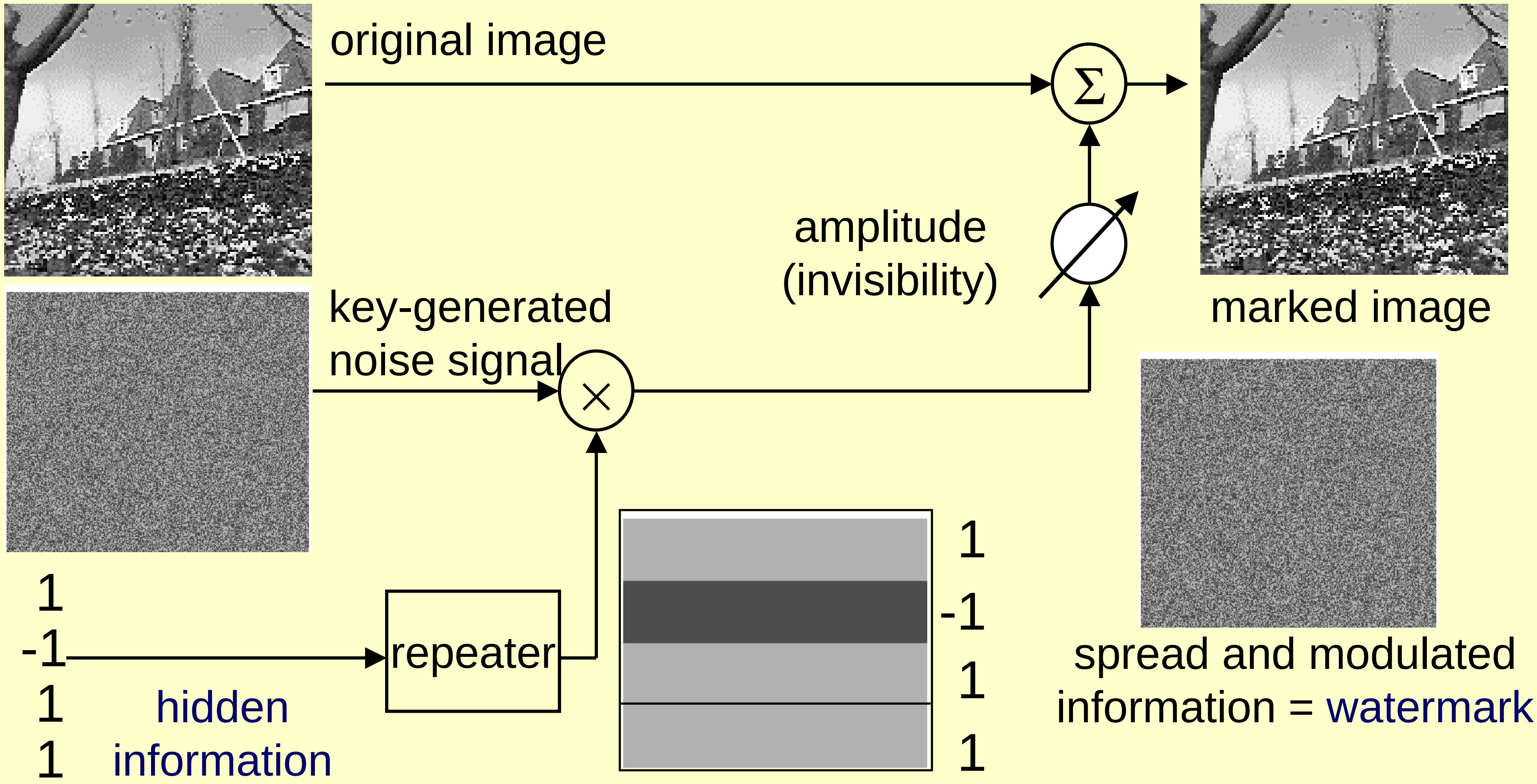


Spread-Spectrum Watermarking

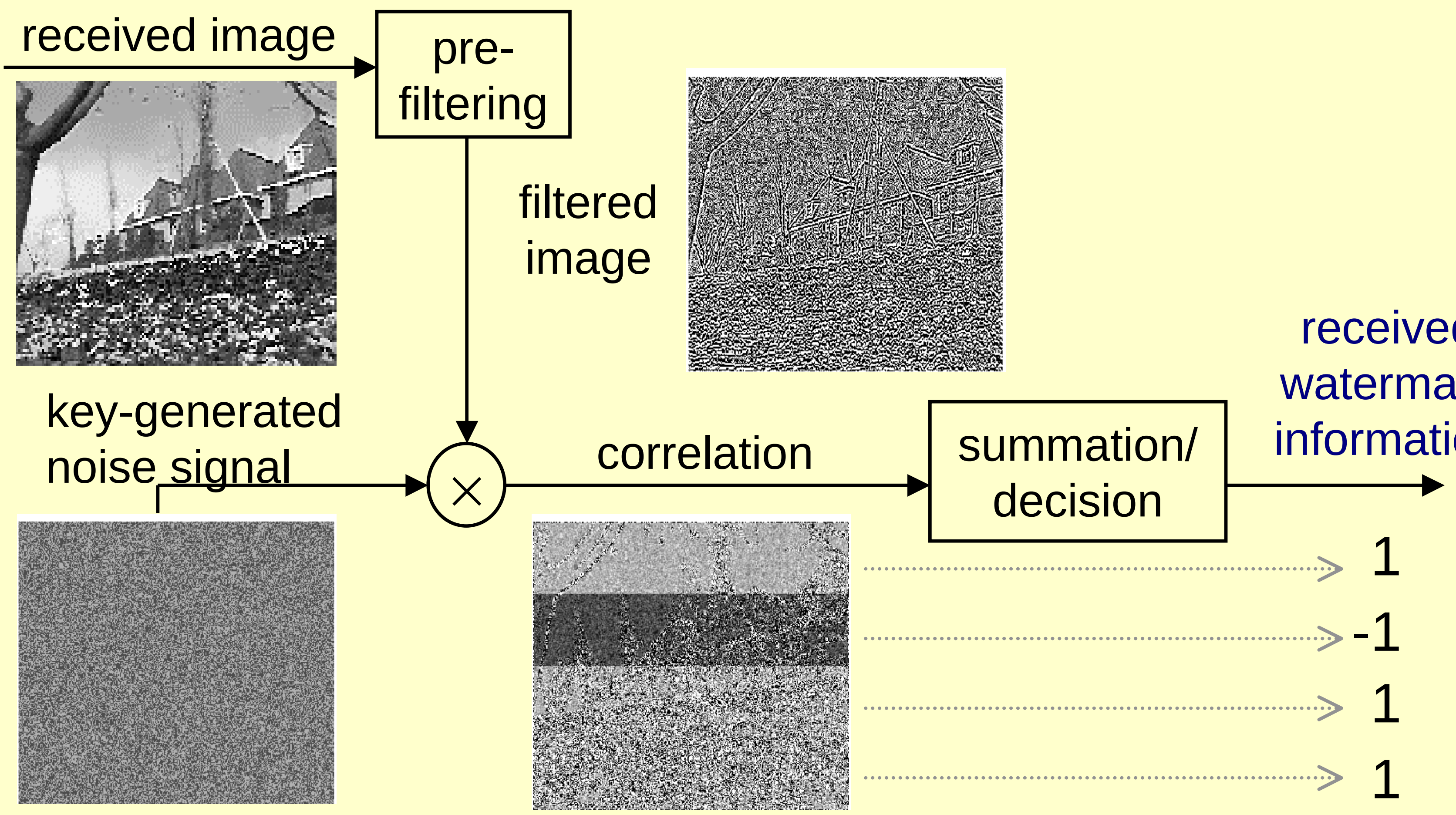
Receiver-Operator Characteristic (ROC) Curve



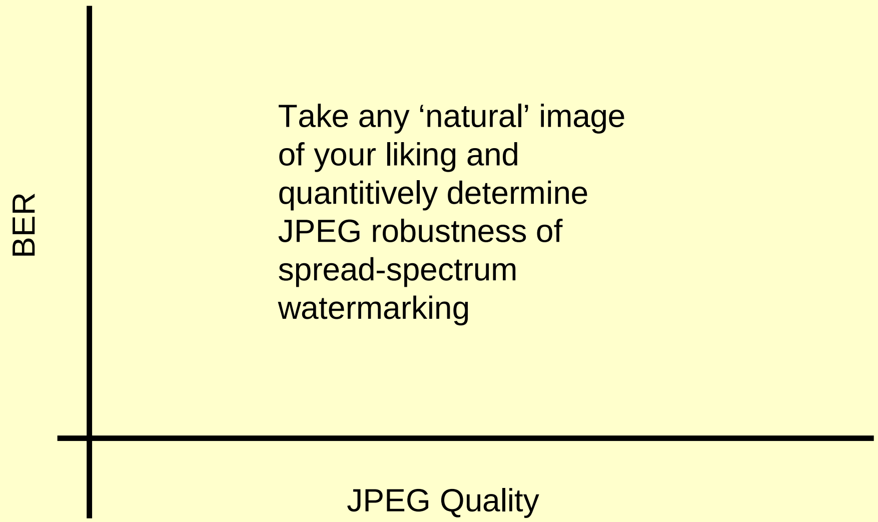
Watermark Embedding



Watermark Retrieval



Spread-Spectrum Watermarking



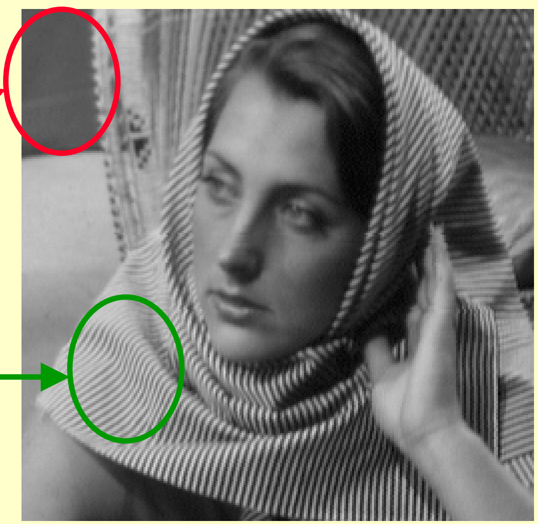
Take any ‘natural’ image of your liking and quantitively determine JPEG robustness of spread-spectrum watermarking

Perceptual Watermarking

- Original x.
- Apply transform T: $y = T(x)$
  - T = I, DCT, FFT, log, … (or any combination thereof)
- Add pseudo-random sequence w: $z = y + w$
  - Allow adaption of w to host signal
    - $\mathbf{Z = Y + \alpha W}$
  - In position
    - only in textured image regions, not in silence
  - In value
    - less energy in flat regions than in textured regions
- Apply inverse transform: $m = T^{-1}(z)$

Perceptual Watermarking

- Example: PatchWork
- T = I
  - Spatial watermarking
- $w = X_A - X_B$
  - Binary {-1,+1}-valued pseudo-random sequence
- Adaptation, e.g.
  - Less power in flat regions
  - More power in textured regions



Perceptual Watermarking

- Received data m'
- Apply inverse transform T^{-1}: $z' = T^{-1}(m')$
- Assume $z' = y' + h*w$
  - Hypothesis testing
  - $h = 0$: not watermarked
  - $h = 1$: watermarked
- Determine optimal detector
  - Prefilter + correlation
  - $D = \langle y',w \rangle + h \langle w,w \rangle$

Popular Example: NEC Scheme

- Heuristic claim
  - watermark should be embedded in the "perceptually significant frequency components" for best robustness
- Embedding
  - N watermark samples w_i ~N(0,1); e.g., $N = 1000$
  - embed in the N largest-amplitude DCT coefficients (except DC coefficient) x_i

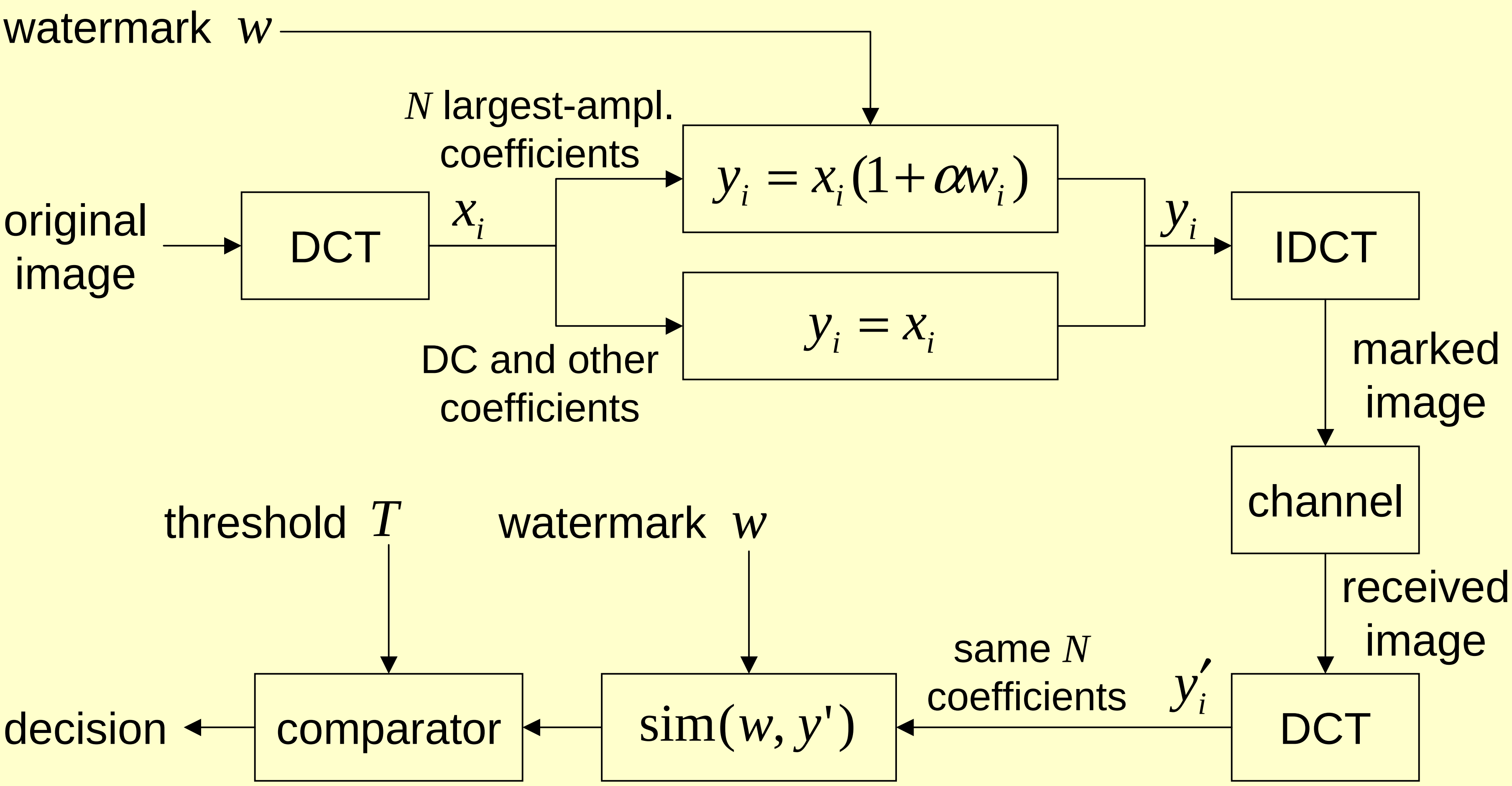
$$y_i = x_i(1 + \alpha w_i)$$

- Detection
  - extract the same N DCT coefficients y_i'
  - compute the similarity (normalized correlation) between y_i' and w_i

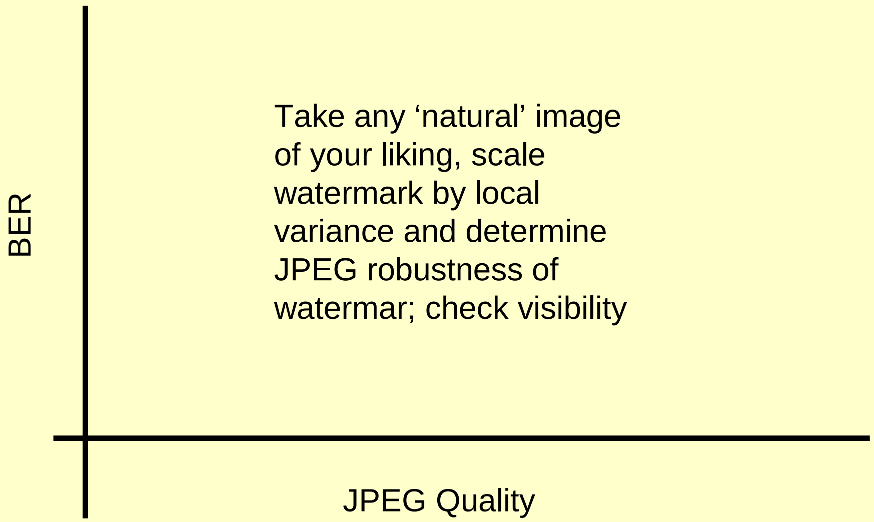
$$\text{sim}(w, y') = \frac{\langle w, y' \rangle}{\sqrt{\langle y', y' \rangle}}$$

  - watermark w is present if $\text{sim}(y', w) > T$

Block Diagram of NEC Scheme



Perceptual Watermarking



TU/e PHILIPS

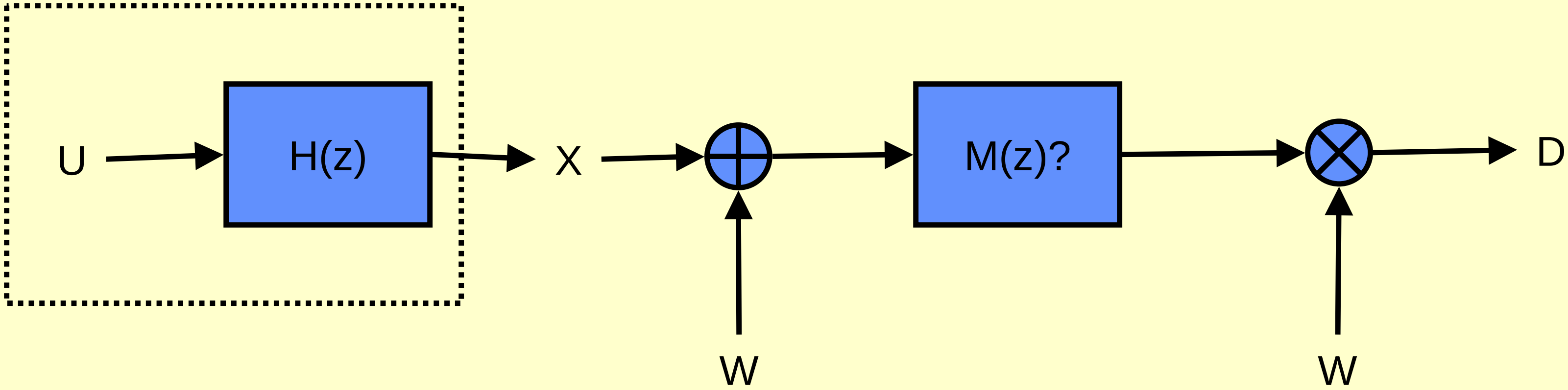
Overview

- Definition
- Why watermarking?
- Example
- Spread-Spectrum
- Matched Filtering
- Watermark parameters
- Attacks

Matched Filtering

- Audio-visual data are usually not well modelled as Gaussian iid sources!
- For images (for neighbouring pixels)
  - $E[X[i]\, X[i+1]] / \sigma_X^2 \approx 0.9$
- Better model X = H * U, where
  - H is low pass
  - U is random iid source
- Example : X[i+1] = a X[i] + U[i+1]
  - $a \approx 0.9$
  - $H(z) = (1 - a\, z^{-1})^{-1}$

Matched Filtering



- Correlation in z-domain notation
  - $A(z) = \Sigma\, a_i\, z^{-i}$
  - $[A(z)]_0 = a_0$
  - $\Sigma\, a_i\, b_i = [A(z)\, B(z^{-1})]_0$

- $D = [(M(z)\, H(z)\, U(z) + M(z)\, W(z)\,)\, W(z^{-1})]_0$

Matched Filtering

- Cost function
  - $C_M =$

    $= (\text{Righthand term})^2 / E[\text{variance lefthand term}]$

    $= [M(z)W(z)\ W(z^{-1})]_0^2 / E[[(M(z)\ H(z)\ U(z)\ W(z^{-1})]_0^2]$

- Simplification
  - $C_M =$

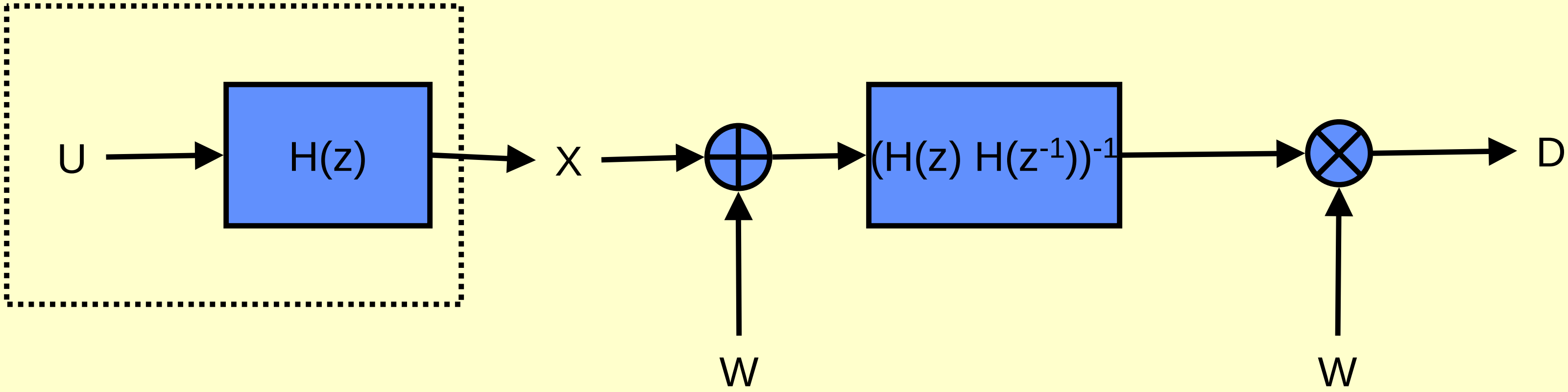
    $= (N^2\,[M(z)]_0\ \sigma_W^4) / (N\ \sigma_W^2\,\sigma_U^2[M(z)\ M(z^{-1})\ H(z)\ H(z^{-1})]_0)$

    $= N\ \ (\sigma_W^2 / \sigma_U^2)\ \ ([M(z)]_0 / [M(z)\ M(z^{-1})\ H(z)\ H(z^{-1})]_0)$

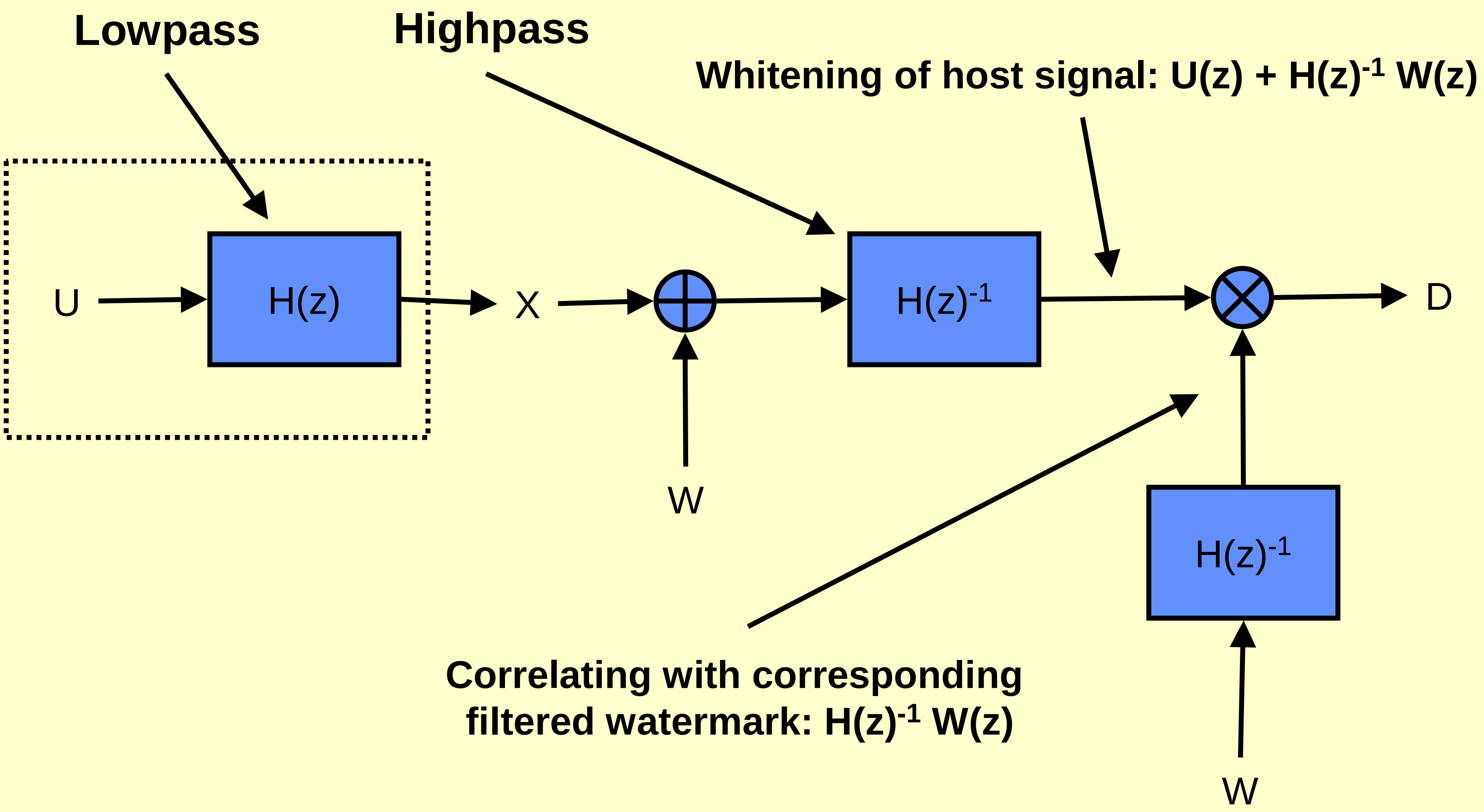
Matched Filtering

- Optimize in the frequency domain
  - $\mu_i = M(\omega_i)$, $\eta_i = H(\omega_i)$
  - $C_M = \Sigma\, \mu_i / (\Sigma\, \mu_i^2\, \eta_i^2)$
  - We may assume $\Sigma\, \mu_i = 1$
  - Using Lagrange multipliers we find
  - $\mu_i = 1 / \eta_i^2$
  - $M(z) = (H(z)\, H(z^{-1})\,)^{-1}$

Matched Filtering



Matched Filtering



Spread-Spectrum Watermarking

BER

Take any 'natural' image of your liking and quantitively determine JPEG robustness of spread-spectrum watermarking using 'matched filtering' with separable filter [1 –1; -1 1]

JPEG Quality

Overview

- Definition
- Why watermarking?
- Example
- Spread-Spectrum
- Matched Filtering
- Watermark parameters
- Attacks

Watermark Parameters

- Perceptibility
  - perceptibility of the watermark in the intended application

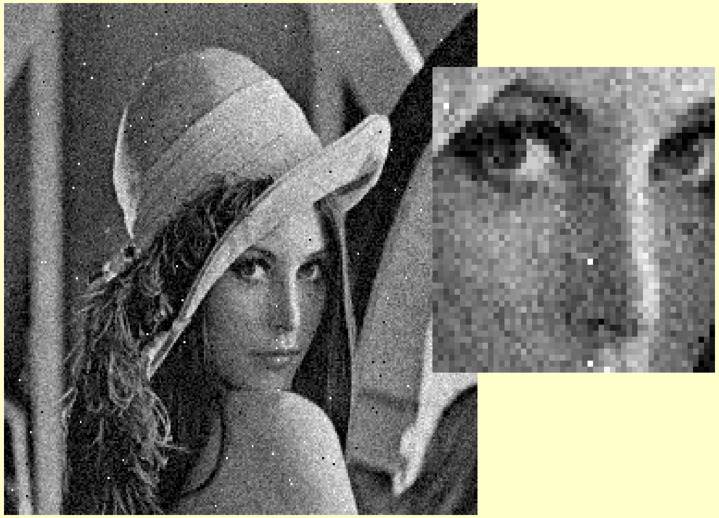
Original image

Image + hidden information

Watermark Parameters

- Robustness
  - resistance to (non-malevolent) quality respecting processing

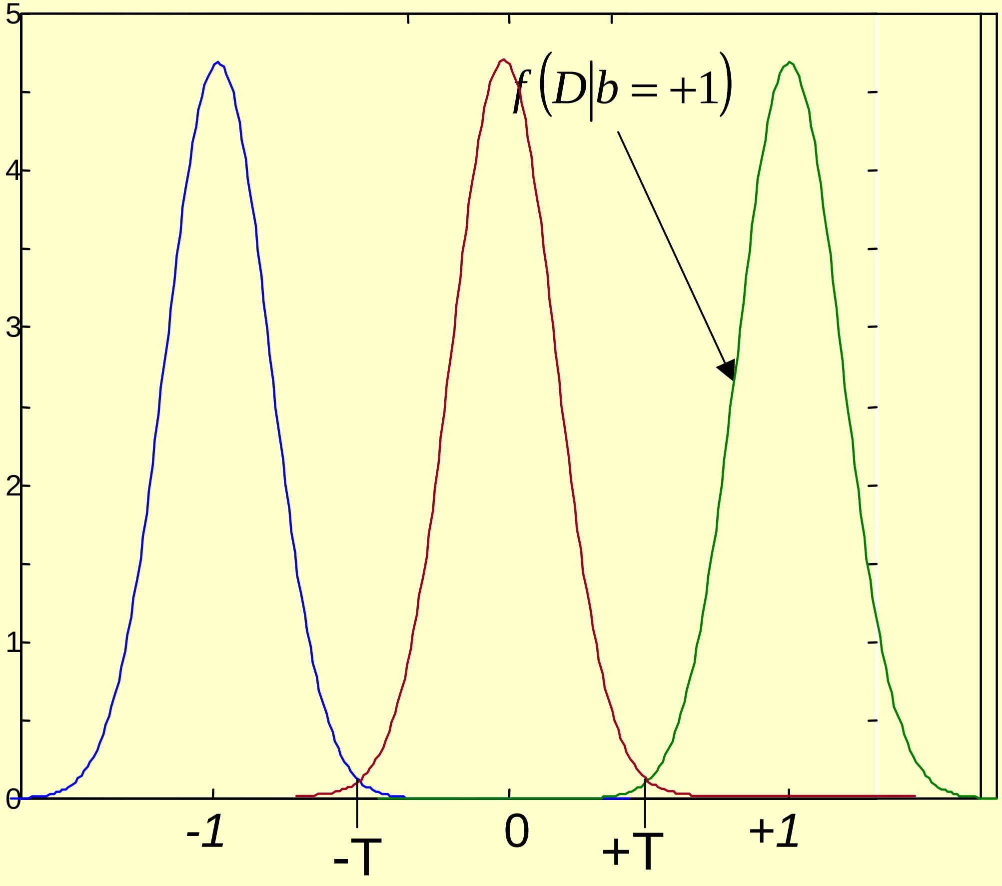
JPEG compression



Additive noise & clipping

Watermark Parameters

- Error Rates



Watermark Parameters

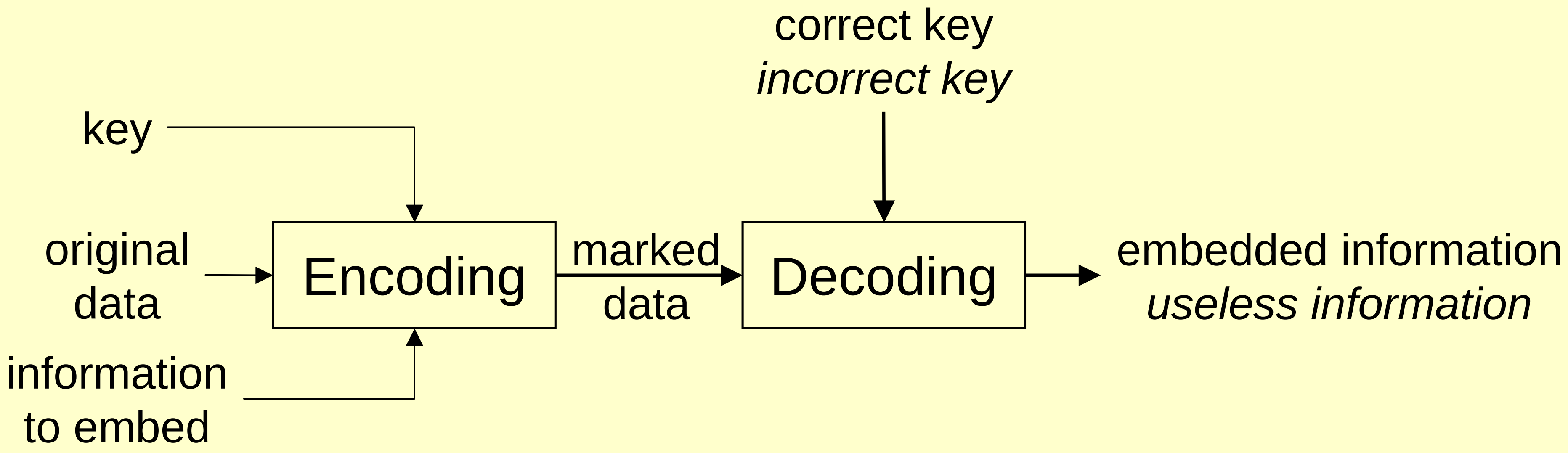
- Complexity
  - hardware & software resources, real-time aspects
  - baseband vs. compressed domain
- Granularity
  - minimal spatio-temporal interval for reliable embedding and detection
- Capacity
  - related to payload
  - #bits / sample

Watermark Parameters

- Layering & remarking
  - watermark modification
- Security
  - vulnerability to intentional attacks
  - Kerkhoffs' principle

Security

- Embedded information cannot be detected, read (interpreted), and/or modified, or deleted by unauthorized parties
- Kerckhoff's principle: Security resides in the secrecy of the key, not in the secrecy of the algorithm.

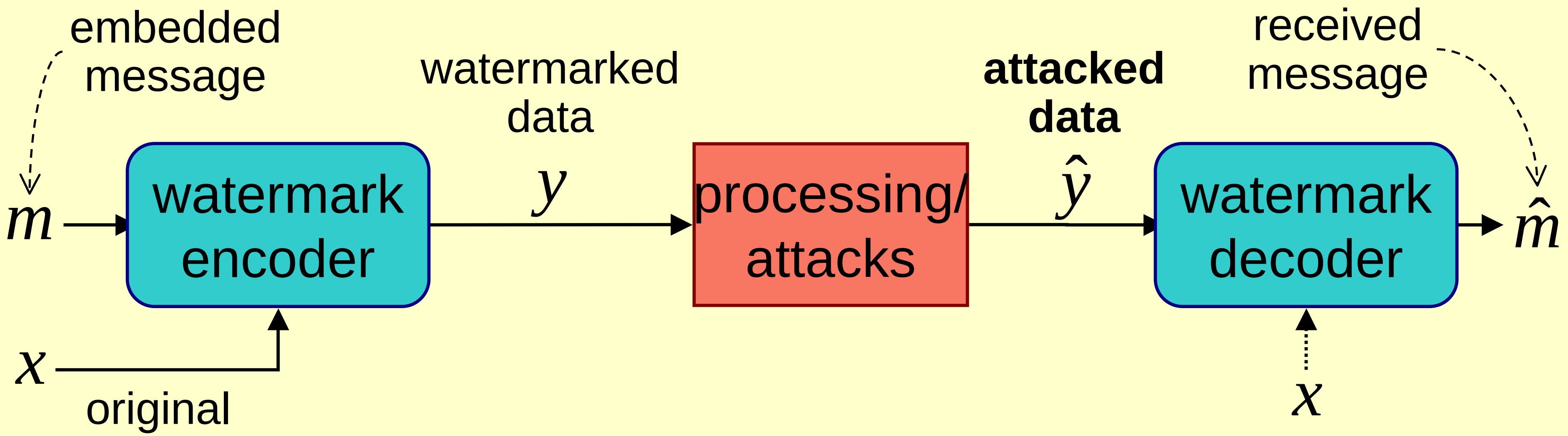


Overview

- Definition
- Why watermarking?
- Example
- Spread-Spectrum
- Matched Filtering
- Watermark parameters
- Attacks

Attacks and Communications Viewpoint

- Watermarked data will likely be processed
- Attack - any processing that may coincidentally or intentionally damage the embedded information
- Treat attacks like a communications channel



Evaluating Robustness

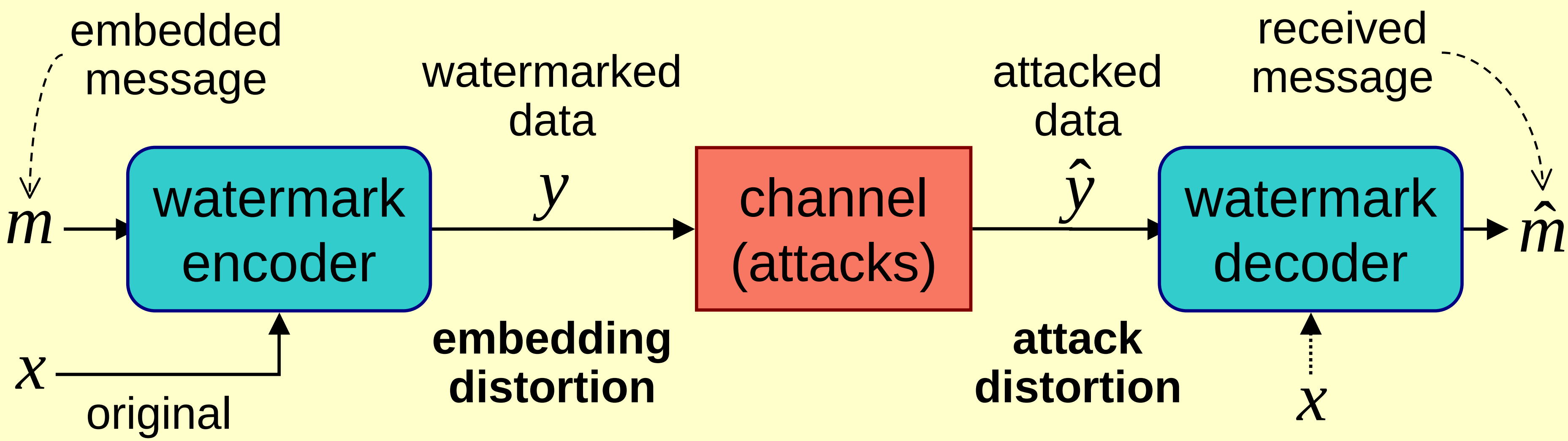
- Robustness: easy to define, hard to evaluate
  - Embedded information cannot be damaged or destroyed without making the attacked data useless
  - How to evaluate robustness in a well-defined sense?

"A watermark is robust if communication cannot be impaired without rendering the attacked data useless."

- Kerckhoff's principle
  - Assume opponent has complete knowledge of your strategy (algorithm and implementation) but lacks a secret (key).

Need for a Distortion Measure

- When is the attacked data useless?
- Quantify “usefulness” of attacked data
- Multimedia $\rightarrow$ measure distortion of attacked data
  - inherently subjective, always debatable
  - imperfect but measurable

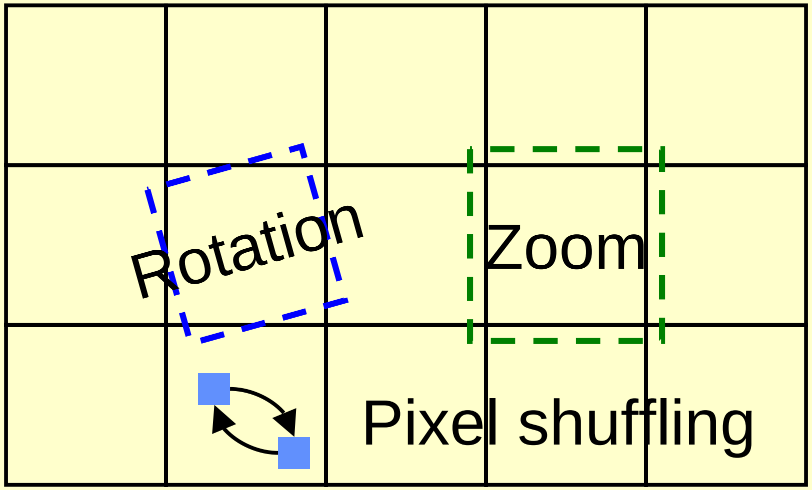


Classes of Attacks

- Simple waveform processing
  - “brute-force” approach
    - impairs watermark and original data
    - compression, linear filtering, additive noise, quantization
- Detection-disabling methods
  - disrupt synchronization
    - geometric transformations (RST), cropping, shear, re-sampling, shuffling
    - watermark harder to locate
  - distortion metric not well defined
  - meaning of watermark presence?
    - change of ROC curve!
- Advanced jamming/removal
  - intentional processing to impair/defeat watermark
    - watermark estimation, collusion (multiple copies)
- Ambiguity/deadlock issues
  - reduce confidence in watermark integrity
    - creation of fake watermark or original, estimation and copying of watermark signal

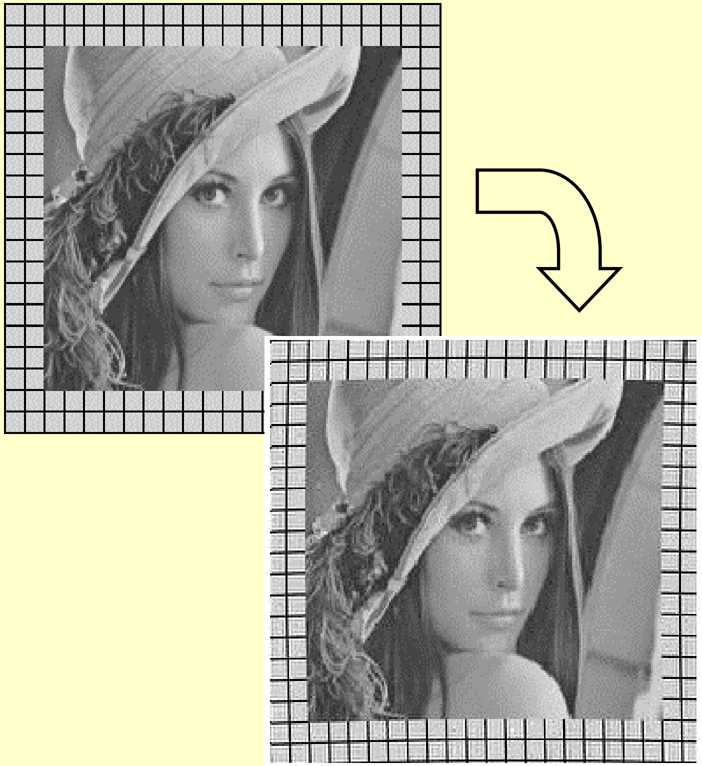
De-synchronization

- Attack
  - harder to find watermark
  - does not remove watermark
- How to measure distortion?
- Spread spectrum
  - fails without sync
  - re-synchronizing difficult
    - noiselike carrier
    - no peaks in frequency



StirMark

- Popular, free WWW software
  - simulate printing and scanning
  - nonlinear geometric distortion + JPEG
- Easy to use and test
- Limitations
  - features available elsewhere
  - purely empirical
    - does not suggest how to improve system
  - does not use Kerckhoff's principle!
    - does not target system weaknesses
    - suboptimal attack
    - false sense of security



Resynchronization Methods

- Use of templates
  - pattern of peaks in frequency domain
    - attacker can locate pattern, too!
  - pattern of local extrema
    - harder for attacker to locate or recognize
    - harder for receiver, too
  - seeking pattern is like seeking watermark signal

- Invariant representations
  - translation invariance
    - Fourier magnitude
  - rotation and scael invariance
    - log-polar mapping

      $$(x, y) \leftrightarrow (\mu, \theta)$$

      $$x = e^{\mu} \cos\theta, y = e^{\mu} \sin\theta$$

    - Fourier-Mellin transform
  - cannot handle aspect ratio changes, shear, etc.

Translation Robustness

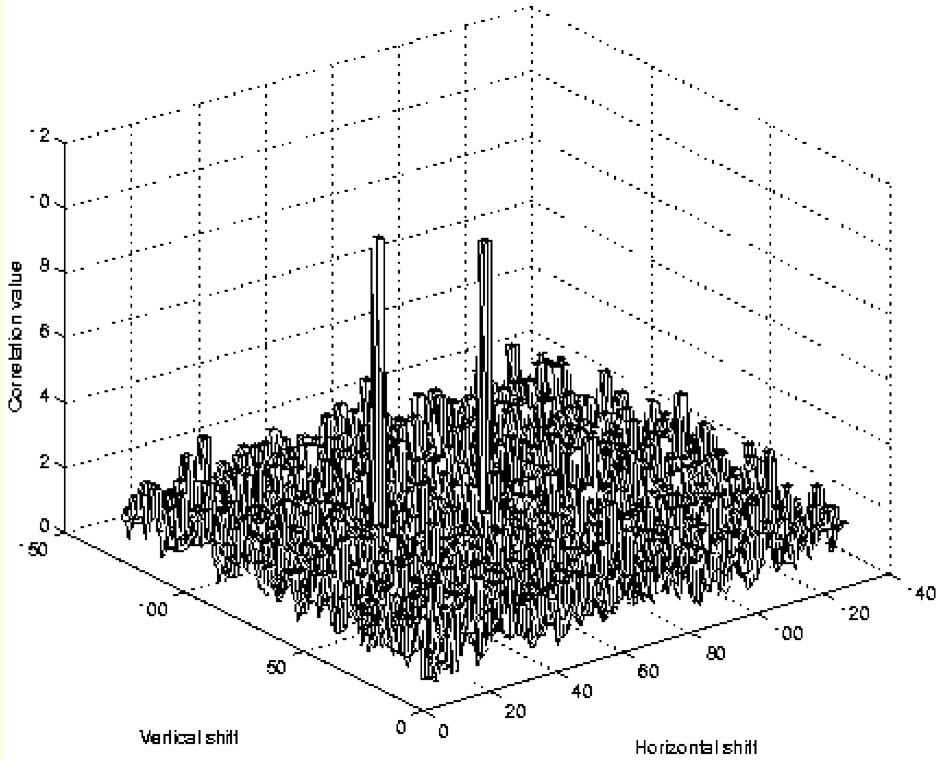
- Original Marked Data
  - **$Y[i] = X[i] + W[i]$**
- Translated Data
  - **$Z[i] = Y[i+k]$**
- Detector strategy (k unknown)
  - Trial and error: correlating at shifted positions
  - **$D[i] = \Sigma_l Z[i-l] W[i]$** (exhaustive search)
  - **$D(z) = W(z) Z(z^{-1})$**
  - Efficient computation with Fourier transform!
  - **$D = FFT^{-1}(FFT(W) * FFT^{*}(Z))$**

Translation Robustness

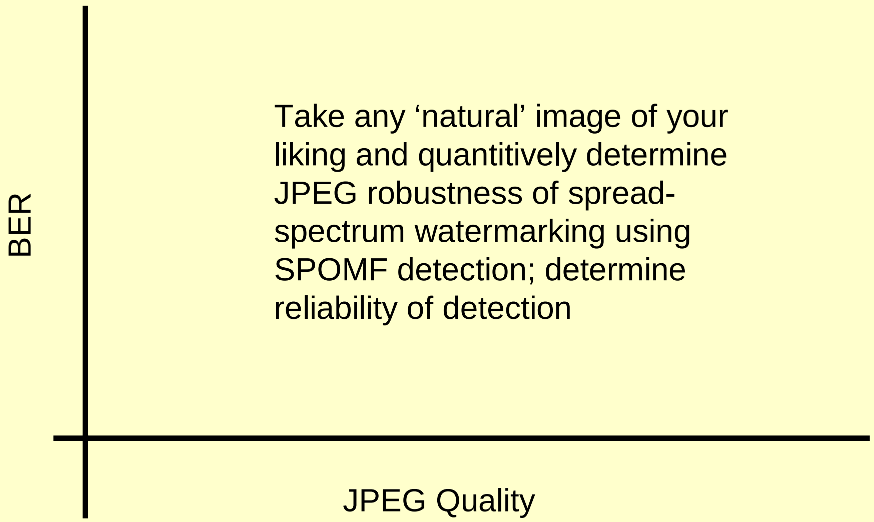
- Integration with Matched Filter
  - **$D = FFT^{-1}(FFT(W) * FFT^{*}(Z) * |FFT(H)|^{-2})$**
  - In many cases, W and H are fixed and their Fourier transforms can be pre-computed and stored.
- Experimentally, retaining only phase information
  - Symmetrical Phase-Only Matched Filtering (SPOMF)
  - **$D = FFT^{-1}(Phase(FFT(W)) * Phase(FFT^{*}(Z)))$**
  - **$Phase(a\, e^{2\pi i\omega}) = e^{2\pi i\omega}$**

Translation Robustness

- Most values D[i] correspond to non-synchronized watermark detections!
- D(z) provides an estimate of the reliability of the watermark detection
- Reliability = |peak value(s)| / σ_{noise}

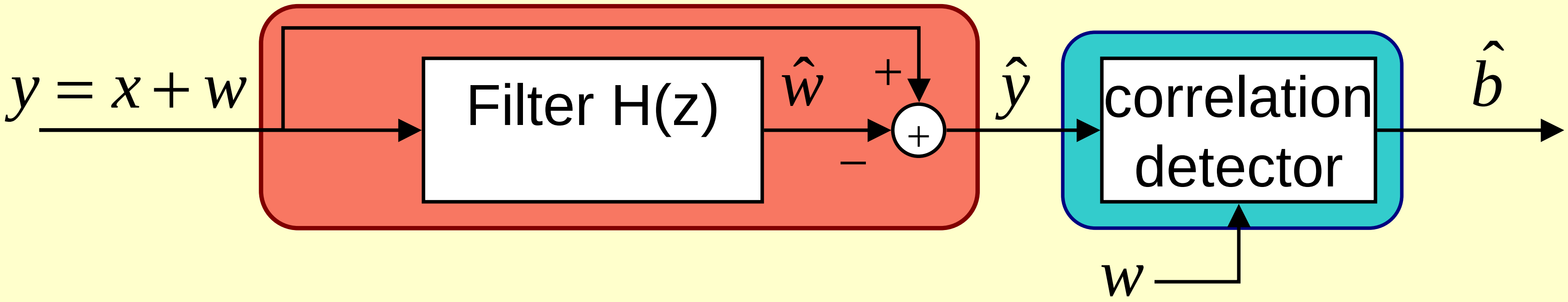


Translation Robustness



Take any 'natural' image of your liking and quantitively determine JPEG robustness of spread-spectrum watermarking using SPOMF detection; determine reliability of detection

Estimation and Removal



- Problem Statement: find watermark W[i] such that for given embedding distortion $N\sigma_W^2$ the detection reliability D and attack distortion D_a are maximized for any estimation filter H(z).

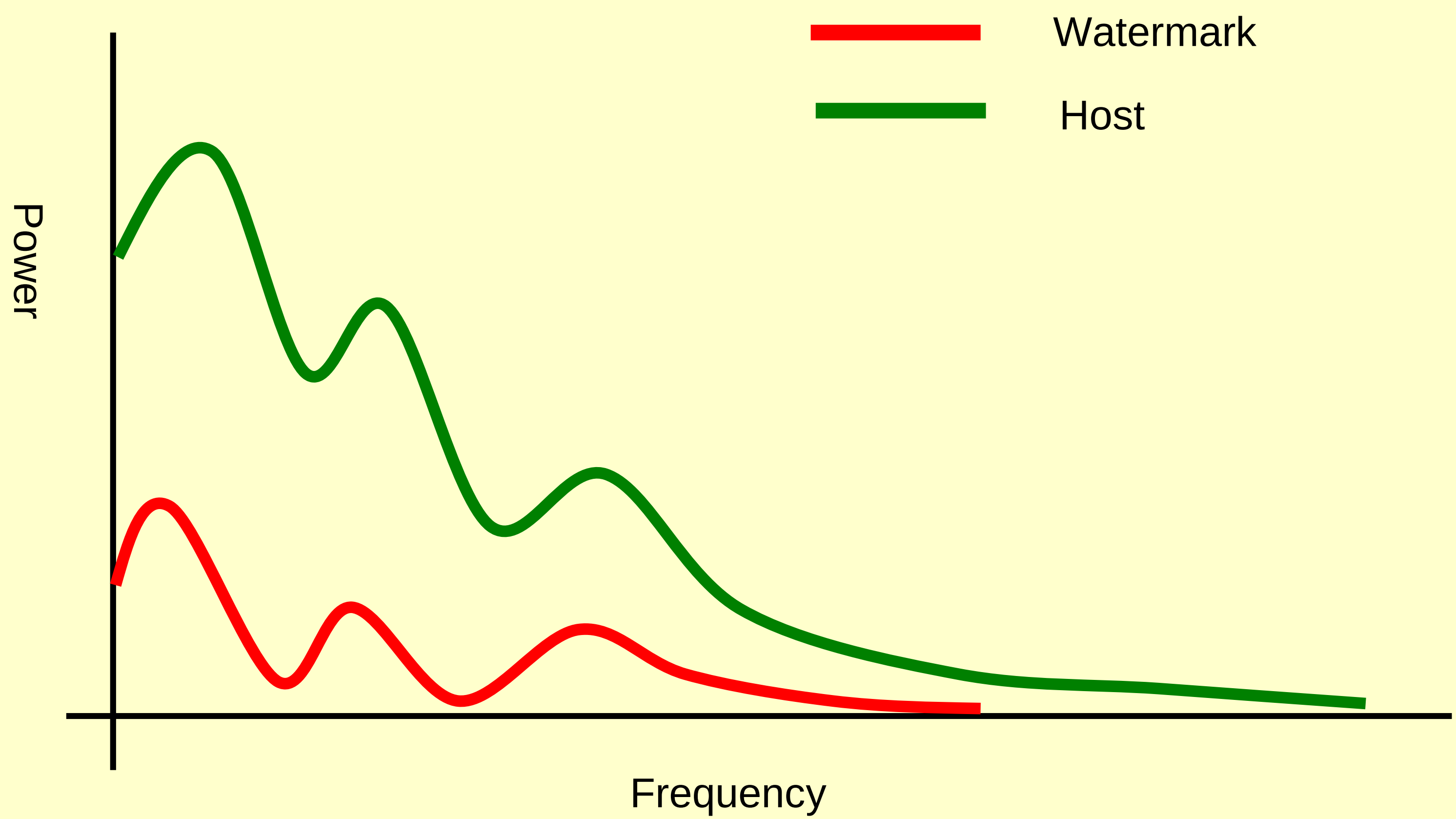
Estimation and Removal

- Problem description in frequency domain
  - $H(z) : \eta_i$
  - $W(z) : \omega_i$
  - $X(z) : \xi_i$
- Conditions
  - $\Sigma\, \omega_i^2 = N\sigma_W^2$
- Maximize
  - Attack distortion: $\Sigma\, (1- \eta_i)^2\, \xi_i^2 + \eta_i^2\, \omega_i^2$
  - Detection reliability: $(\Sigma\, \eta_i\, \omega_i^2)^2 / \Sigma\, \eta_i^2\, \xi_i^2$

Estimation and Removal

- From detection reliability (using Lagrangian multipliers)
  - $\eta_i = a\ \omega_i^2 / \xi_i^2$
- From attack distortion and condition (using Lagrangian multipliers)
  - $\eta_i = \eta = b\ \omega_i^2 / (\xi_i^2 + \omega_i^2)$
- Combining we find for all frequency components
  - $\rho = \omega_i^2 / \xi_i^2$ is fixed
- Power Spectral Condition (PSC) of [Su, Girod]
- Theoretical justification for heuristic arguments
  - Cox et al.

Power Spectral Condition



Estimation and Removal

- Optimal Watermark and Attack Filter
  - $\Phi_W = (\sigma_W^2 / \sigma_X^2)\, \Phi_X$
  - $H = \Phi_W / (\Phi_W + \Phi_X) = \sigma_W^2 / (\sigma_X^2 + \sigma_W^2)$ (scalar!)
- First example of game theory in watermarking
  - Embedder wants to maximize robustness
    - Tool: W(z), Cost: Embedder distortion
  - Attacker wants to minimize robustness
    - Tool: H(z), Cost: Attacker distortion

Estimation and Removal

BER

Take any ‘natural’ image of your liking and quantitively determine robustness of spread-spectrum watermarking under an estimation and removal attack.

JPEG Quality