Digital Watermarking

Ton Kalker

Hewlett-Packard Labs

Overview

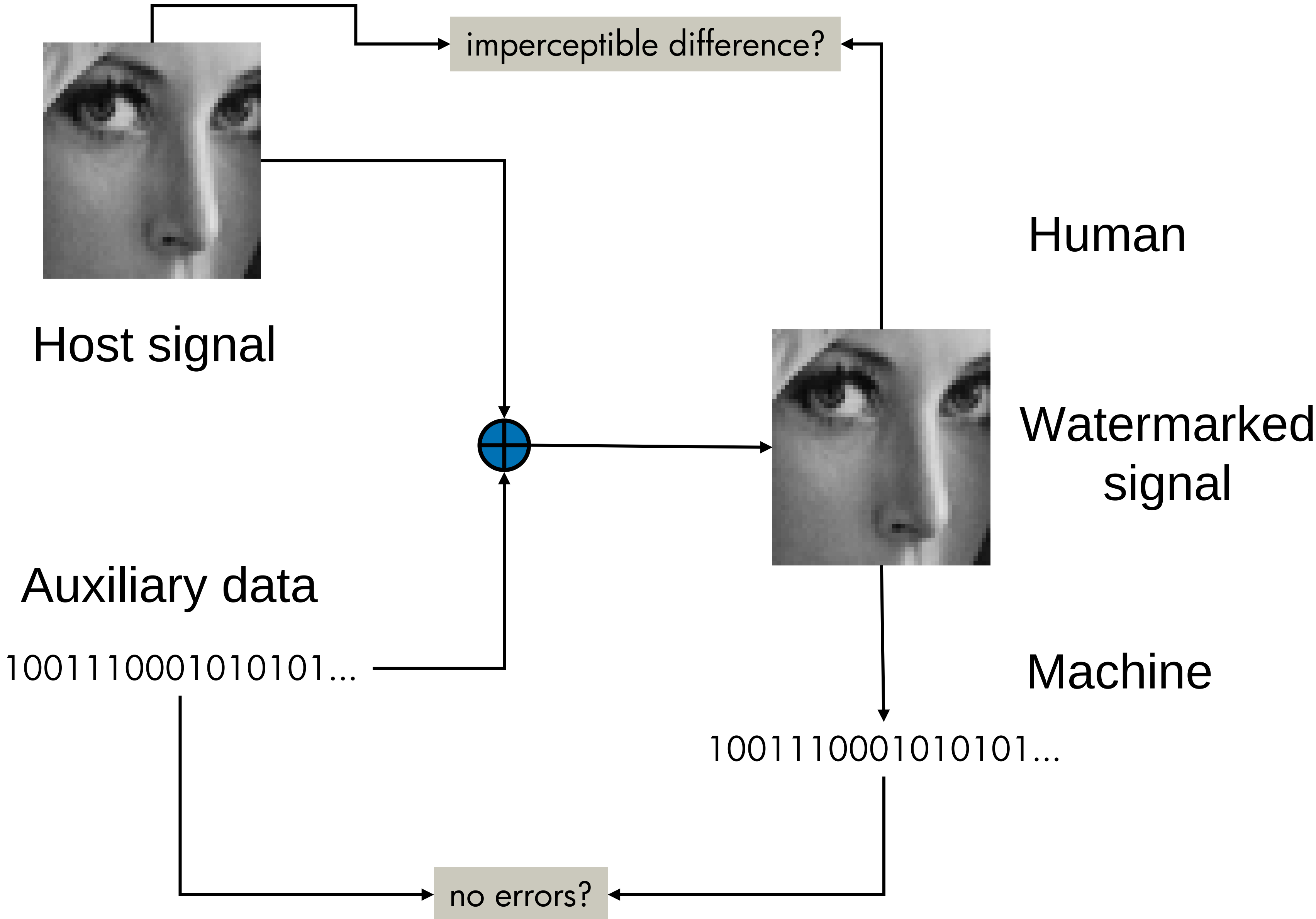
- Part I
  - classification of watermarking
  - basic examples
  - applications
- Part II
  - Spread-Spectrum watermarking
- Part III
  - Quantization Index Modulation
- Part IV
  - Costa’s Theorem

Part I

Introduction & Classification

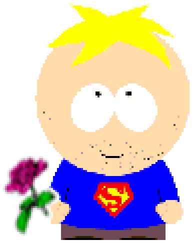
What is Digital Watermarking

- Original signal
  - host (cover)
    - audio, image, video, 3D model, …
- Auxiliary data
  - potentially related to host
- Multiplexed into one signal
  - Watermarked signal
- Two receivers
  - Humanoid receiver
    - signal detector
    - host signal
  - Mechanical receiver
    - watermark detector
    - auxiliary data


imperceptible difference?
Human
Host signal
Watermarked signal
Auxiliary data
1001110001010101...
Machine
1001110001010101...
no errors?

Players

- Simon (sender)
  - Access to host signal
  - Transmitting message embedded in host



- Robert (human receiver)
  - Access to watermarked signal
  - Access to machine for message reading

- Evan (human or not)
  - Man in the middle
  - Intentional and/or non-intentional interference
    - Intentional: attacker
    - Non-intentional: channel
  - Has no access to (shared) secrets by Simon and Robert

Signal Roles

- M : transmitted message
  - Simon embeds in
- C_o : host signal
  - Simon modifies to
- C_w : watermarked signal
  - Evan modifies to
- C_{nw} : degraded & watermarked signal
  - Robert restores to
- C_n : restored signal
- M_n : estimated message

Classification: steganography

- Steganography
  - Secret writing
- Context
  - Simon free to choose <u>any</u> host
- Goal
  - Communicate reliably a secret message to Robert
  - Hiding the presence of the message to Evan
- Note
  - Host distortion may potentially be large!

SimpleStego (Memon et al.)

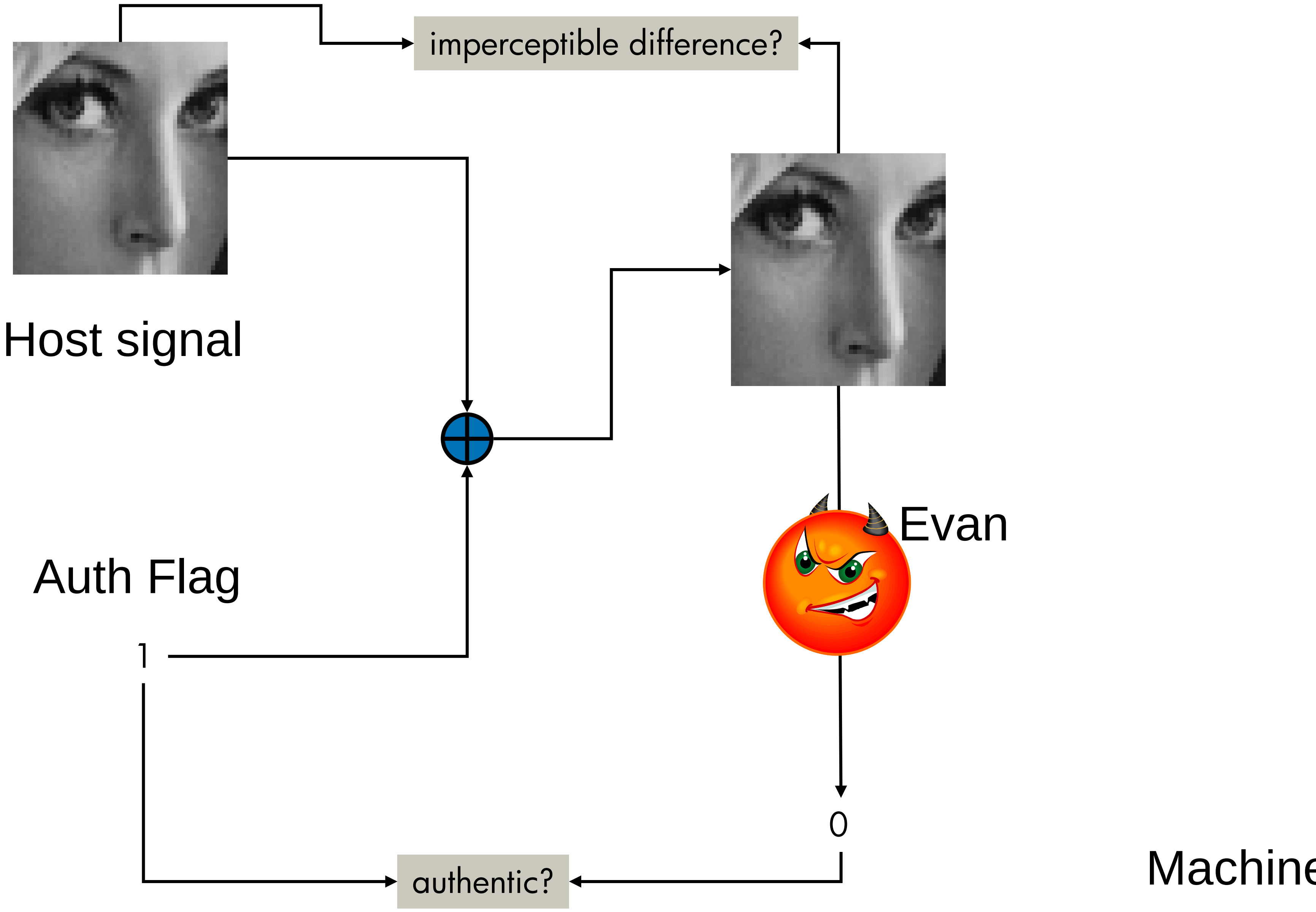
- Initialization
  - Simon and Robert agree upon a common cryptographic n-bit hash function $h = H(C)$
- Loop
  - Simon chooses an n-bit message M.
  - Simon shoots $O(2^n)$ pictures with his HP camera
  - After $O(2^n)$ pictures, Simon will have a picture C such that $H(P) = M$
  - Simon sends C
  - Robert retrieves M

SimpleStego (Memon et al.)

- Theorem
  - For SimpleStego, Evan cannot distinguish between an picture encoding a message or not
  - SimpleStego is secure
- Issues
  - SimpleStego is impractical
    - Complexity
- Steganography objective
  - Design practical secure stego methods
  - Design stego detection methods

Classification: Authentication watermarking

- Context
  - Simon is given a specified host signal
- Goal
  - Transmit authenticity flag
    - One message only
  - Any interference by Evan flips the flag
  - Robert can verify authenticity
- Note
  - Embedded digital signature


imperceptible difference?
Host signal
Auth Flag
1
Evan
0
authentic?
Machine

SimpleAuth

- Initialization
  - Simon and Robert agree upon a common and public cryptographic n-bit hash function h = H(C)
  - Simon and Robert agree upon a common secret n-bit message M.
  - Simon is given signal C
- Loop
  1. Simon randomly modifies C yielding Q ~ C
  2. If not H(Q) = M, go to (1).
  3. If H(Q) = M, transmit Q

SimpleAuth

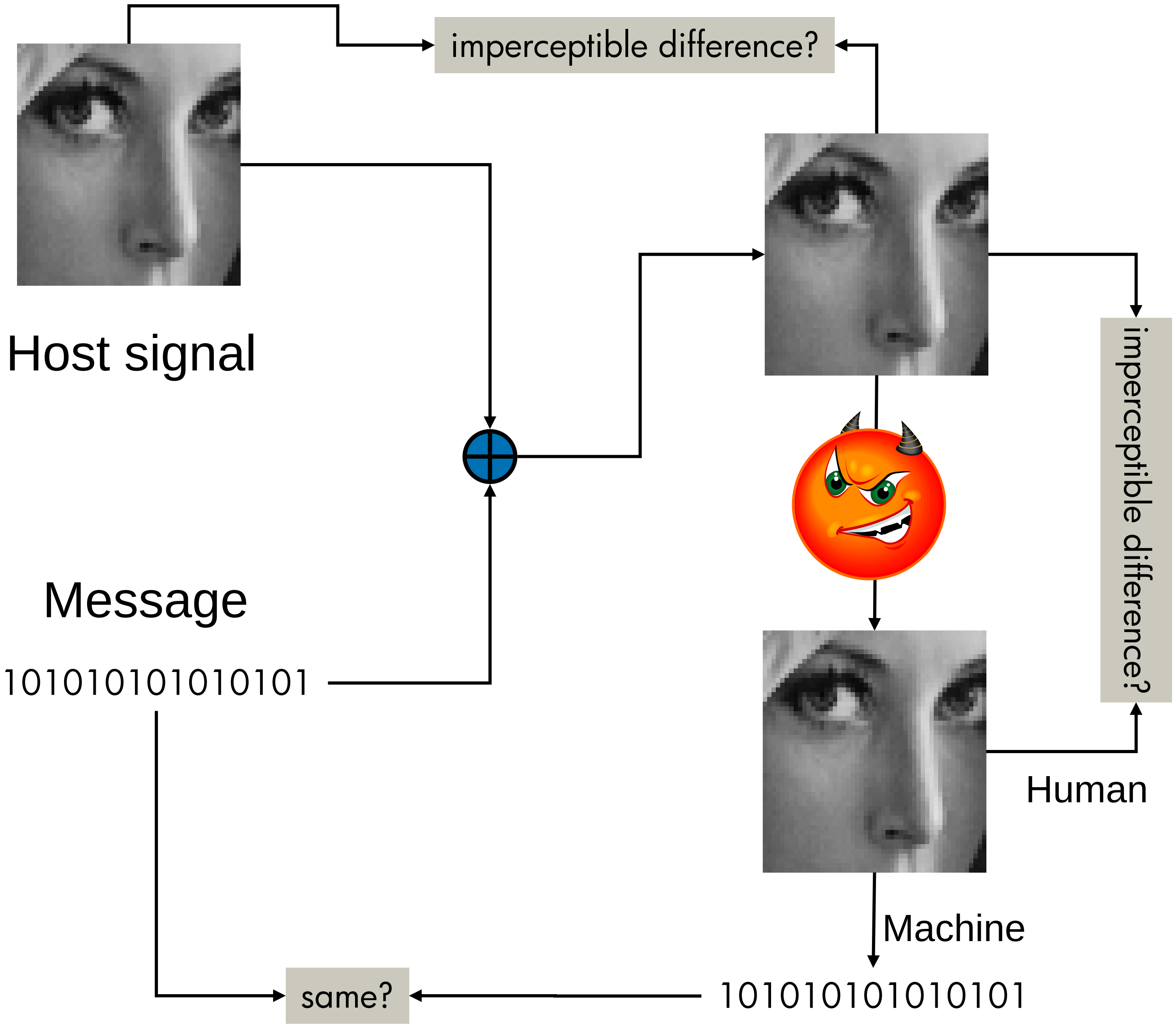
- Theorem
  - If n large enough, any modification of the transmitted signal Q by Evan will result in a flip of the authentication flag.
- Issues
  - SimpleAuth is impractical
    - Complexity of Simon and Robert is equal
- Authentication objectives
  - Design practical secure watermark authentication methods
  - Allow for localization of interference
  - Allow for benign modifications

Classification: Robust Watermarking

- Context
  - Simon is given a specified host signal

- Goal
  - Transmit a message M
  - Any restricted interference by Evan retains M
    - Typically a distortion constraint
  - Evan cannot read, modify or erase the message M
  - Robert can reliably read M

- Note
  - Distortion constraints are typically not well-modeled
  - In practical situations, Evan might resort to
    - Exploiting the weakness of perceptual models
    - Ignoring his imposed interference constraints


imperceptible difference?
Host signal
Message
101010101010101
imperceptible difference?
Human
Machine
101010101010101
same?

LSB Watermarking

- Initialization
  - Host signal P is an nxn image with 8-bit pixel values
  - Simon and Robert agree upon a secret pseudo-random common nxn bit array X.

- Transmission
  - Simon transmits the bit ‘b’ by replacing the LSB-plane of the image by ‘Y = b XOR X’
  - Embedding distortion: 0.5 bit/pixel

- Channel
  - Evan restricted to only replace 25% of the LSB values: Y $\rightarrow$ Z
  - Channel distortion: 0.25 bit/pixel

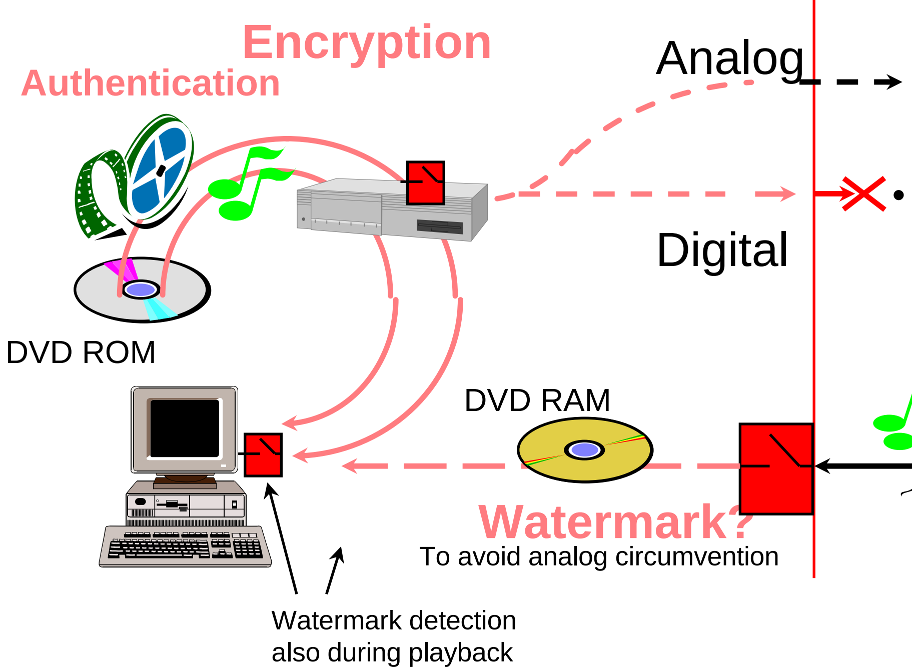
- Detection
  - Robert correlates LSB plane of Z with X
  - If n large, Robert will retrieve message bit b with high probability

LSB Watermarking

- If Evan obeys constraints
  - LSB watermarking robust
- However
  - Interference constraint not perceptually motivated
  - Evan is allowed less distortion than Simon
- Objectives
  - Robust watermarking with
    - Relevant distortion constraints
    - Provable security

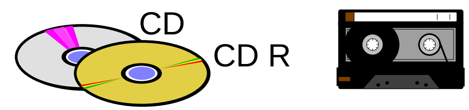
Compliant World

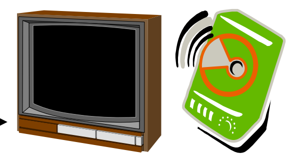
- All content is encrypted on all digital interfaces
- Link-by-link encryption; devices internally process clear content
- Controlled by CSS, 5C, 4C, ...
- Includes DVD players, DVD RAM, SDMI audio, DVD audio, PC’s



Non-Compliant World

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



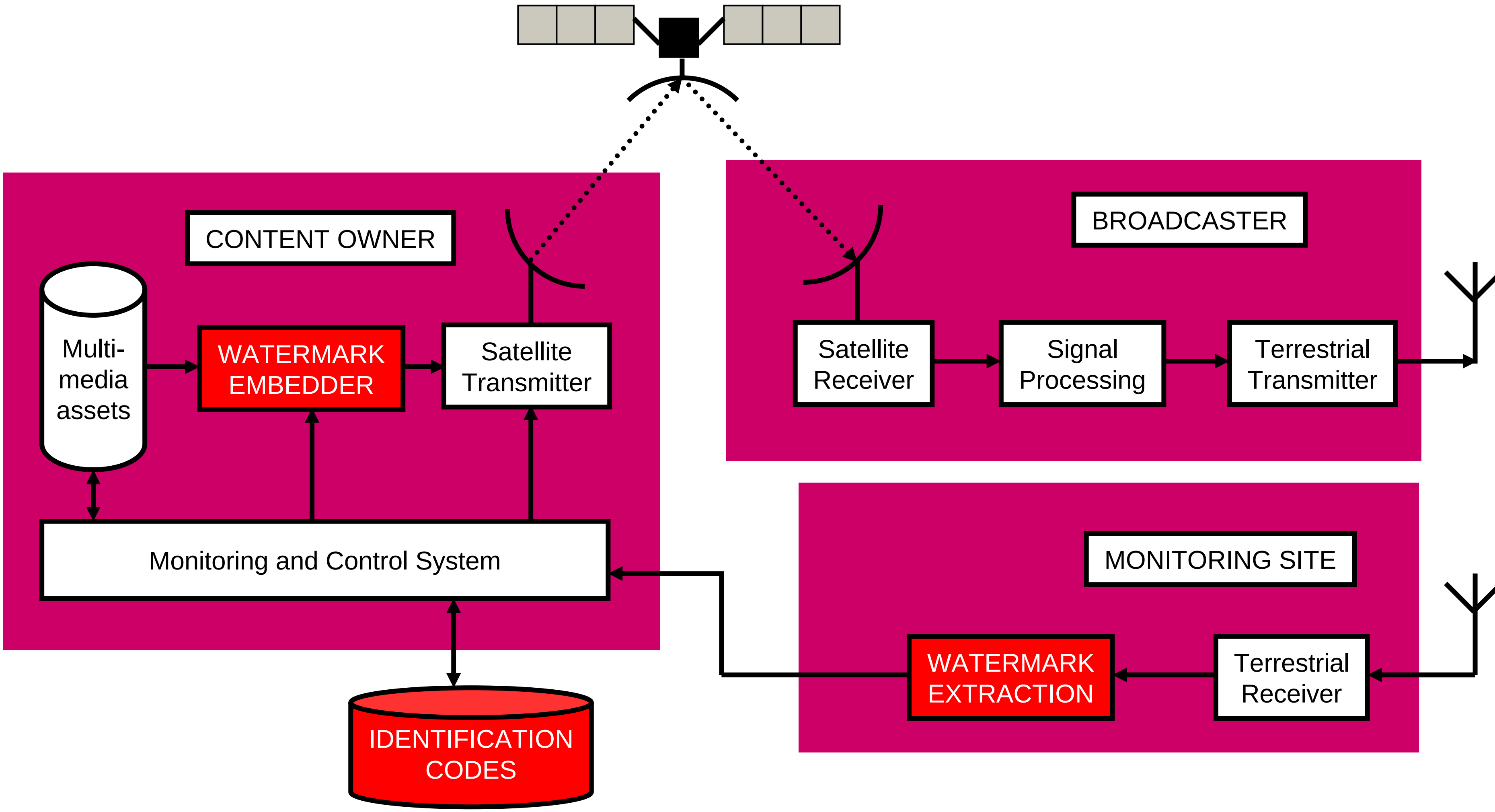


- Macrovision spoilers
- Watermarks

- By licensing contract no unprotected output

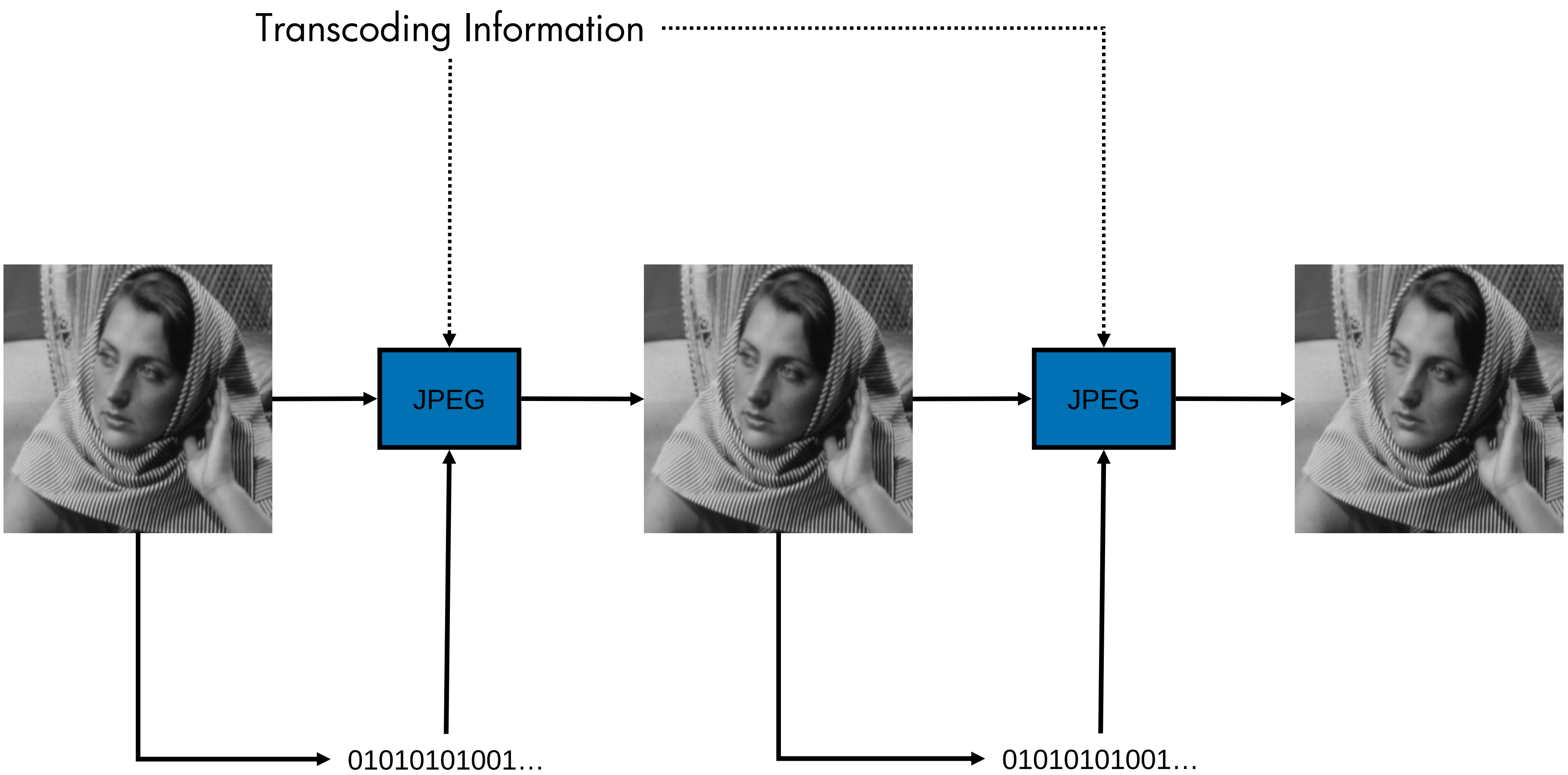
- New laws in US and EU

Broadcast Monitoring

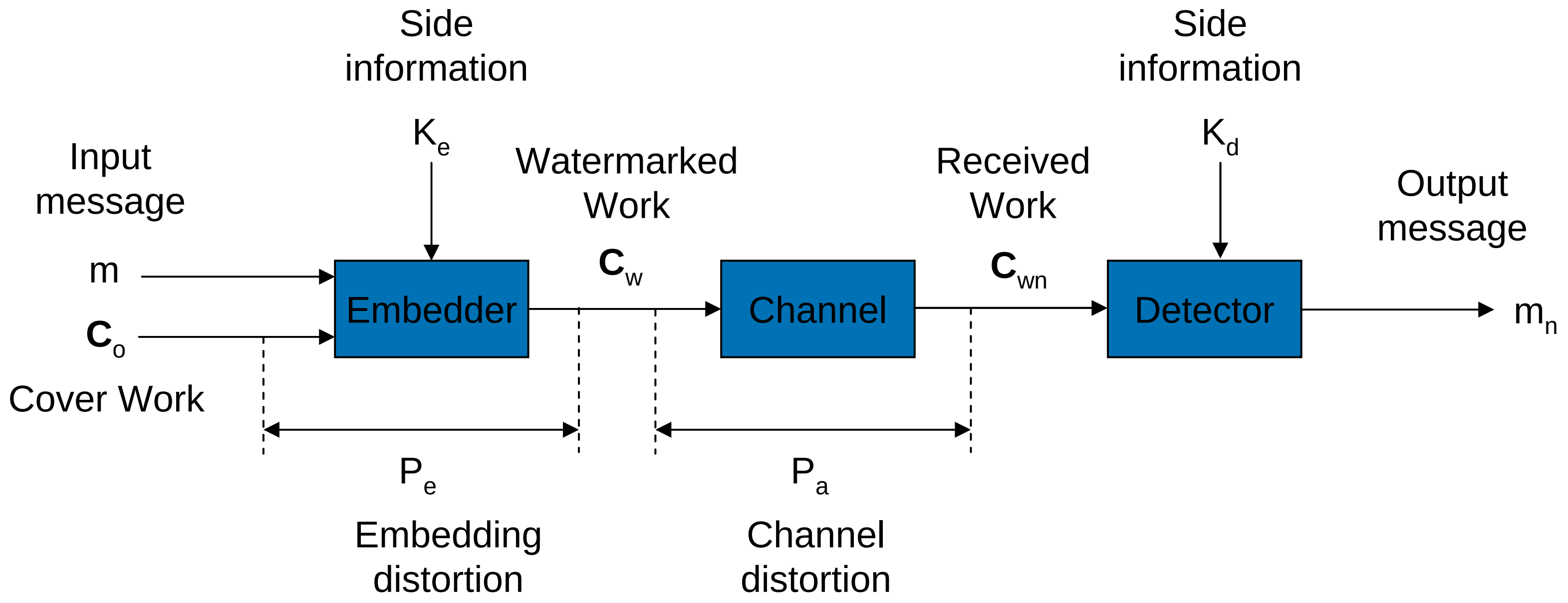


Name That Tune

Helper Data for Processing



Formal Model



- WNR = Watermark to Noise Ratio
  - Channel / Embedding
  - WNR large: high throughput
- WDR = Watermark to Document Ratio
  - Embedding / Host
  - WDR large: high througput

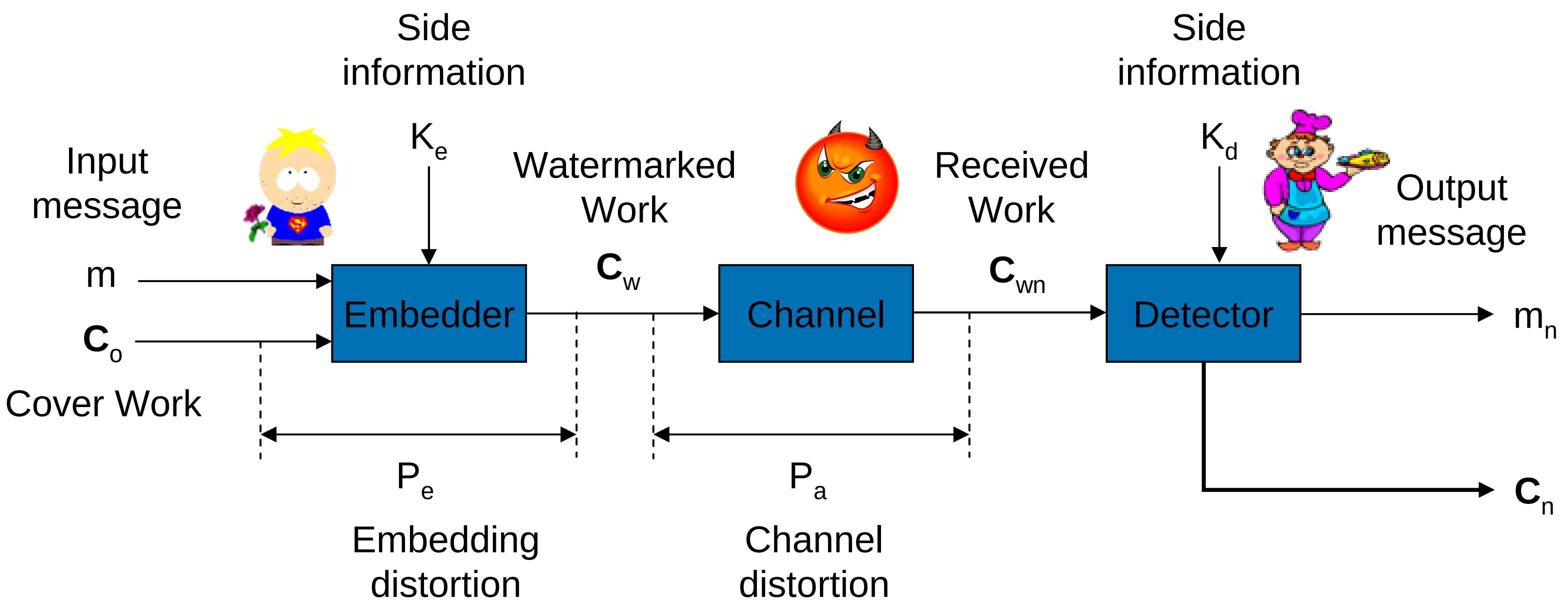
<u>Basic questions</u>

- What is the maximal rate of reliable communication?
- What is the coding scheme to achieve maximal rate?

Classification: Reversible Watermarking

- Context
  - A given host signal C_o and a message M
- Goal
  - Transmitting M embedded in C_o
  - Retrieving M from received signal C_{nw}
  - Restoring C_o from received signal C_{nw}
- Note
  - In most reversible scenarios Evan is absent
  - Theory in the case of presence of Evan is not completely understood

Formal Reversible Model



SimpleRev

- Initialization
  - C is iid B(r) source sequence of length n
    - $C = \{c_1, c_2, \ldots, c_n\}$, all c_i independent
    - $\text{Prob}(c_i = 1) = r$, $0 < r < 1$
  - Hamming distance
  - Evan absent

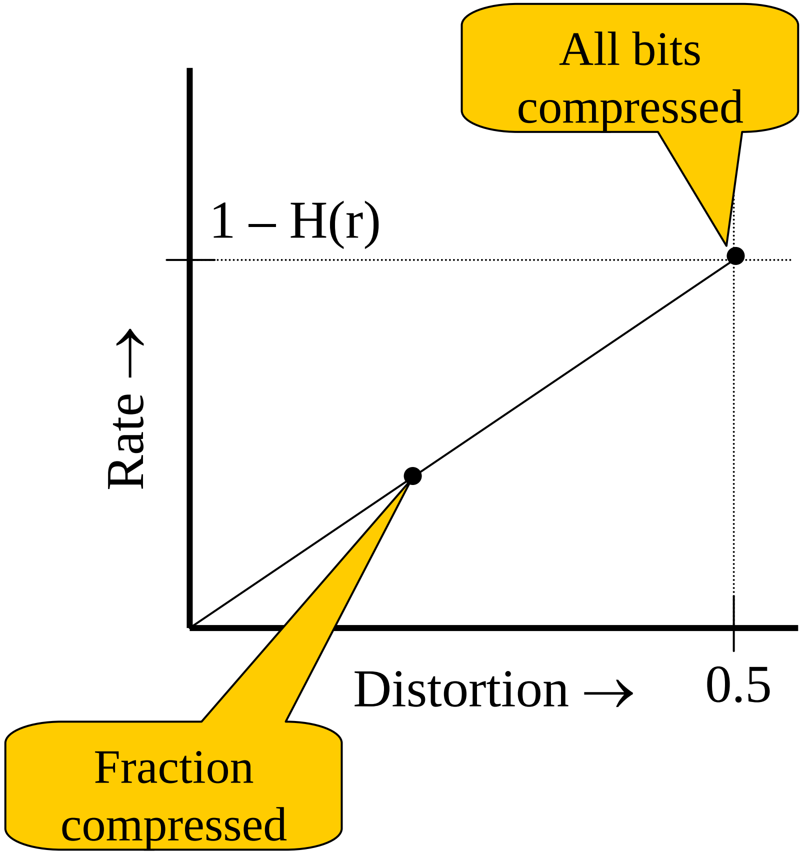
- Procedure
  - Compress C, say using Huffman encoding: >C<
  - |>C<| ~ n H(r)
  - $H(r) = -r \log(r) - (1-r) \log(1-r)$: binary entropy
  - Add n (1 –H(r)) random message bits

- Reversing
  - Strip message bits
  - Decompress

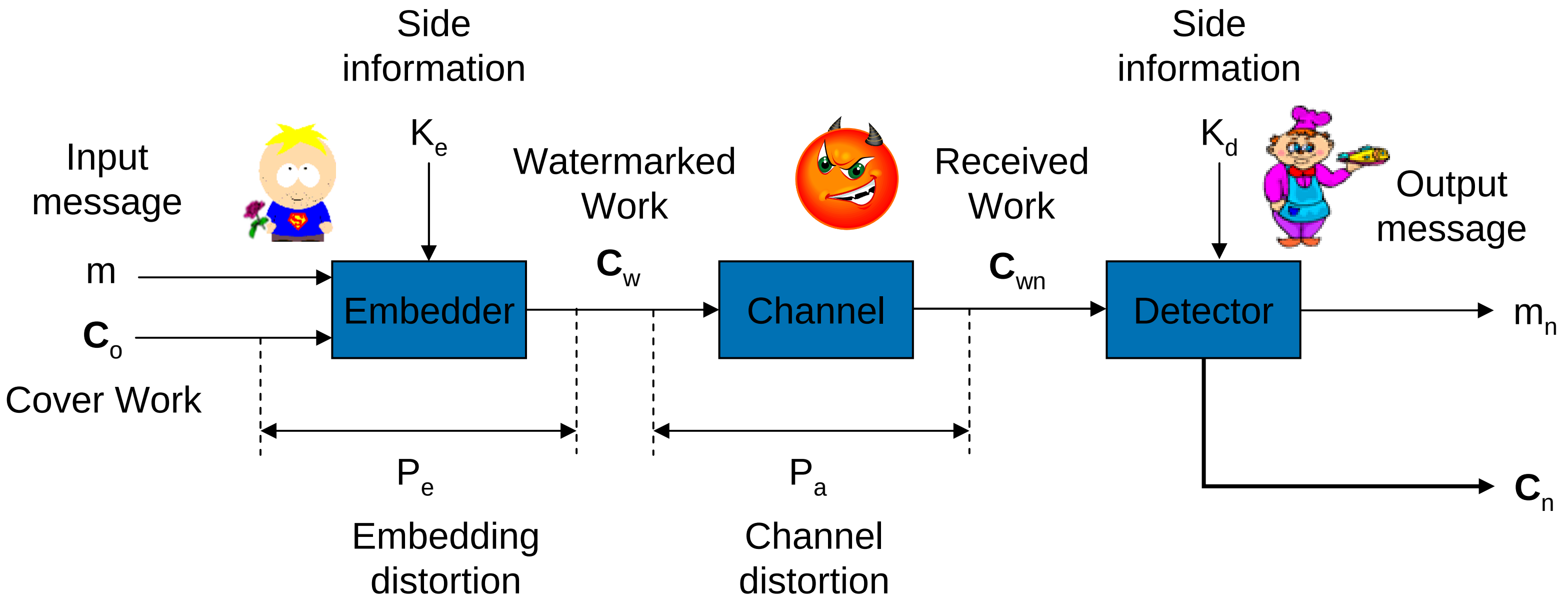
SimpleRev

- Resulting parameters
  - Distortion: **D = 0.5** bit per sample
  - Rate: **R = 1- H(r)** bit per sample
- Generalization
  - Apply previous procedure only for a fraction α of the bits in P**.**
- Resulting parameters
  - Distortion: **D = 0.5** α bit per sample
  - Rate: **R = (1- H(r))** α bit per sample
- **R(D)** relation (time-sharing)

R = 2 (1 – H(r)) D



Formal Reversible Model

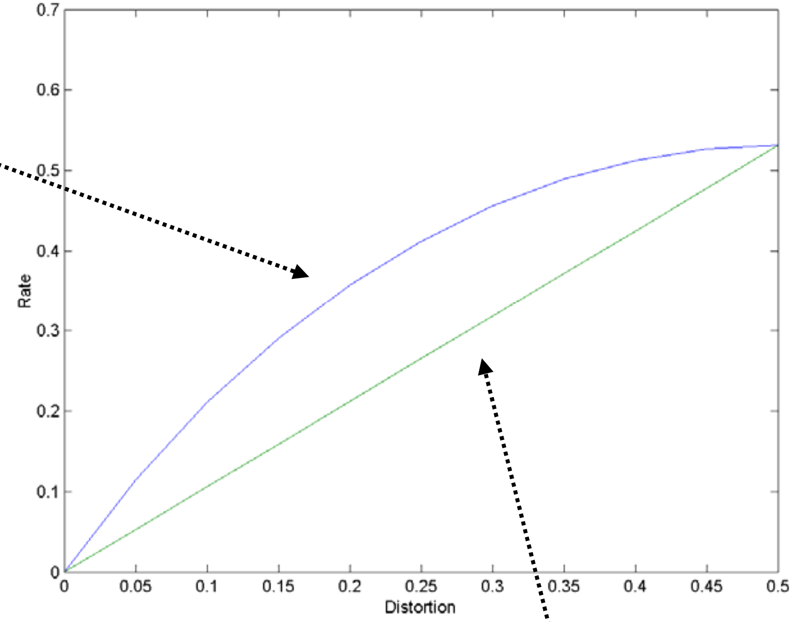


<u>Basic questions</u>

- What is the maximal rate of reliable communication?
- What is the coding scheme to achieve maximal rate?
- Is the previous scheme optimal?

Optimal Reversible Watermarking

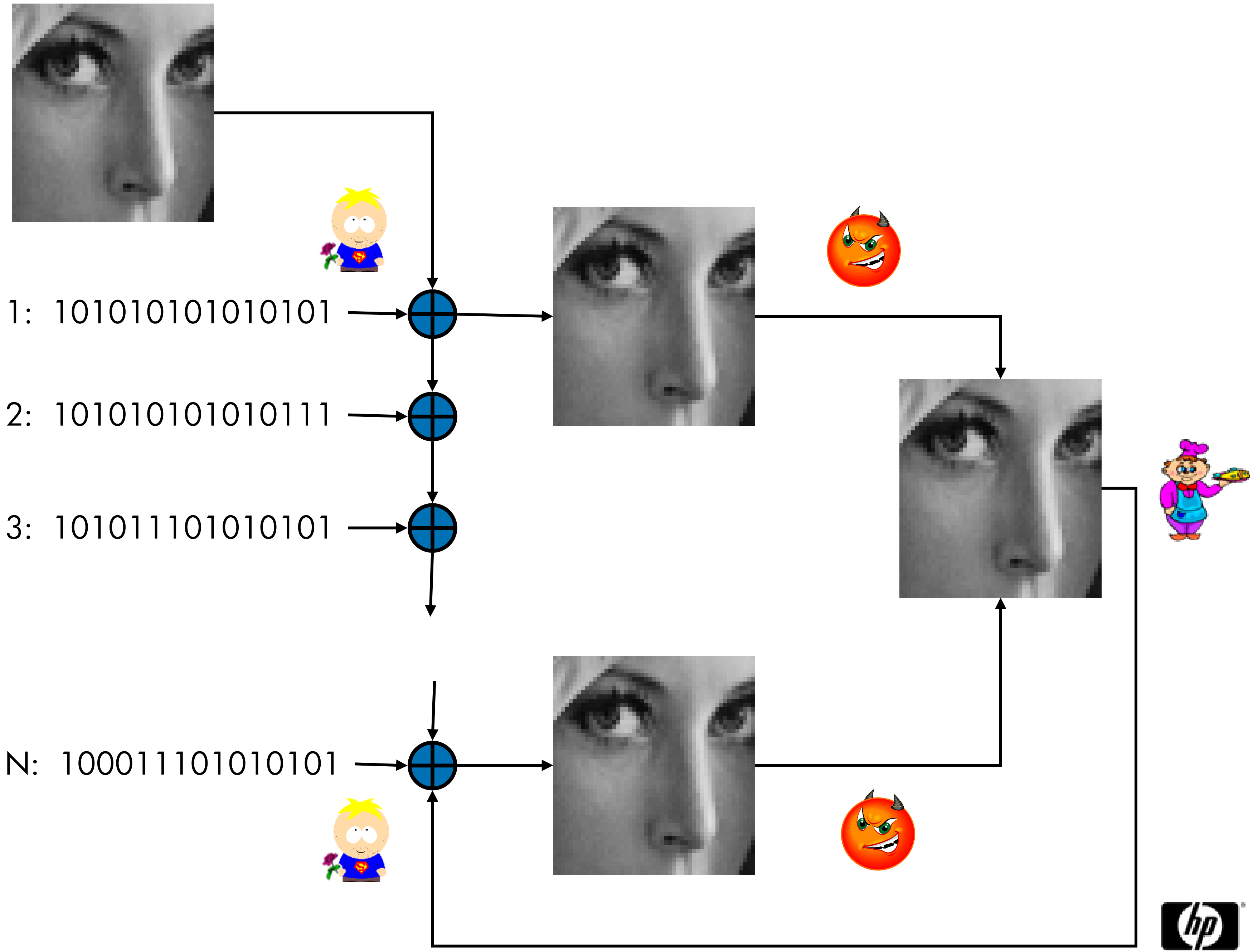
$$R(D) = H(r + (1 - 2r) D) - H(r)$$



$$R = 2 (1 - H(r)) D$$

Classification: Fingerprinting

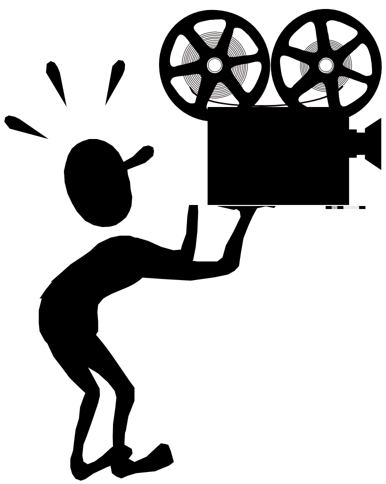
- Context
  - A group of N users
  - A unknown group S of k colluders (multiple Evans)
  - A single host signal C_o
- Goal
  - Embedding a message m_i in C_o for each user I
  - Retrieving at least on identity I in S from a colluded version $[[C_S]]$
  - where [[.]] is some averaging operator
- Note
  - some applications require the retrieval of all of S


1: 101010101010101
2: 101010101010111
3: 101011101010101
N: 100011101010101

Fingerprinting Application

- Alternative to Digital Rights Management (DRM)
  - DRM = pro-active protection of content
  - active enforcement of allowed usage rules
    - FairPlay (iTunes), MS-DRM (Napster), OMA-DRM (Cingular), Helix (Real), …
  - non-interoperable walled gardens
- Fingerprinting
  - retro-active enforcement of usage rules
  - content labeled with user identity
  - unauthorized distribution is traceable
    - even after collusion!

Digital Cinema



Watermark Parameters

- Perceptibility
  - perceptibility of the watermark in the intended application

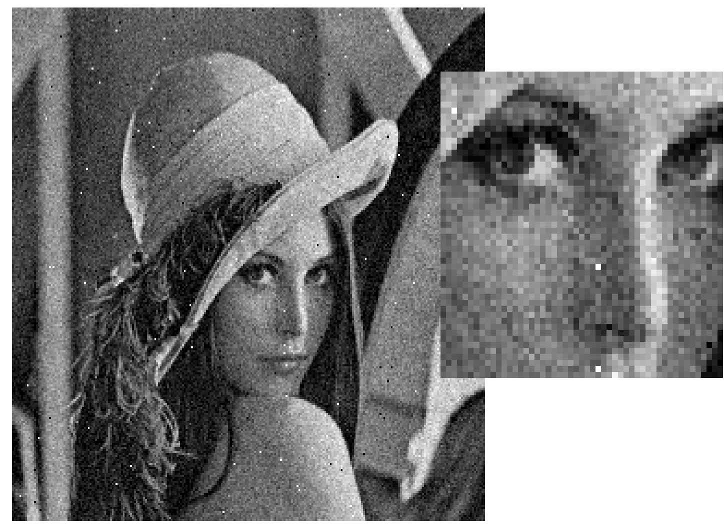
Original image

Image + hidden information

Watermark Parameters

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

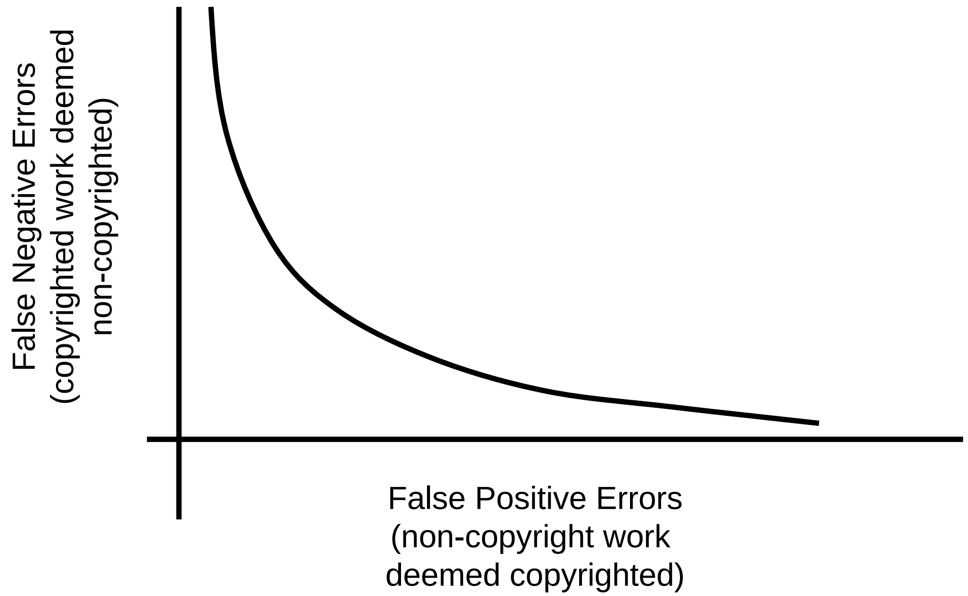
JPEG compression



Additive noise & clipping

Watermark Parameters

- Error Rates
  - example: copyright detection



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

Part II

Spread-Spectrum Watermarking

Patchwork

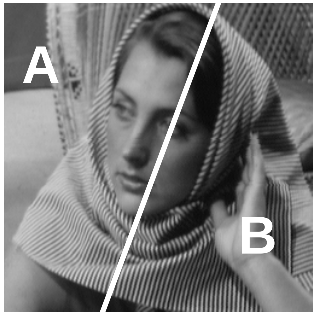
- 2 disjoint sets, A and B, of $N/2$ pixels each
  - pixels in each set (“patch”) chosen randomly
  - assumption:

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

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

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

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



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]

hp invent

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

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

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

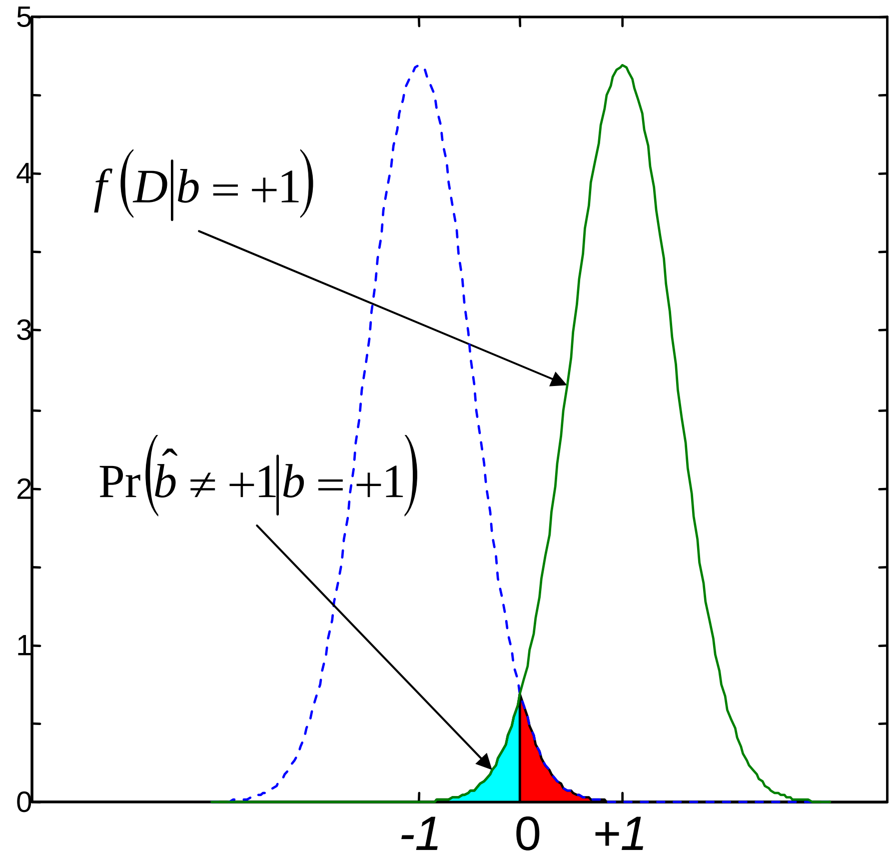
Detection (effectiveness)

- 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

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



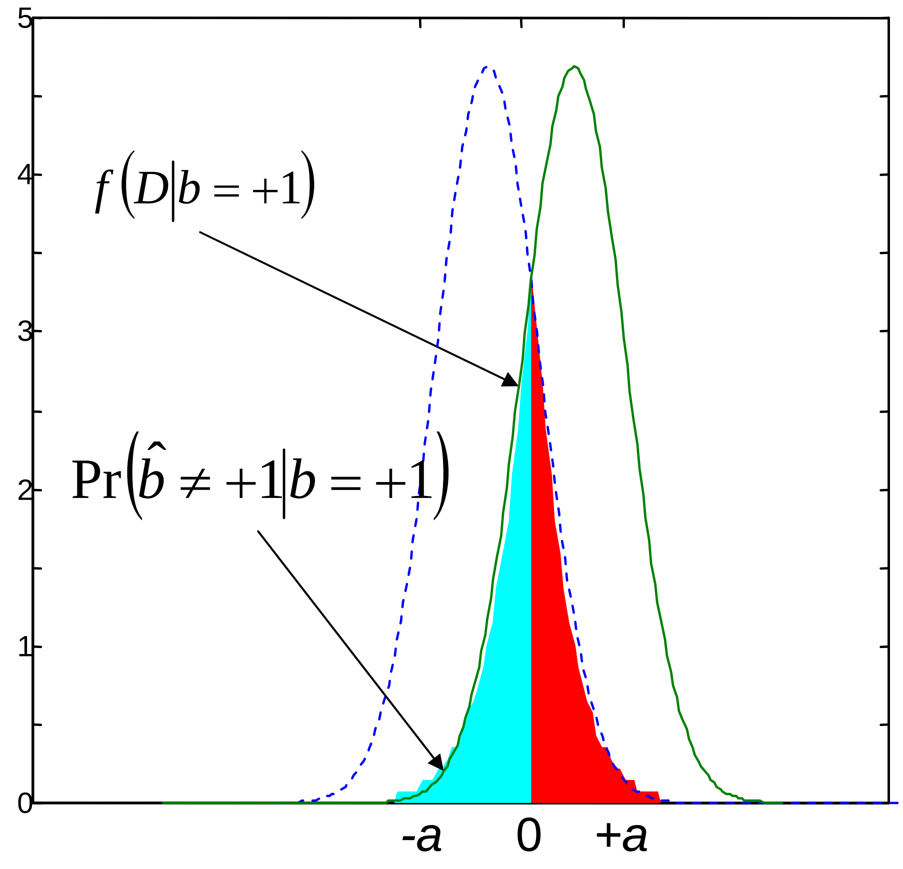
Detection (robustness)

- 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

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



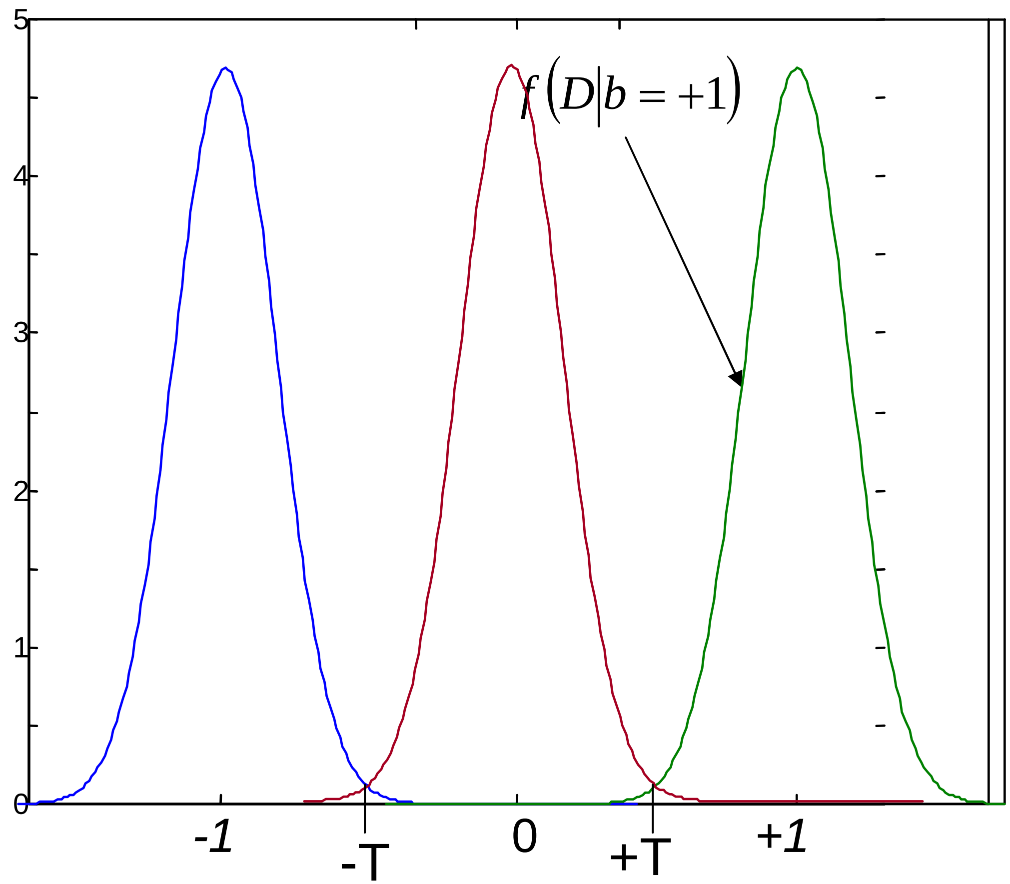
Detection (false positives)

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

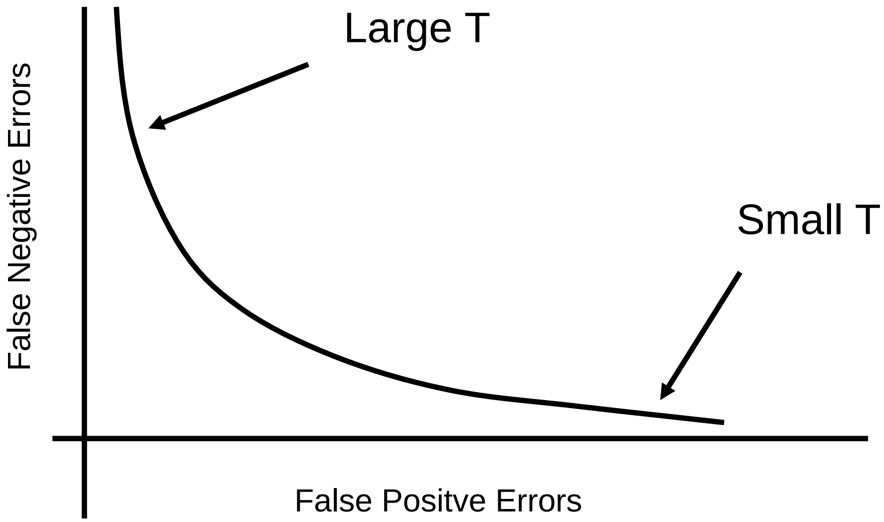
$$\hat{b} = \begin{cases} +1, & \text{if } D > +T; \\ -1, & \text{if } D < -T; \\ 0, & \text{if } |D| \le T. \end{cases}$$

- Probability of false positive

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

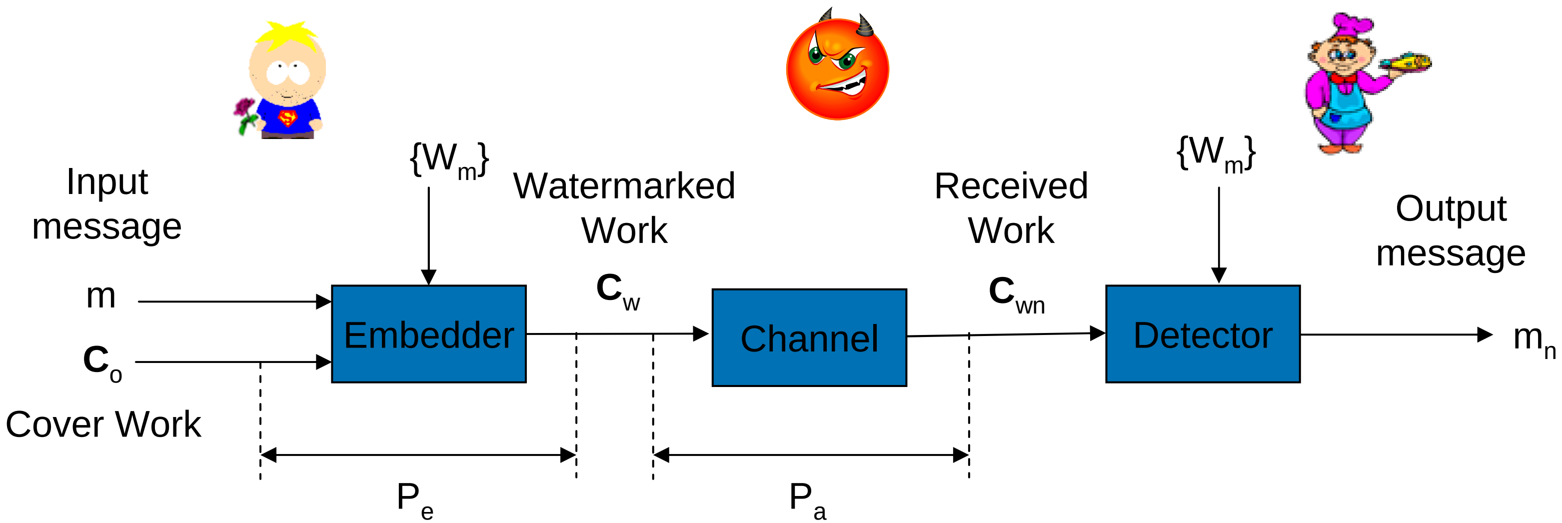


Error Rates



Transmitting n-bit messages

- Initialization
  - for each message $m \in \{0, \ldots, 2^n\}$ select a watermark sequence W_m
  - Simon and Robert share the code book $\{W_m\}$
- Loop
  - Simon chooses message m
  - Simon adds W_m to host C_o
  - Robert correlates C_{nw} with every element in code book
  - Robert declares the message m' such that $W_{m'}$ has the largest correlation with C_{nw}



Input message
m
C_o
Cover Work
$\{W_m\}$
Embedder
Watermarked Work
C_w
Channel
Received Work
C_{wn}
$\{W_m\}$
Detector
Output message
m_n
P_e
P_a

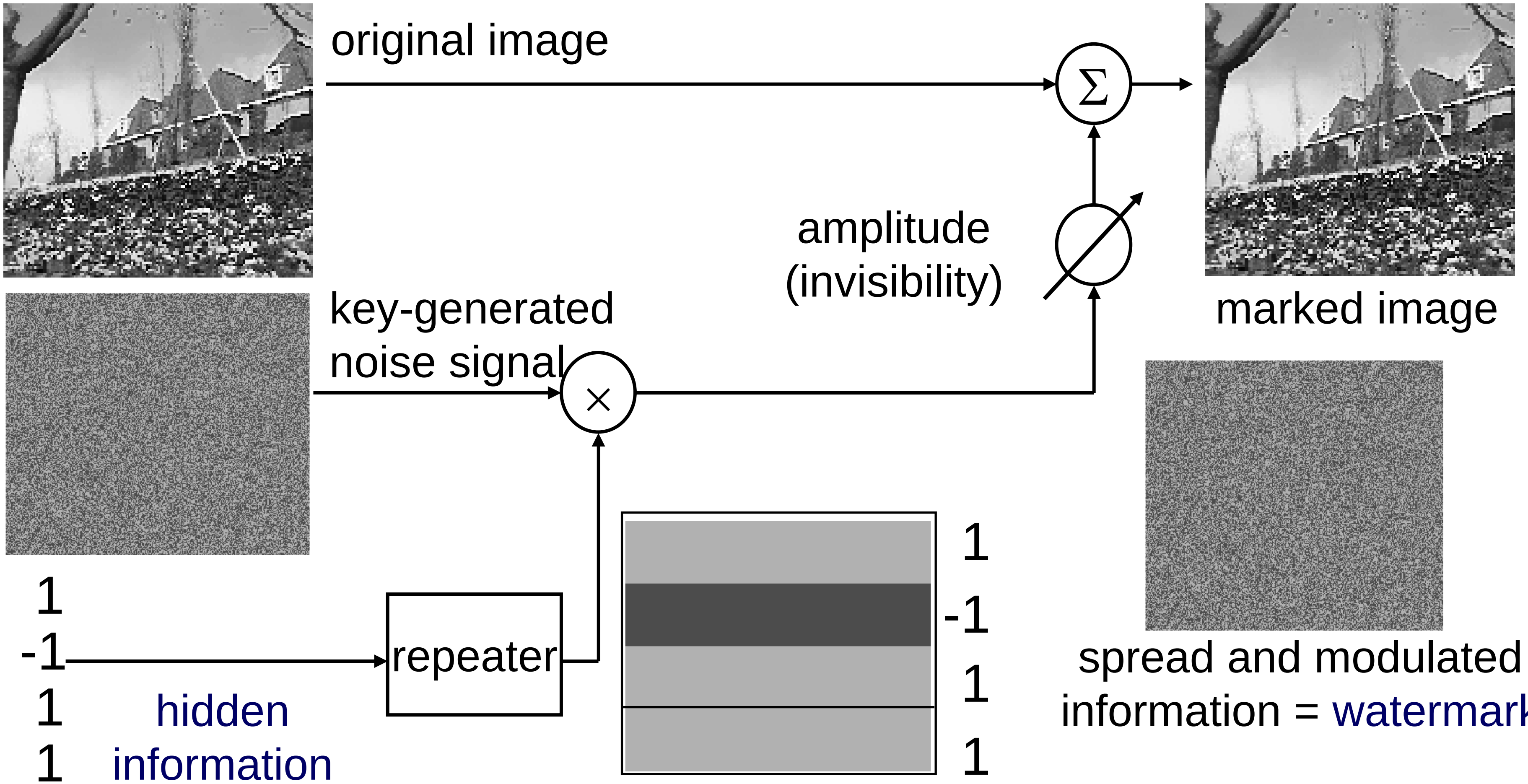
Practical Spread-Spectrum

- Message M is represented as n-bit structure

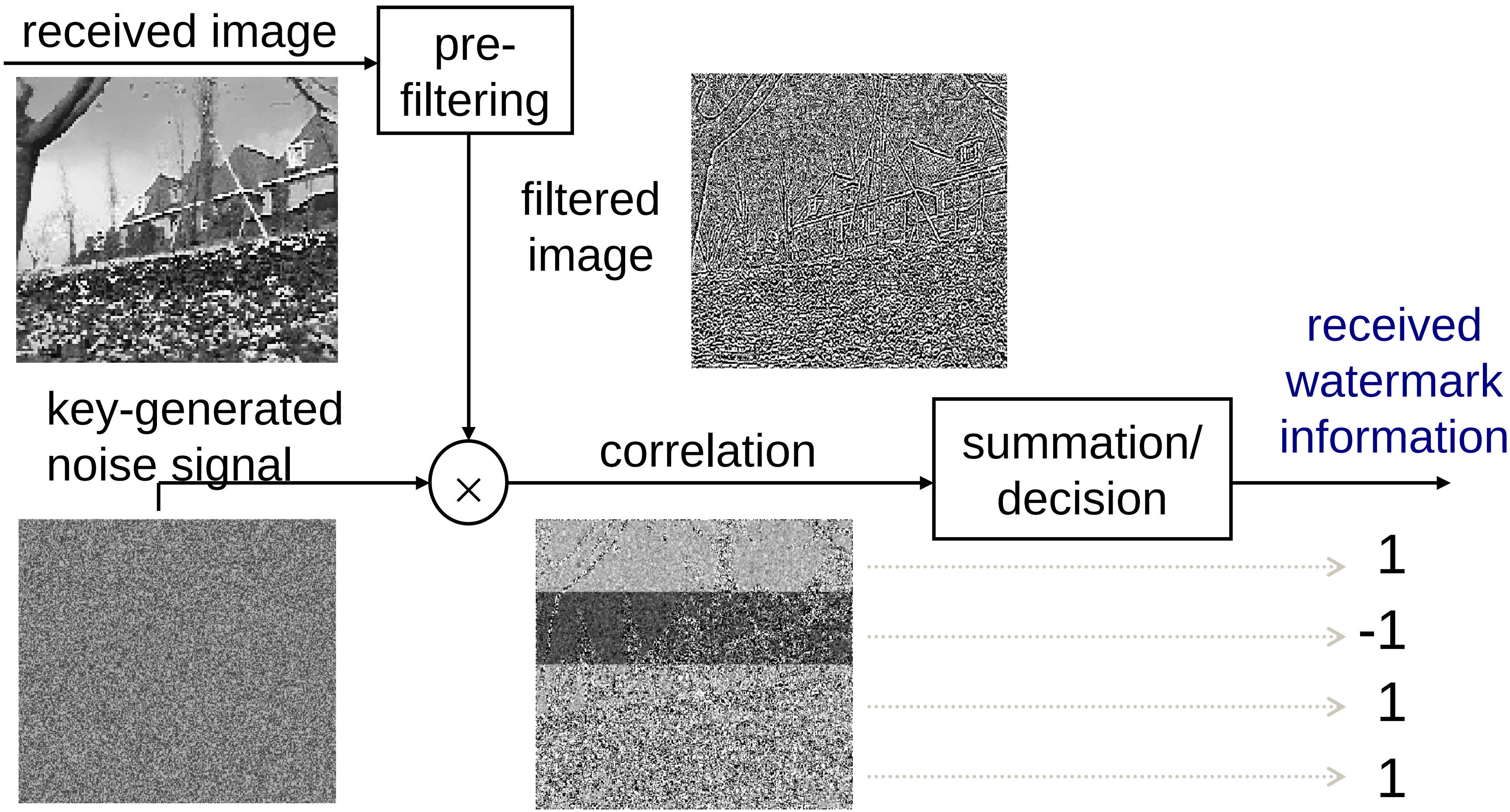
- Each bit is associated with anti-podal pair of watermark sequences
  - $Y = X + W$
  - $Y = X - W$

- M is transmitted and received bit by bit

Watermark Embedding



Watermark Retrieval

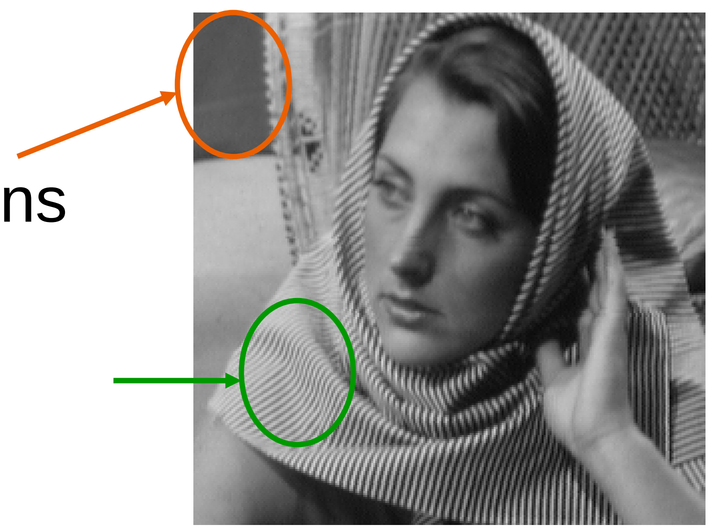


Perceptual Watermarking

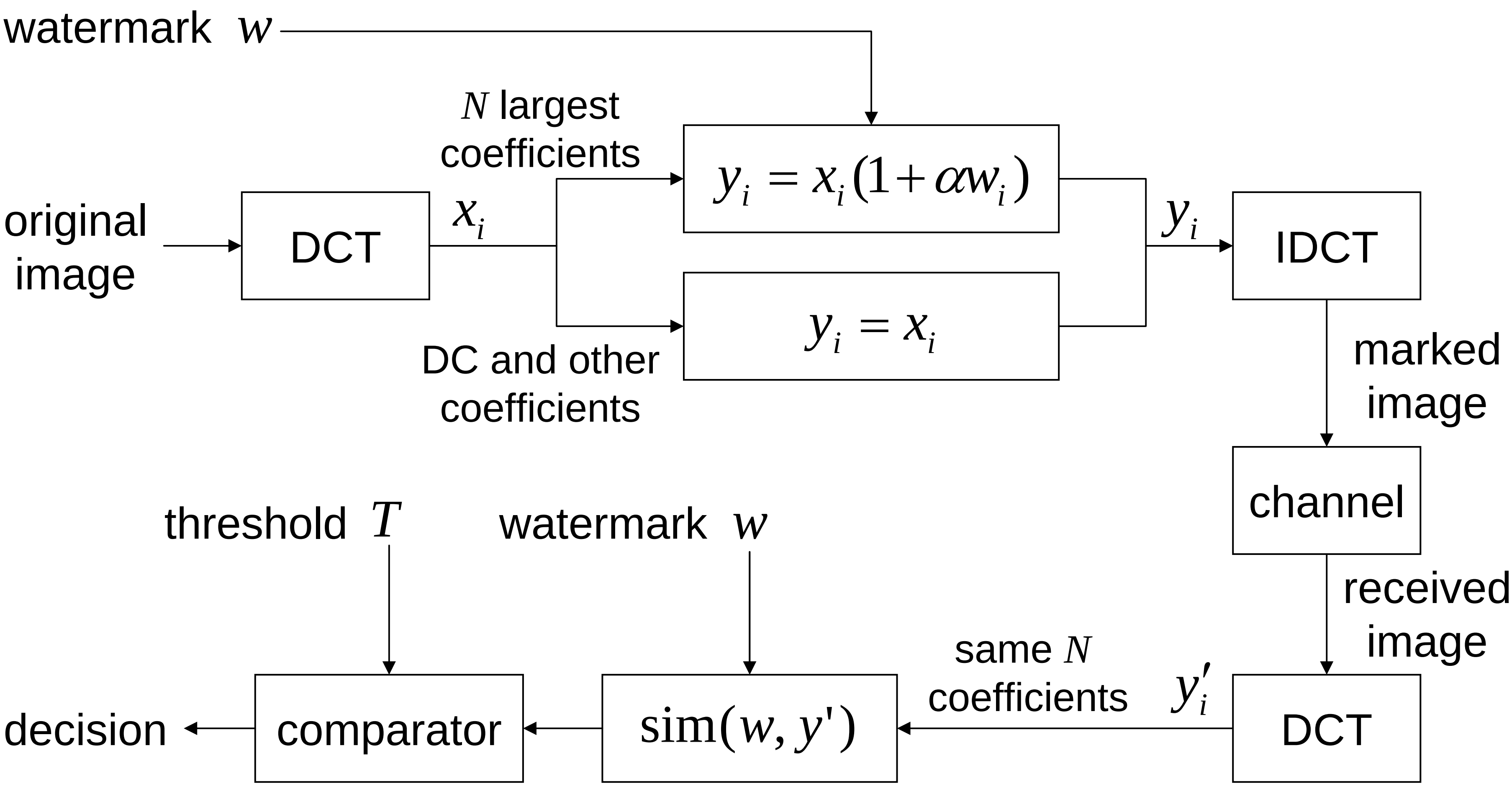
- Original x.
- Apply transform T: $y = T(x)$
  - T = I, DCT, FFT, log, … (or any combination thereof)
- Add pseudo-random sequence w: $z = y + w$
  - Allow adaptation of w to host signal
    - $\mathbf{Z} = \mathbf{Y} + \alpha \mathbf{W}$
  - In position
    - only in textured image regions, not in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

Perceptual Watermarking

- $T = I$
  - 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



Cox Image Watermarking Scheme



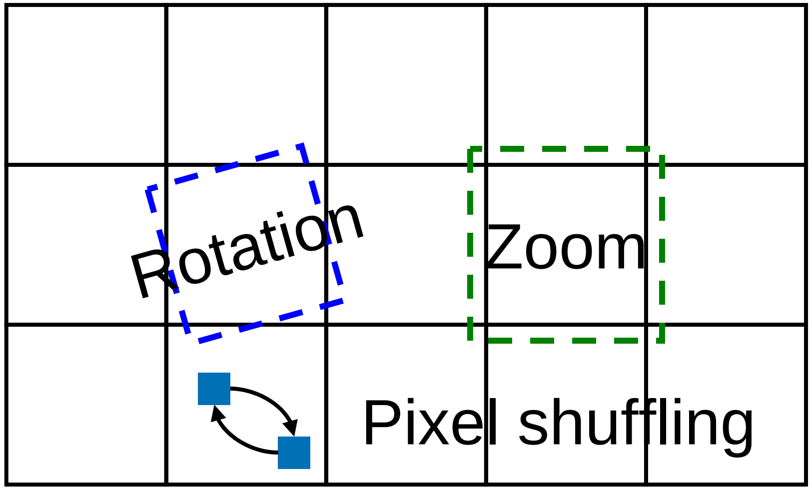
invent

Evan's options

- 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
- 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

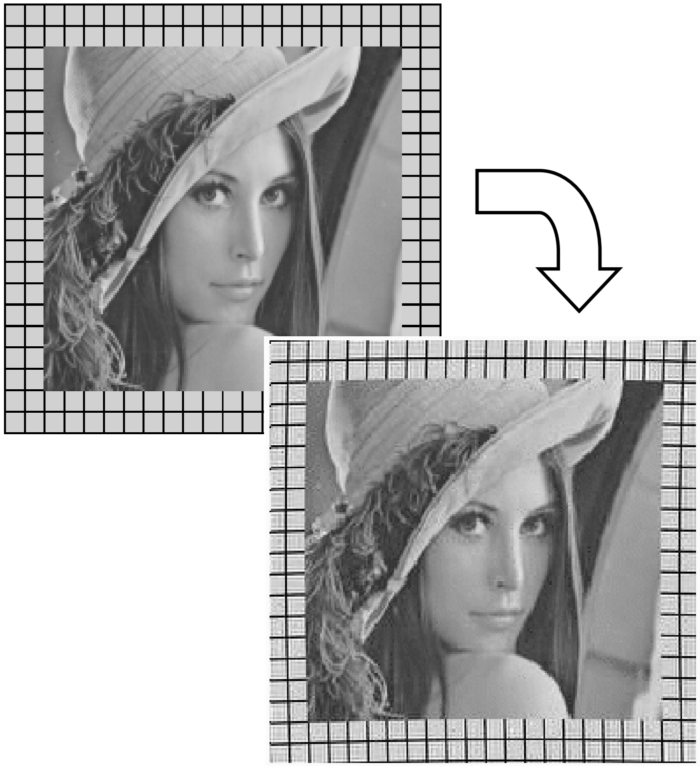
De-synchronization

- 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

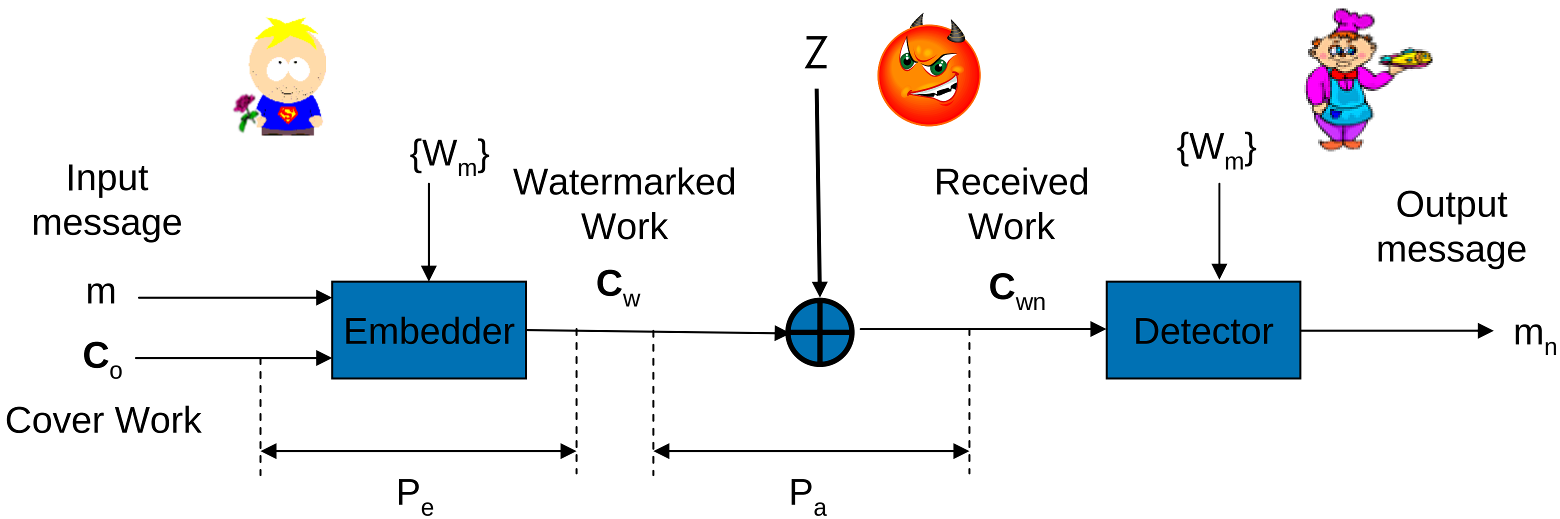


Optimal Rate Question

- Given a some statistical constraints on
  - the host C_o
    - model and energy
  - the embedding distortion P_e
    - type and power
  - the channel distortion P_a
    - type and power
- and allowing for arbitrary long signals,
- what is the maximal <u>rate</u> (number of messages per sample) that can be achieved?

Maximal Transmission Rate

- Assumptions
  - C_o is a white Gaussian signal of power P_o
  - The embedding power is restricted to P_e
  - Evan implements an Additive White Gaussian Noise (AWGN) channel of Power P_a



Spread-Spectrum Bound

- Observation
  - host signal and channel are AWGN to the watermark signal W_m
- Shannon's Theorem applies

$$R = \frac{1}{2}\log(1 + \frac{P_e}{P_o + P_a})$$

- For small WDR and modest WNR

$$R = \frac{1}{2}\log(1 + \frac{P_e}{P_o})$$

  - Host interference dominates

Performance regions

- WDR small

$$R = \frac{1}{2}\log(1+\frac{P_e}{P_o}) \approx \frac{1}{2P_o}P_e$$

  – rate grows linear with embedding power

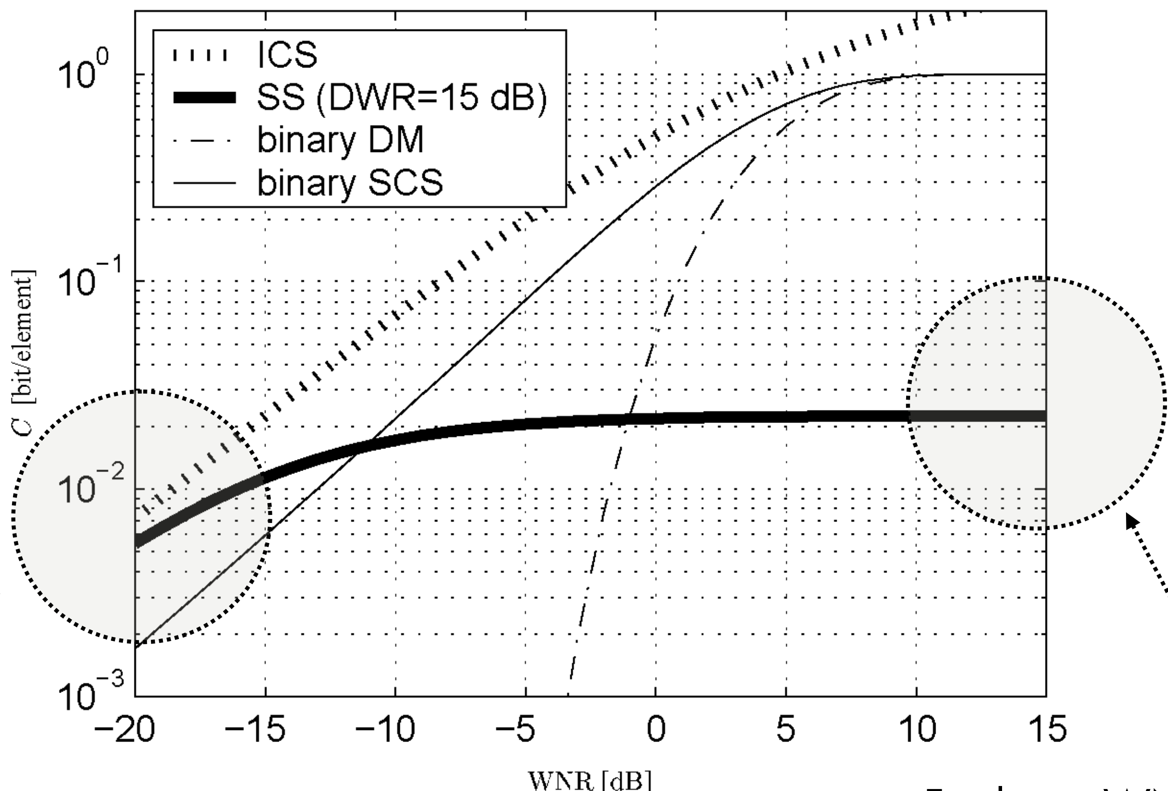
- WDR large

$$R = \frac{1}{2}\log(1+\frac{P_e}{P_o}) \approx \frac{1}{2}\log(\frac{P_e}{P_o}) = c + \frac{1}{2}\log(P_e)$$

  – grows logarithmic with embedding power

Performance graph

Eggers, Girod ©



For low WNR Spread-Spectrum approaches rate of optimal scheme ICS

For large WNR Spread-Spectrum underperforms with respect to the ICS scheme.