SCIENCES

Image, Field Director – Laure Blanc-Feraud

Compression, Coding and Protection of Images and Videos,
Subject Head – Christine Guillemot

Multimedia Security 1

Authentication and Data Hiding

Coordinated by
William Puech

Color section

Figure 1.1. *An example showing how an image has been modified several times in a row, each person who had lost favor seeing their image removed from the photo. Only Joseph Stalin appears in all four photos*

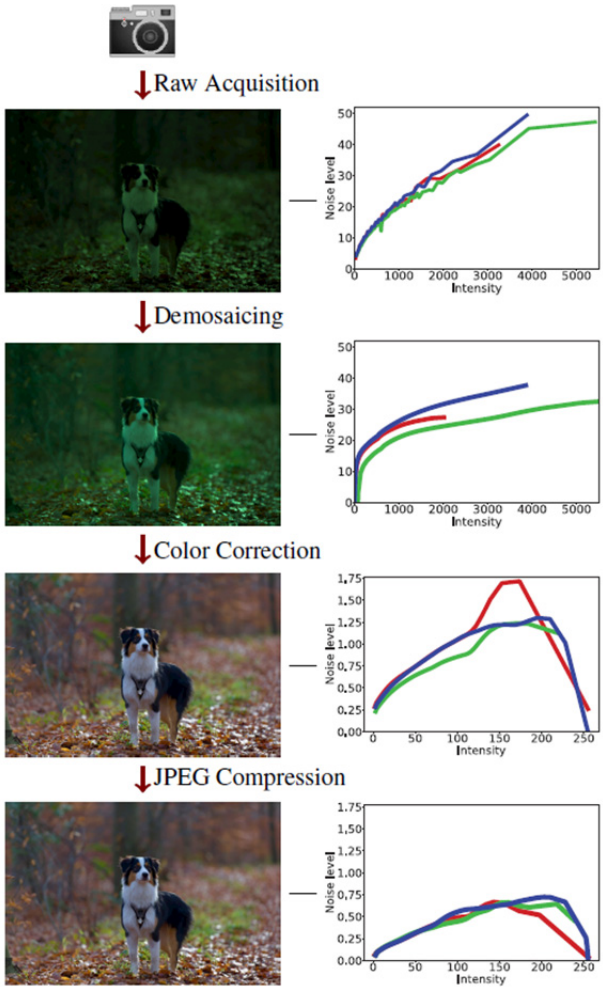


Figure 1.2. *Simplified processing pipeline of an image, from its acquisition by the camera sensor to its storage as a JPEG-compressed image. The left column represents the image as it goes through each step. The right column plots the noise of the image as a function of intensity in all three channels (red, green, blue). Because each step leaves a specific footprint on the noise pattern of the image, analyzing this noise enables us to reverse engineer the pipeline of an image. This in turn enables us to detect regions of an image which were processed differently, and are thus likely to be falsified*

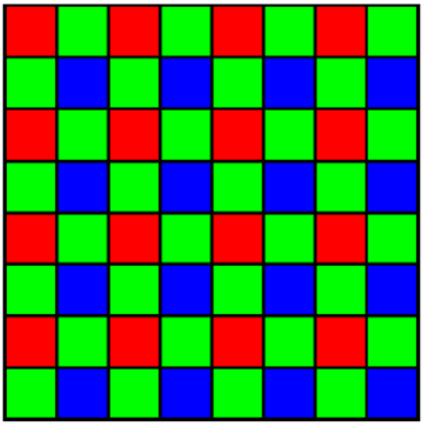


Figure 1.3. *The Bayer matrix is by far the most used for sampling colors in cameras*

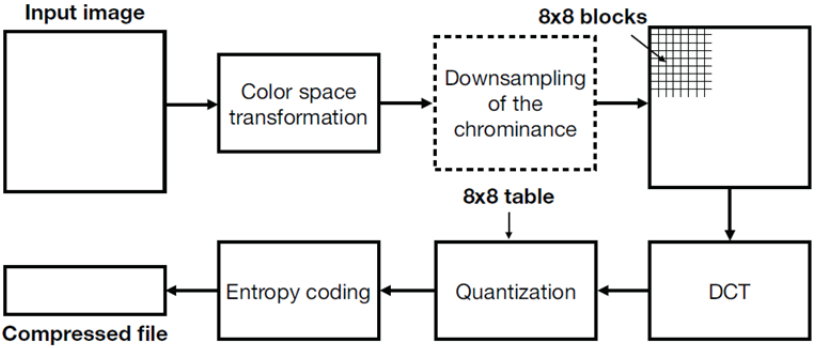


Figure 1.4. *JPEG compression pipeline*

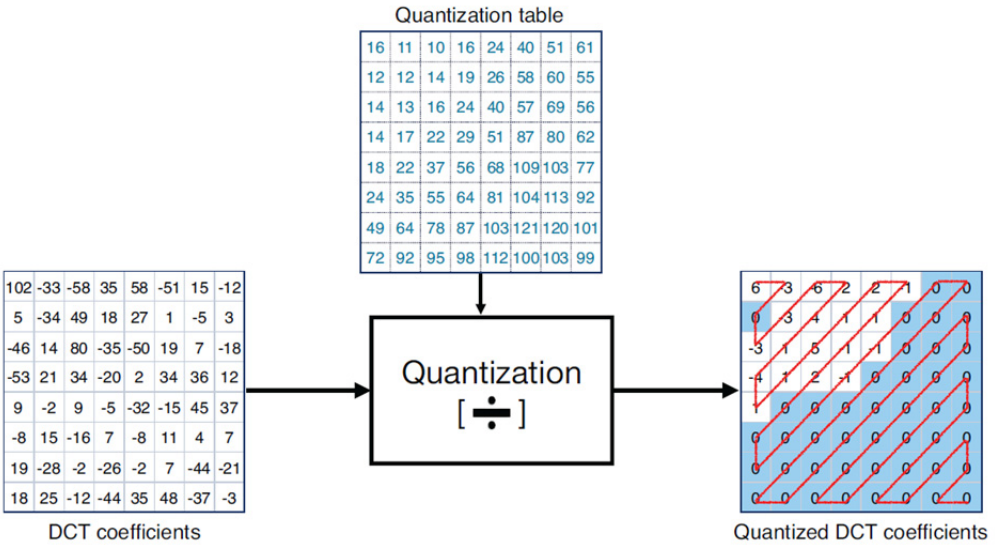


Figure 1.5. *An example of the impact of quantization on a DCT block. Each DCT coefficient is quantized by a value found in a quantization matrix. Rounding to the nearest integer results in many of the high-frequency coefficients being set to zero. Each block is zig-zagged to be encoded as a vector with a sequence of zeros*

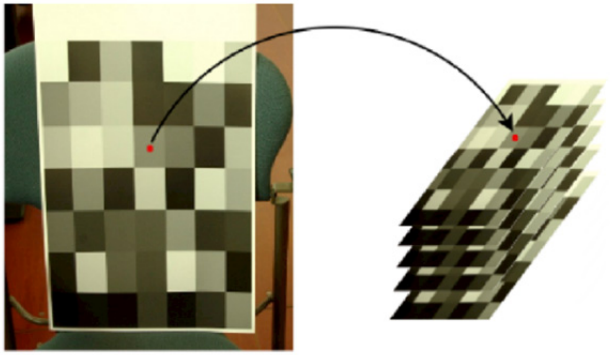
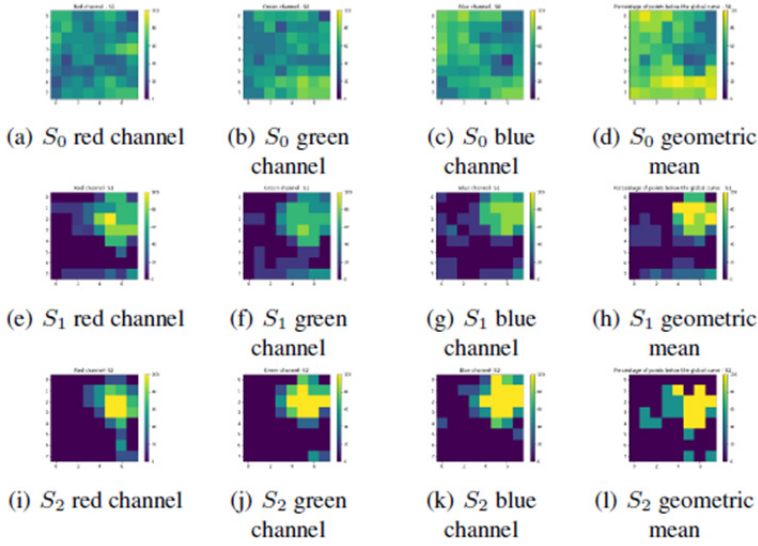


Figure 1.6. *Calibration model used for the construction of the temporal series*

(a) Original image (b) Auxiliary image (c) Falsified image

Figure 1.7. *Example of falsification: the vase in b) has been cut out and copied onto a), which gives c)*

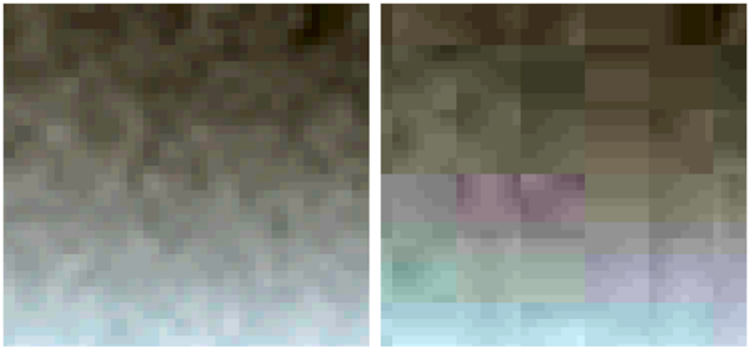


(a) S_0 red channel (b) S_0 green channel (c) S_0 blue channel (d) S_0 geometric mean

(e) S_1 red channel (f) S_1 green channel (g) S_1 blue channel (h) S_1 geometric mean

(i) S_2 red channel (j) S_2 green channel (k) S_2 blue channel (l) S_2 geometric mean

Figure 1.8. *Percentage of points below the global noise curve and geometric mean for each macro-block at S_0, S_1 and S_2*



(a) Uncompressed image (b) Compressed image with quality of 80

Figure 1.9. *Close-ups on an image before and after compression. The contrast has been enhanced to observe the JPEG artifacts, in particular the blocking effect, allowing us to see the edges of the 8 × 8 blocks*

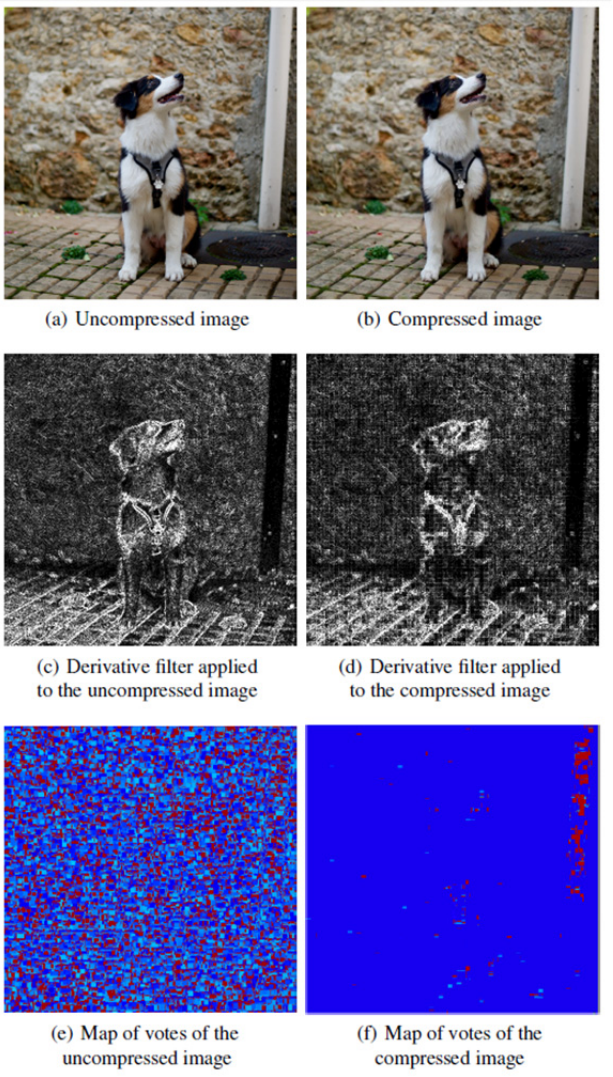


Figure 1.10. *Derivative filter and vote map applied to the same image without compression in a) and after JPEG compression of quality 80 in b)*

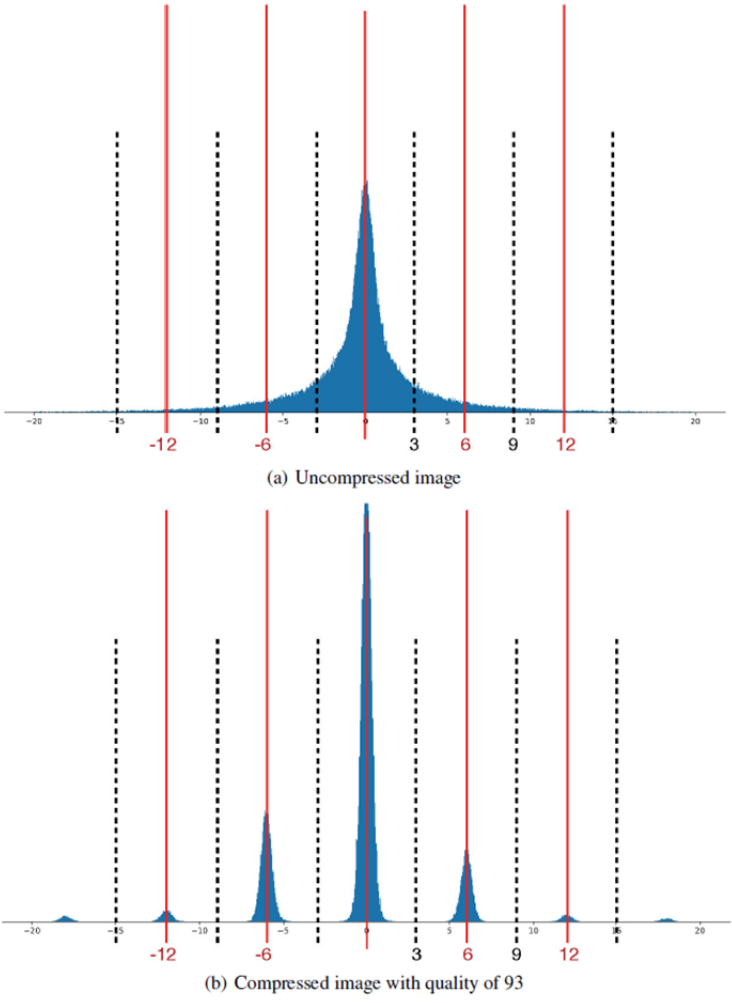
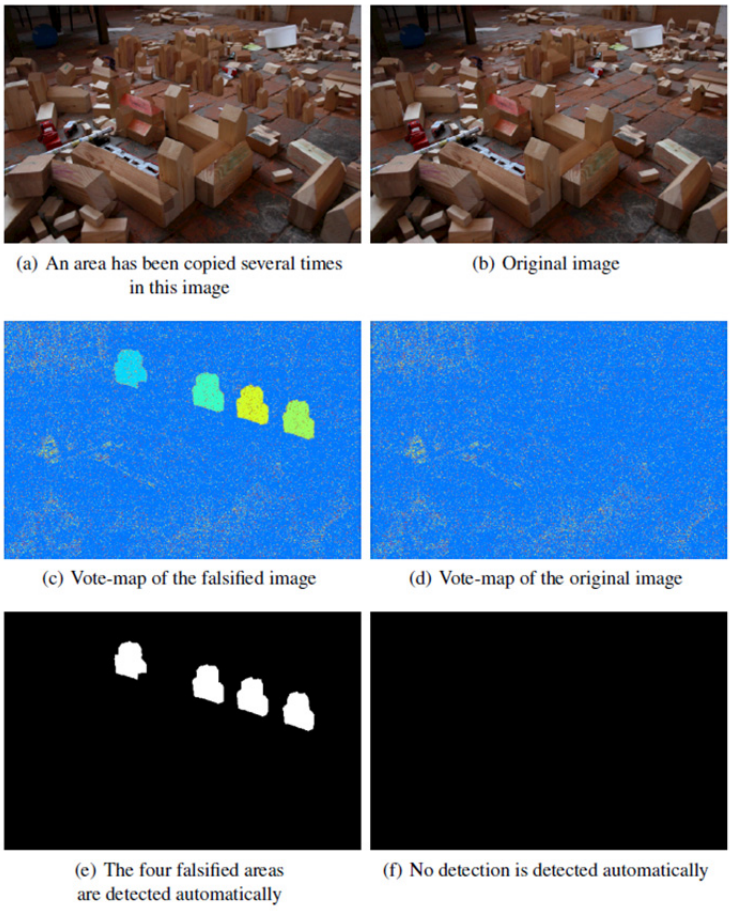


Figure 1.11. *Histogram of a DCT coefficient for an image before and after compression. There is a clear structure after quantization of the coefficients. The value of quantization is* $q = 6$



(a) An area has been copied several times in this image

(b) Original image

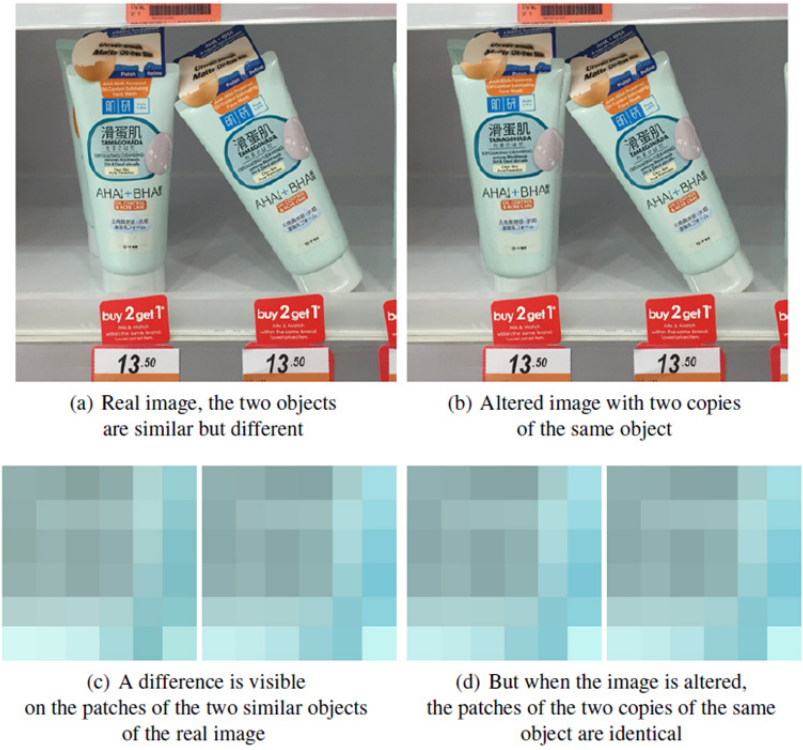
(c) Vote-map of the falsified image

(d) Vote-map of the original image

(e) The four falsified areas are detected automatically

(f) No detection is detected automatically

Figure 1.12. *In a), an area has been copied four times. The original image is shown in b)*



(a) Real image, the two objects are similar but different

(b) Altered image with two copies of the same object

(c) A difference is visible on the patches of the two similar objects of the real image

(d) But when the image is altered, the patches of the two copies of the same object are identical

Figure 1.13. *The image in a) represents two similar, but different objects, while the image in b) represents two copies of the same object. Both images come from the COVERAGE database (Wen* et al. *2016)*

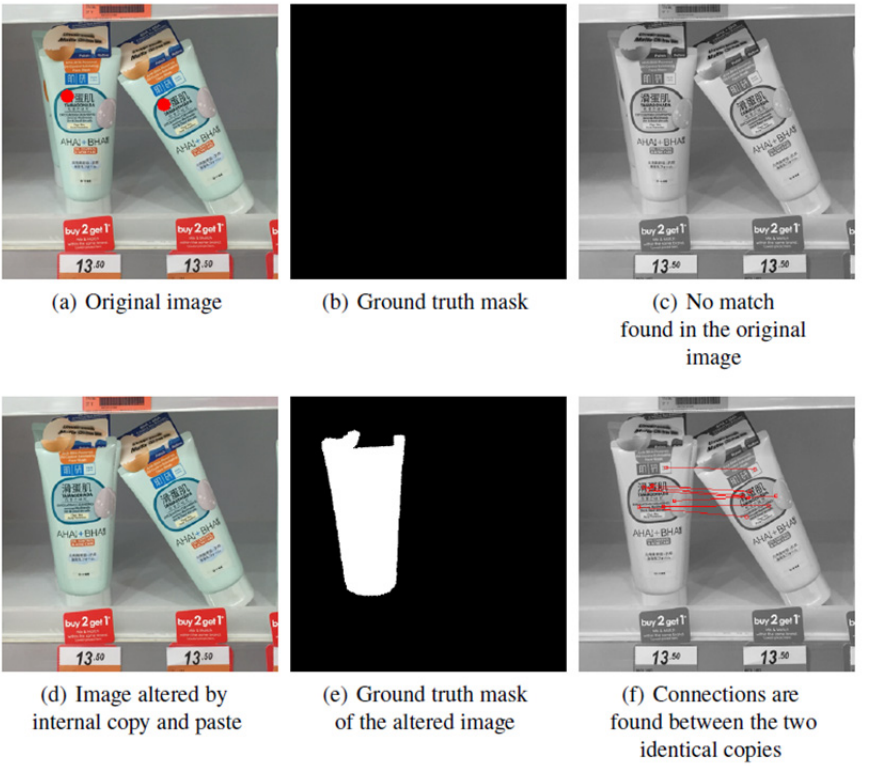


Figure 1.14. *Example of detection of copy–paste type modification on the images in Figure 1.13. The original and altered images are in (a) and (d), respectively, the ground-truth masks in (b) and (e), and the connections (Ehret 2019) between the areas detected as too similar in (c) and (f)*

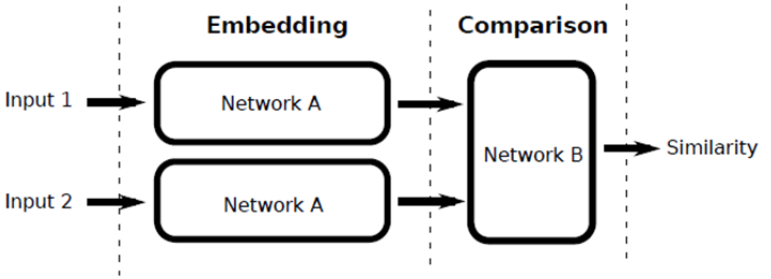
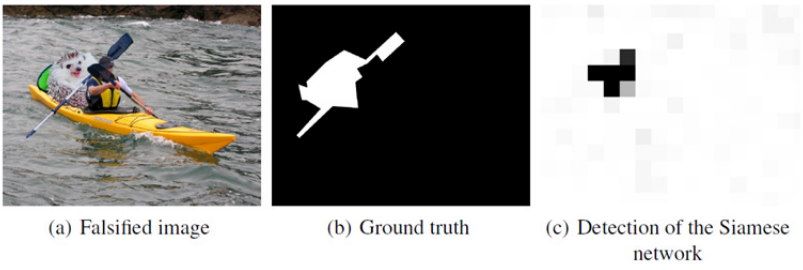


Figure 1.15. *Structure of the Mayer and Stamm (2019) network to compare the source of two patches. The same first network A is applied to each patch to extract a residue. These residues are then passed on to a network B which will compare their source and decide if the patches come from the same image or not*



(a) Falsified image (b) Ground truth (c) Detection of the Siamese network

Figure 1.16. *Example of modification detection with the Siamese network (Mayer and Stamm 2019)*

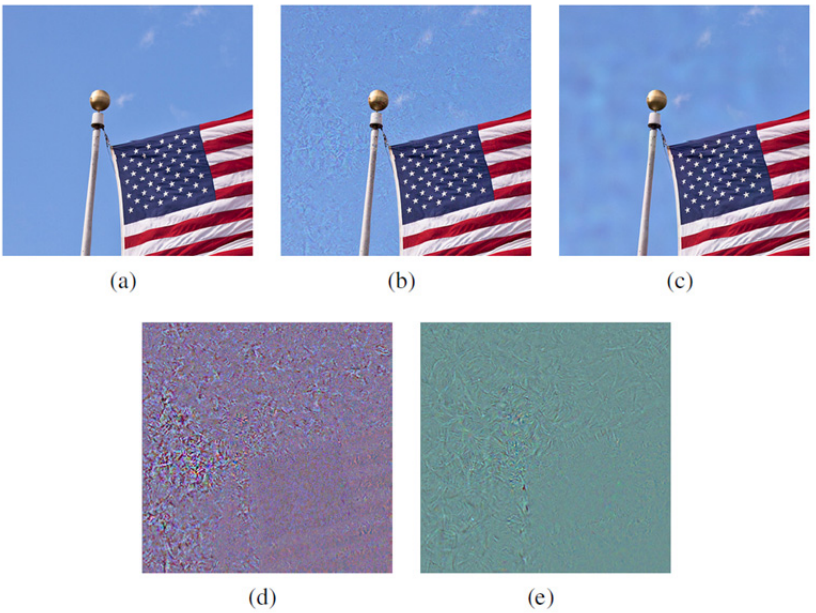


Figure 2.1. *The original image and adversarial images; the manipulations are almost invisible, the classification is incorrect*

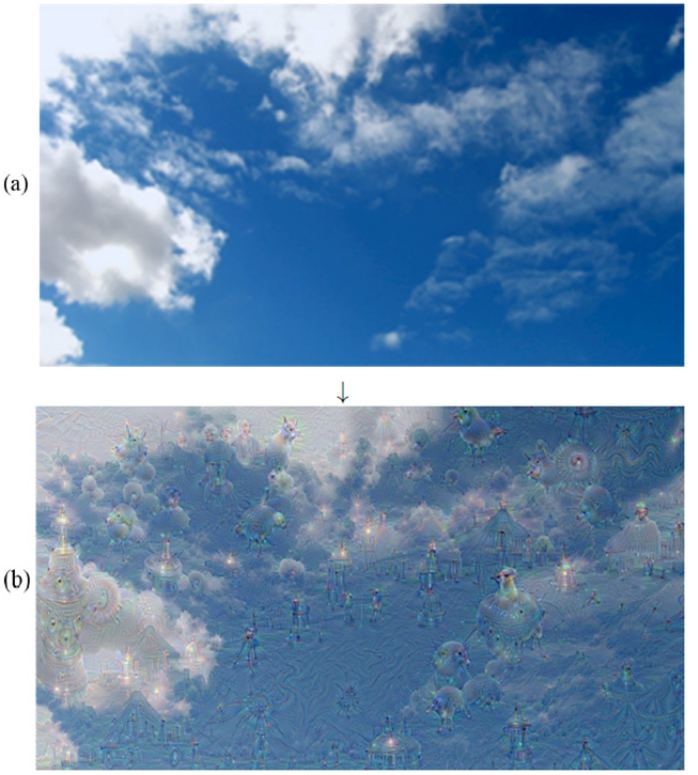


Figure 2.2. *Illustration of the Deep Dreams process applied to the original image (a) (source: (Wikipedia 2020))*

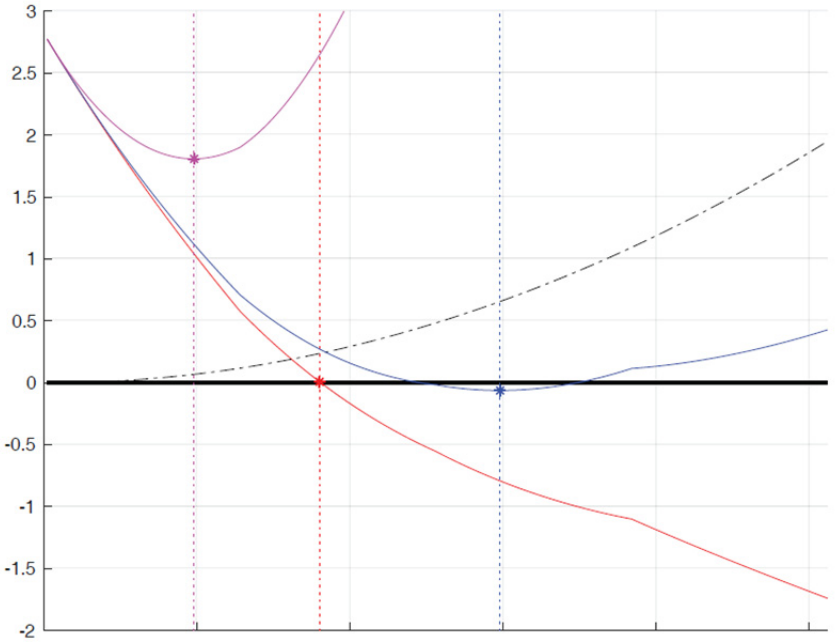


Figure 2.3. *Illustration of L-BFGS in 1D*

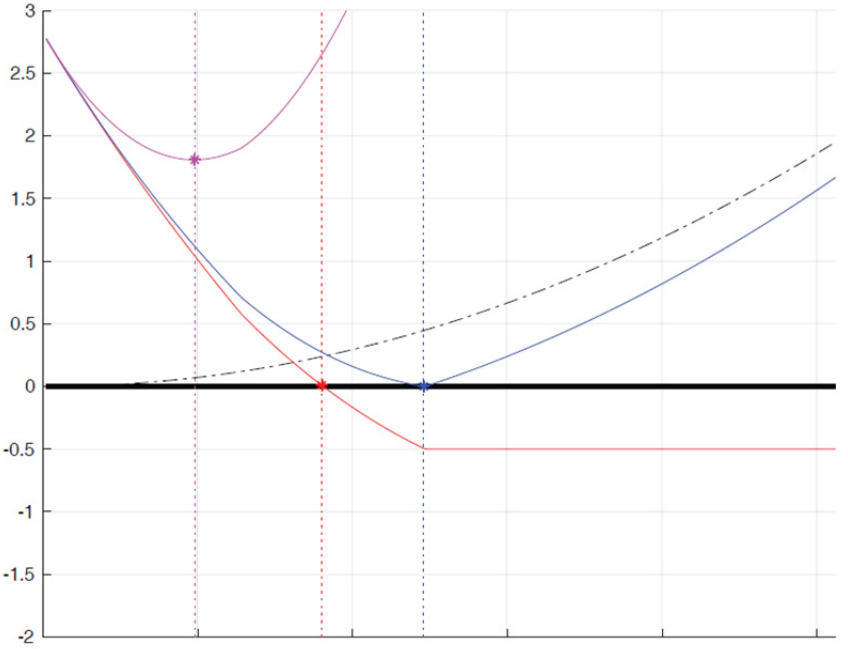


Figure 2.4. *Illustration of C&W in 1D. The perturbation* **r** *is collinear to the gradient of the objective function. This is the same configuration as in Figure 2.3, except the margin* $\mu = 0.5$ *for the threshold of equation [2.15]. Notice its effect: the blue minimum is closer to the red asterisk*

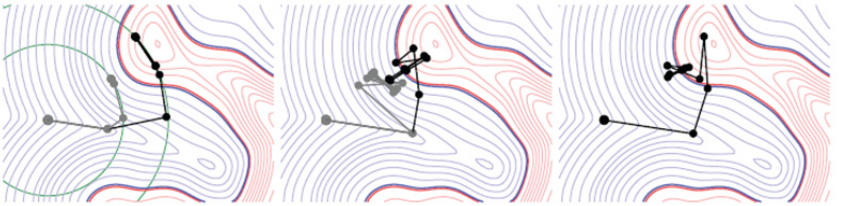


Figure 2.5. *Illustration, in two dimensions, of adverse attacks on a binary classifier*

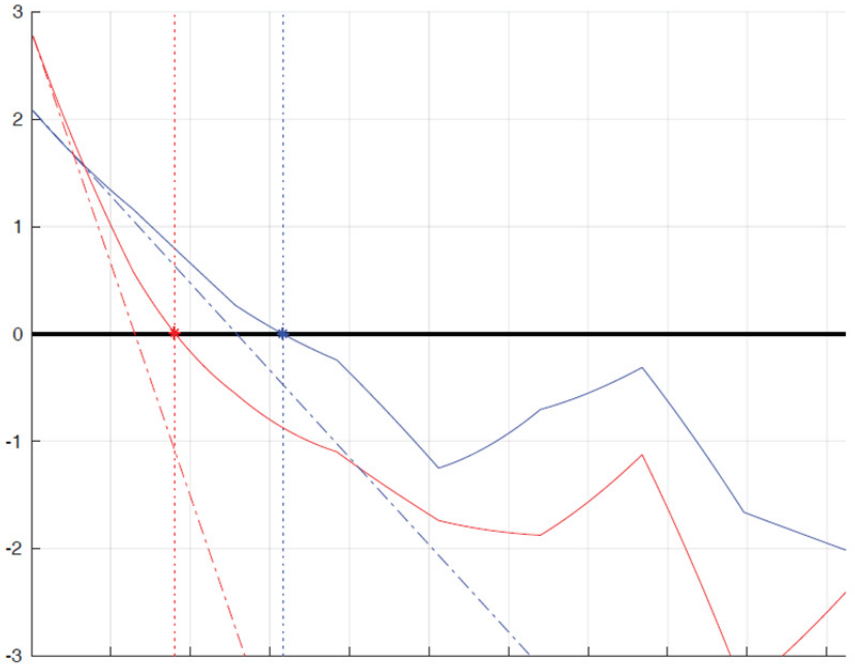


Figure 2.6. *Illustration of DeepFool*

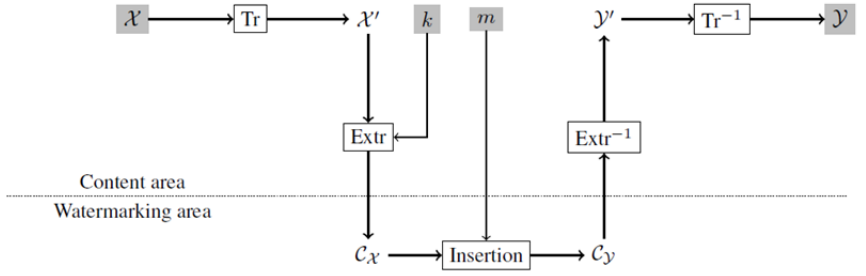


Figure 3.1. *Classical diagram of watermark insertion*

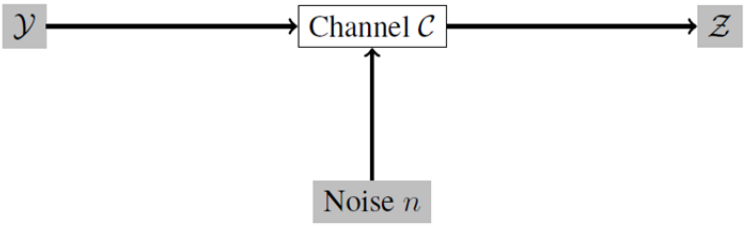


Figure 3.2. *Diagram of the transmission of an image in a channel affected by noise*

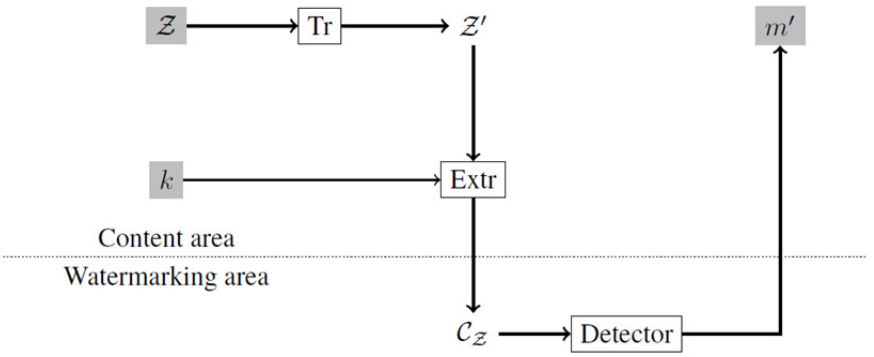


Figure 3.3. *Classical diagram of detection of a mark*

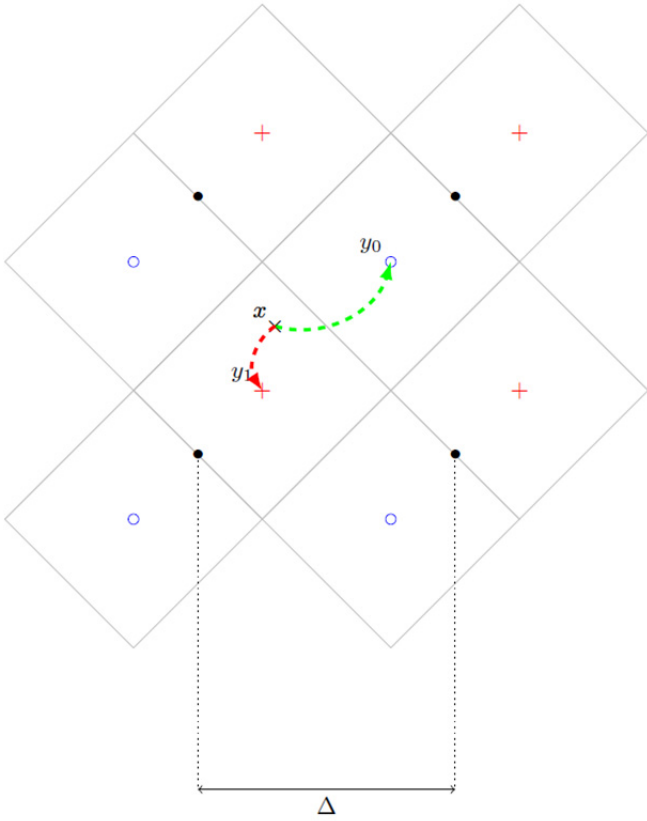


Figure 3.4. *Representation of the quantization space (or Euclidean network) in dimension L = 2. The symbol + represents bit 1 (cosets Λ_1) and ◦ represents bit 0 (coset Λ_0)*

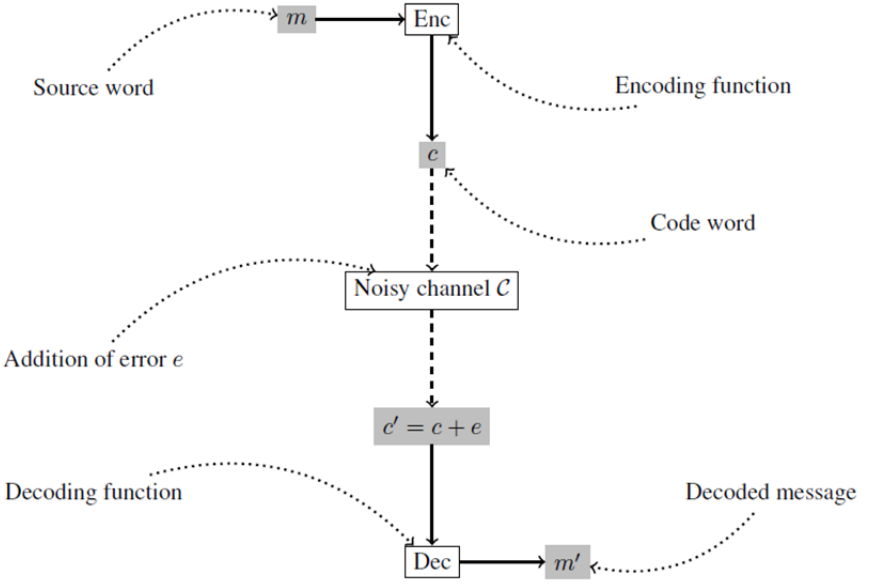


Figure 3.5. *The different stages allowing the reliable transmission of a message in a channel*

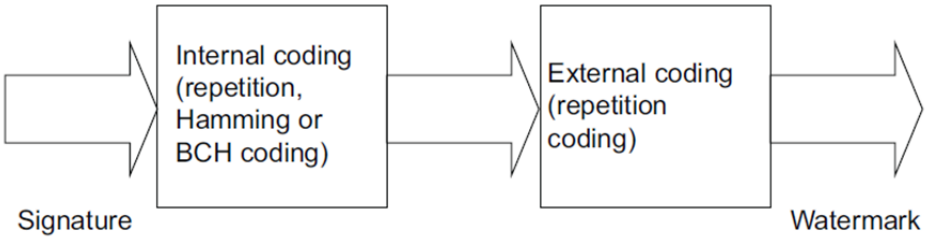


Figure 3.6. *Strategy by "concatenation codes"*

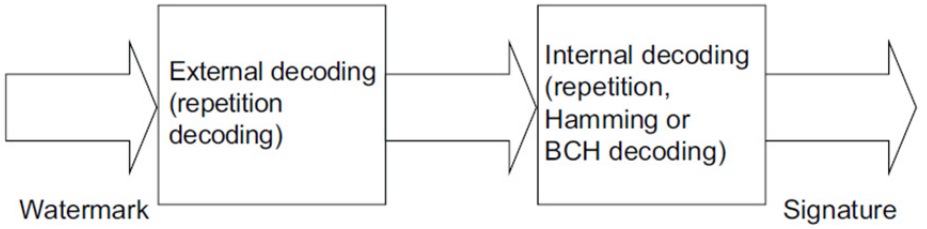


Figure 3.7. *Decoding in the case of the code by "concatenation codes"*

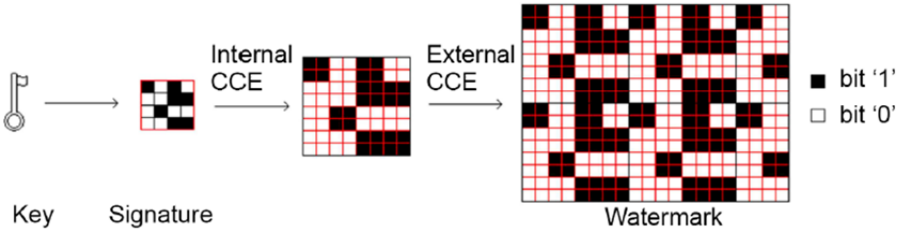


Figure 3.8. *Illustration of the creation of a mark by code concatenation*

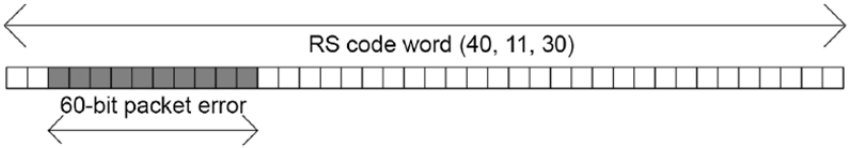


Figure 3.9. *Performance of RS codes against packet errors*

$$m = (m_1, \ldots, m_k) \longmapsto \underbrace{c = mG_1 = (c_1, \ldots, c_{n_1})}_{\substack{\text{Encoding of } m \text{ by } \mathcal{C}_1 \\ n_1 - k \text{ repetition bits}}} \longmapsto \underbrace{d = cG_2 = (d_1, \ldots, d_{n_2}).}_{\substack{\text{Encoding of } c \text{ by } \mathcal{C}_2 \\ n_2 - n_1 \text{ repetition bits}}}$$

Figure 3.10. *Concatenation diagram (or hybrid coding) of two correction codes $\mathcal{C}_1(n_1, k)$ (inner coding) and $\mathcal{C}_2(n_2, n_1)$ (outer coding) of respective generator matrices G_1 and G_2. The word m is a required message of k bits. The final code word d is inserted into the image as a mark*

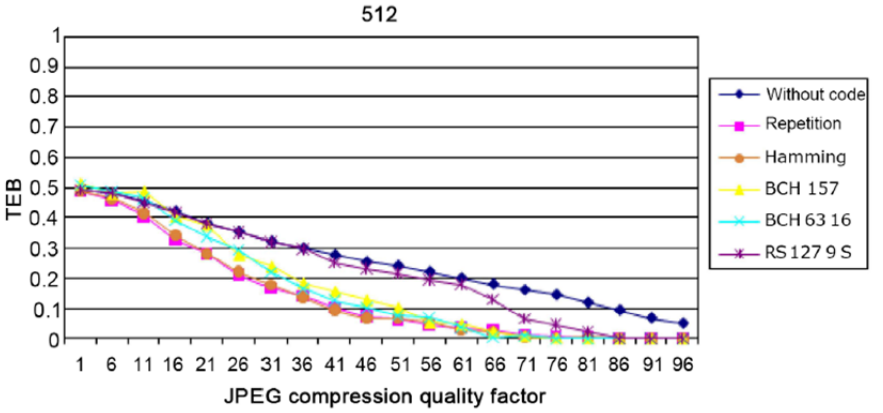


Figure 3.11. *Comparison of the robustness of different codes against the JPEG attack*

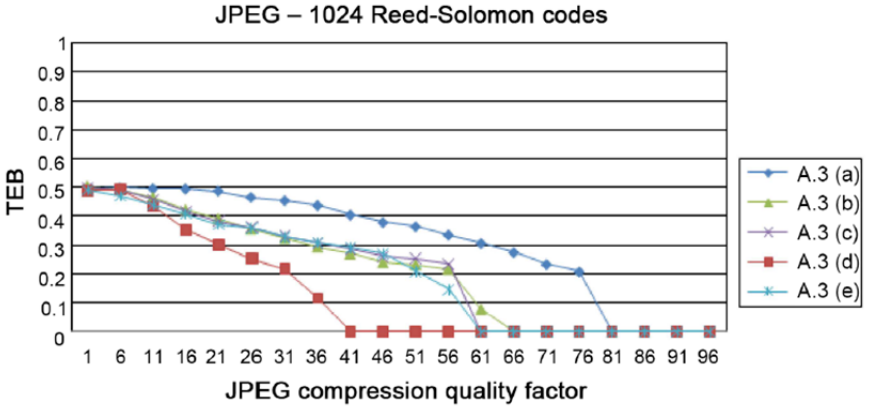


Figure 3.12. *Comparison for different images of the Kodack database of robustness with an RQ encoding and an attack JPEG Quality = 40*

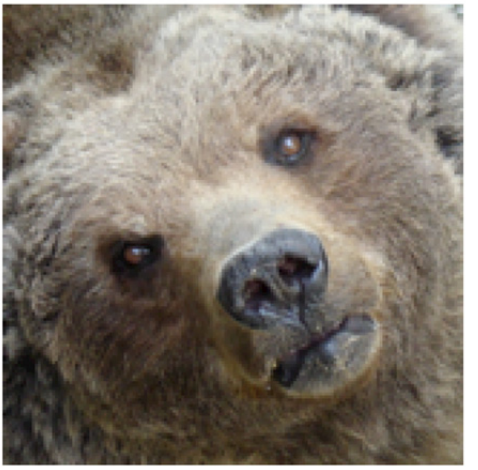


Figure 3.13. *Image of a bear*

Figure 3.14. *Image of a plant*

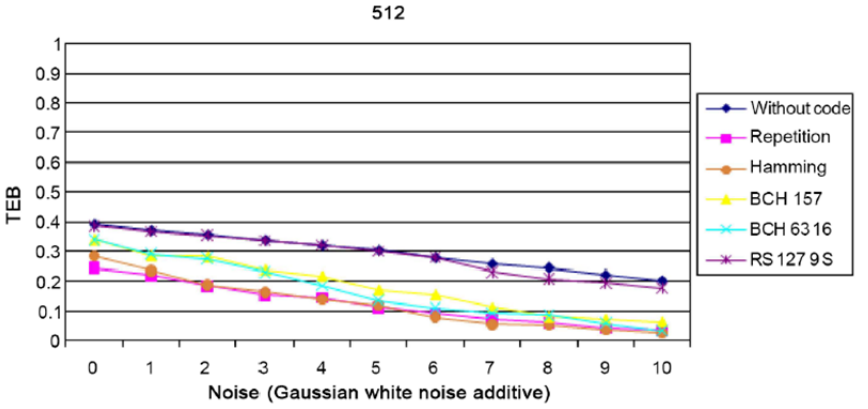


Figure 3.15. *Comparison of the different encodings against an attack by adding Gaussian white noise*

Figure 3.16. *Effects of changes to saturation on the Lena image. a) Minimal change to saturation: −100, b) maximal change to saturation: 100*

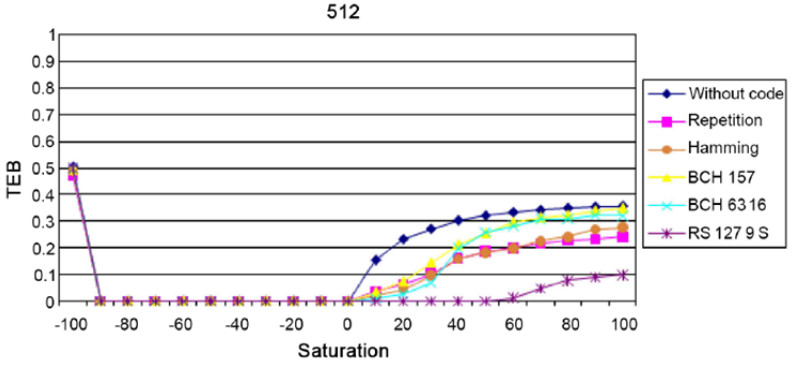


Figure 3.17. *Comparison of coding processes faced with saturation for an image of size 512 × 512*

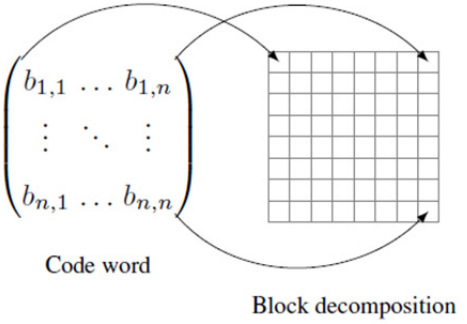


Figure 3.18. *Insertion strategy using rank metric code and image block decomposition. Each bit $b_{i,j}$ is associated with a block*

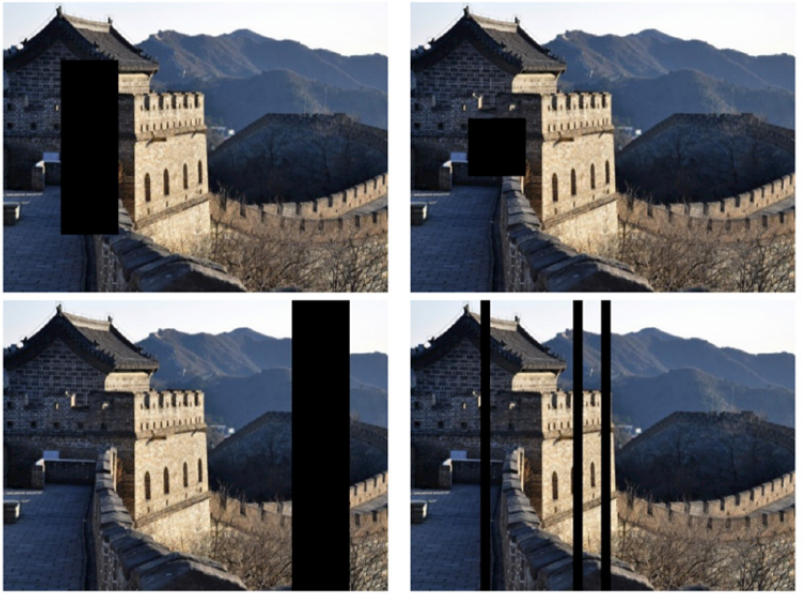


Figure 3.19. *Cropped images with errors from the same rank*

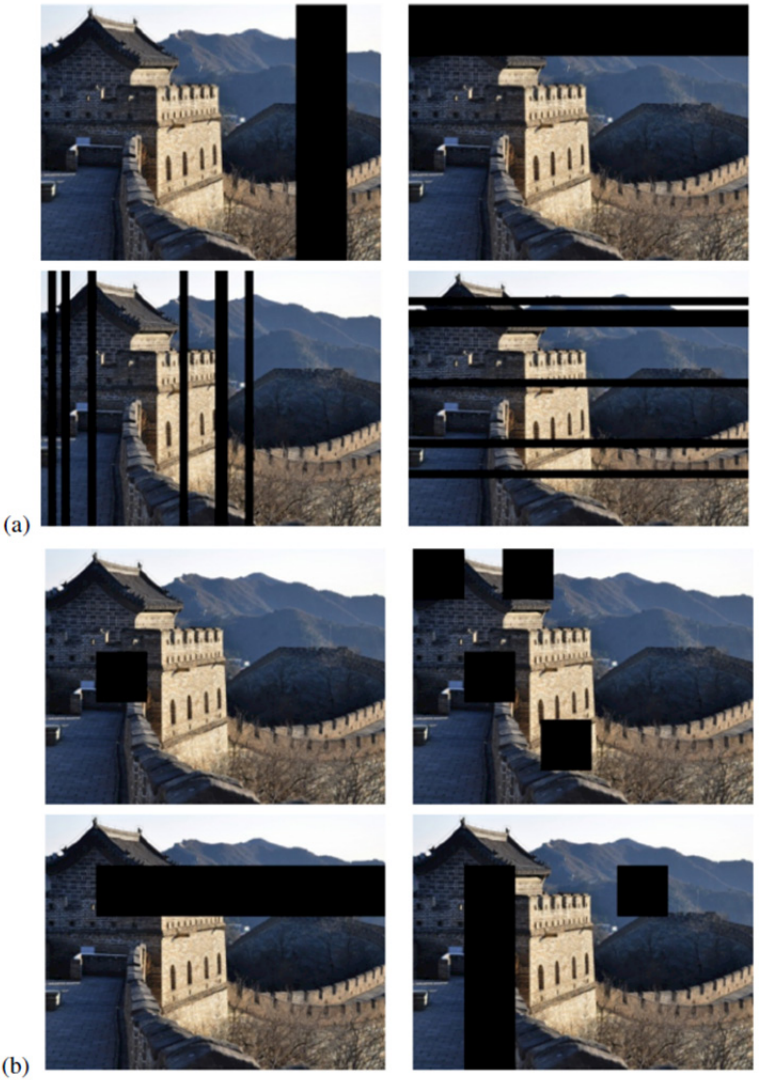


Figure 3.20. *Types of image cropping. The top two rows (a) represent type 1 errors and the two bottom rows (b) represent type 2 errors*

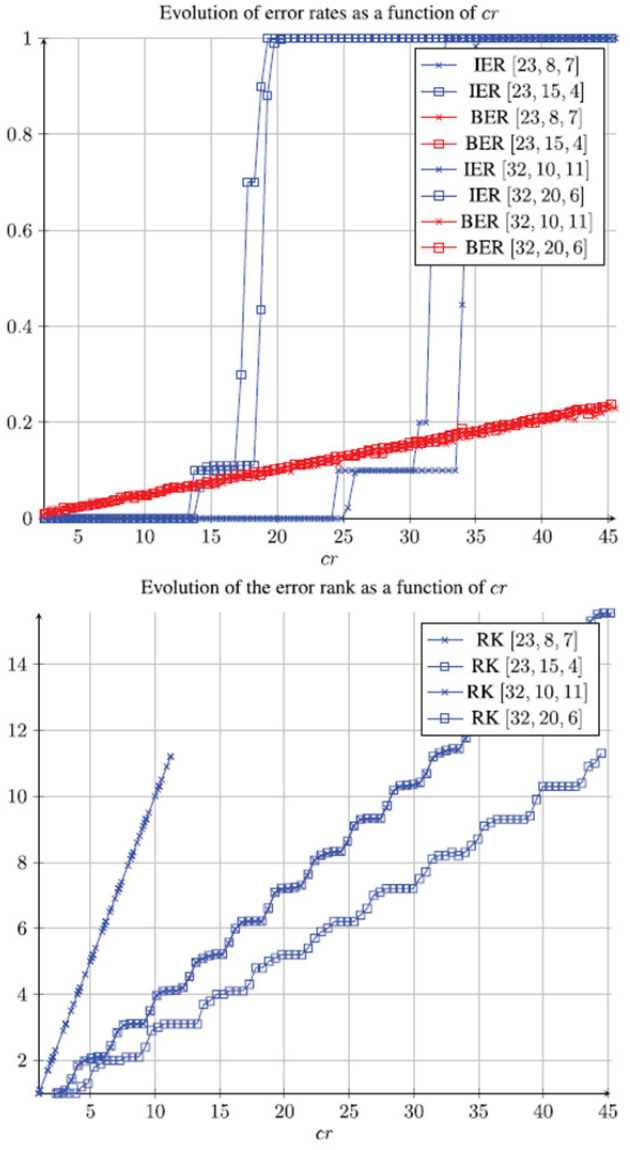


Figure 3.21. *Average error rate and rank as a function of the cropping percentage* cr

Figure 3.22. *Examples of attacked images with the poorest detection performance. The error rank is large ($\geq (n-k)/2$) compared with the percentage of cropped area (cr')*

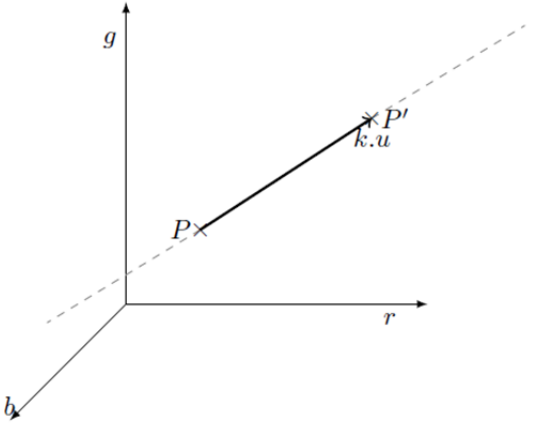


Figure 4.1. *Quantization in the RGB color space on an oriented line by a direction vector u*

(a) Source image (b) Random direction constant

(c) u_g direction constant (d) Optimal direction suitable for each color

Figure 4.2. *Example of inserting a mark with different approaches and direction vectors with equivalent digital distortion (overall, the signal to noise ratio is equivalent for the three images). The image used is a cropped version of the image kodim23.png from the Kodak Image Database (available at: http://r0k.us/graphics/kodak/kodim23.html)*

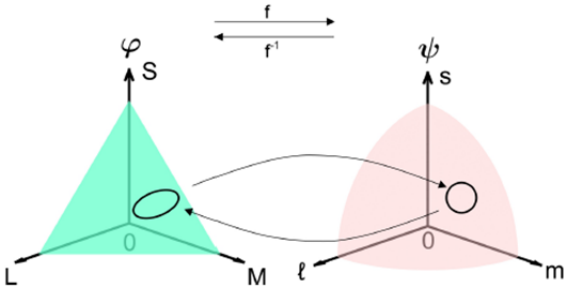


Figure 4.3. *Relationship between physical space φ and perceptual space ψ*

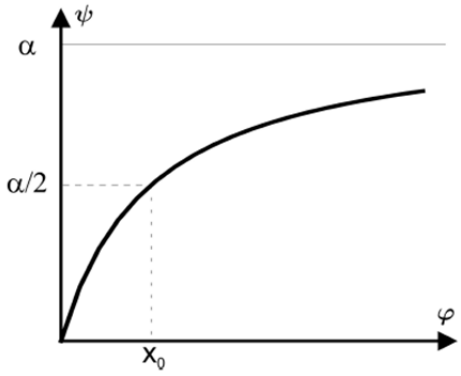


Figure 4.4. *Nonlinear function of perception*

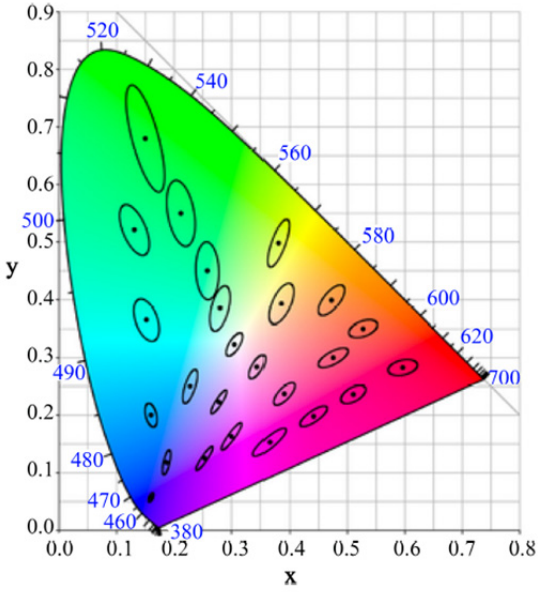


Figure 4.5. *MacAdam ellipses in the luminance plane of the color space xyY, 1931. Ellipses are 10 times larger than their original sizes*

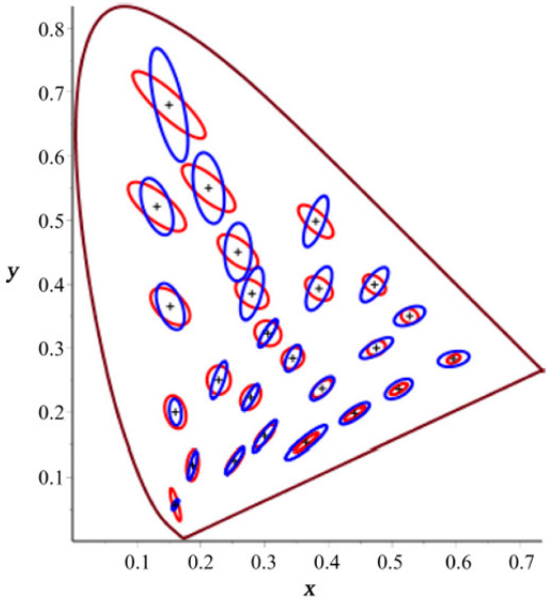


Figure 4.6. *The ellipses obtained from the psychovisual model almost correspond to the MacAdam ellipses after an optimal choice of constants (Alleysson 1999)*

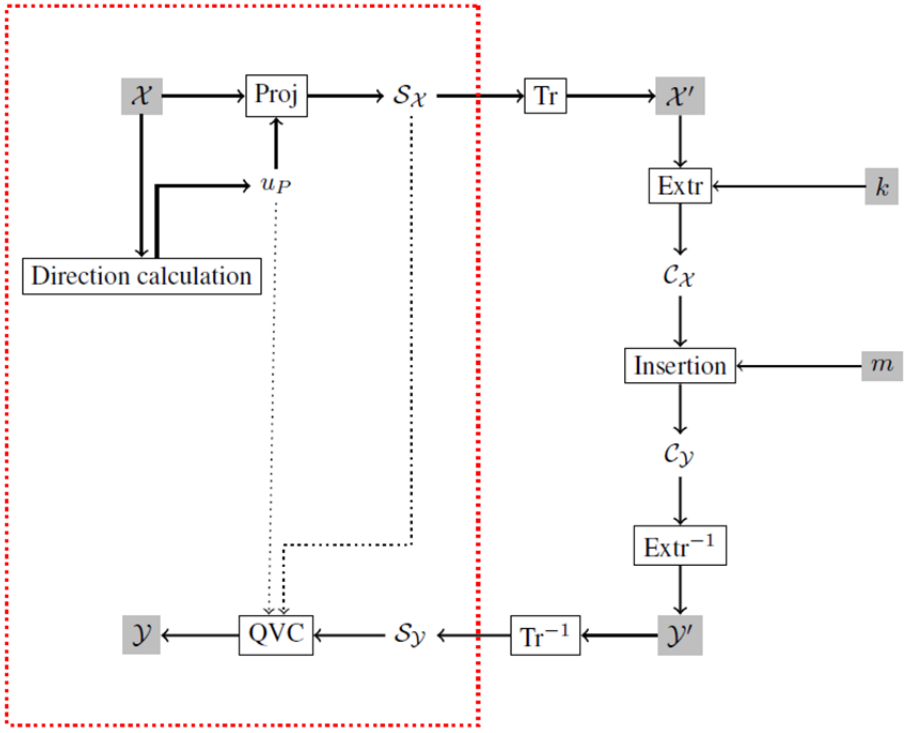


Figure 4.7. *Classical insertion diagram combined with color vector quantization (QVC) based on a psychovisual model. The elements within the red frame represent the steps of vector quantization. Tr() and Extr() are the space transformation and coefficient extraction functions, k is the secret key and m is the binary message*

Figure 4.8. *Pairs of images (host image $\mathcal{X}$, associated scalar image $\mathcal{S}_{\mathcal{X}}$). Random images from the Corel database (available at: https://sites.google.com/site/dctresearch/Home/content-based-image-retrieval)*

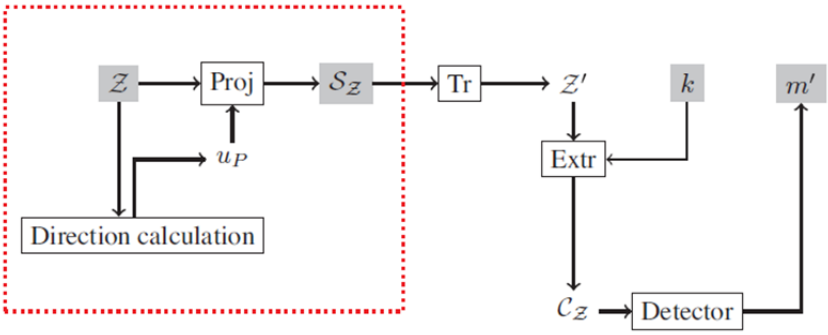


Figure 4.9. *Classical mark detection diagram. As with insertion, we find the extraction part of the scalar image $\mathcal{S}_{\mathcal{Z}}$ framed in red*

Figure 4.10. *Cropped color images (Lena and Kodak base of size 60 × 60) marked with the LQIM method (GA approach), DWR ≃ − 5.5 dB on average and ER = 0.5*

Figure 4.11. *Cropped color images (Lena and Kodak base of size 60 × 60) marked with the LQIM method (AA approach), DWR ≃ − 5.5 dB on average and ER = 0.5*

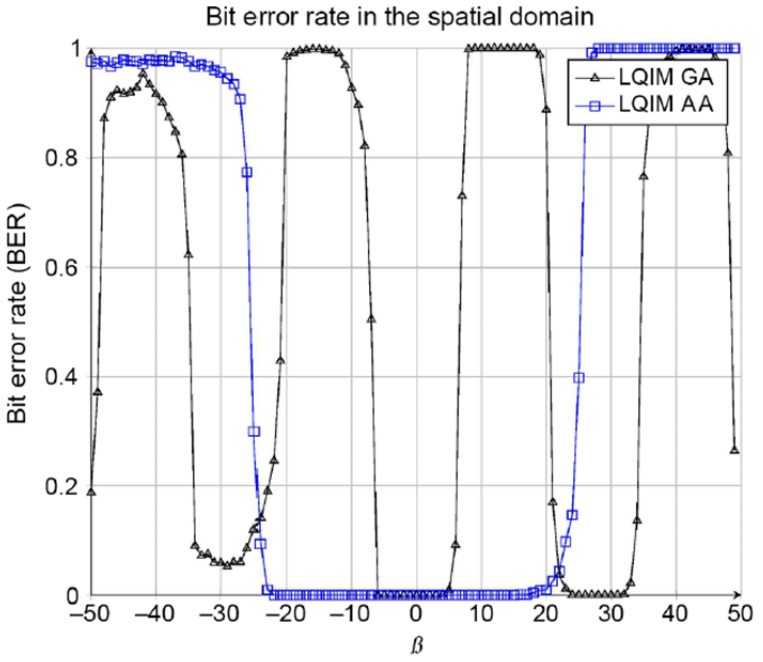
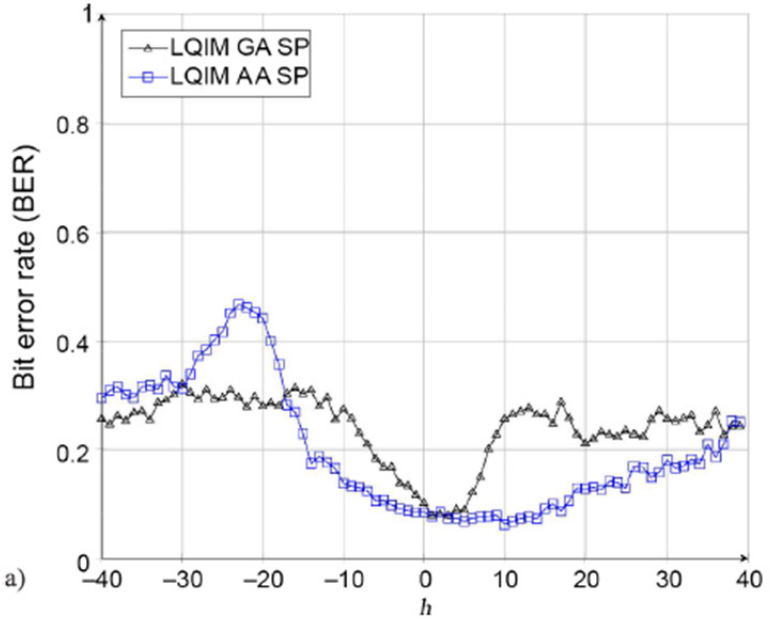
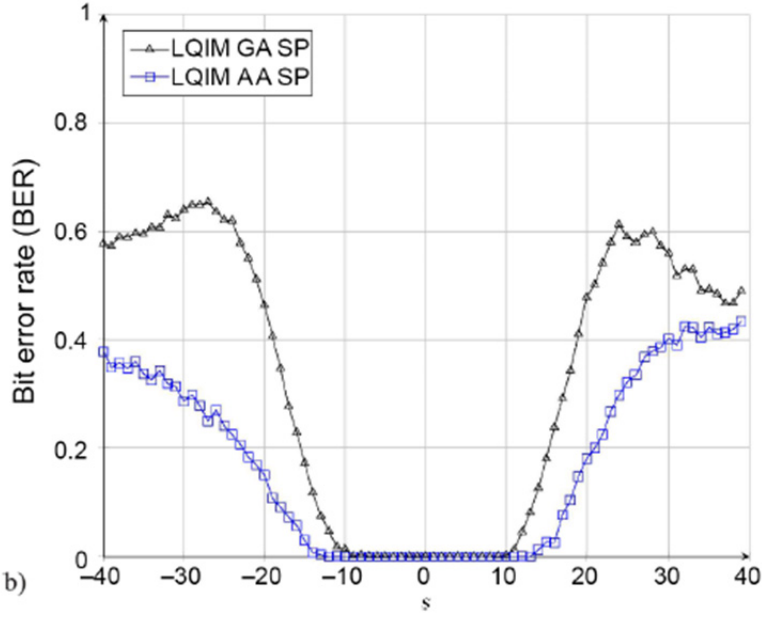


Figure 4.12. *Binary error for methods GA and AA depending on the parameter* β


LQIM GA SP
LQIM AA SP
Bit error rate (BER)
1
0.8
0.6
0.4
0.2
0
−40
−30
−20
−10
0
10
20
30
40
h

a)


LQIM GA SP
LQIM AA SP
Bit error rate (BER)
1
0.8
0.6
0.4
0.2
0
−40
−30
−20
−10
0
10
20
30
40
s

b)

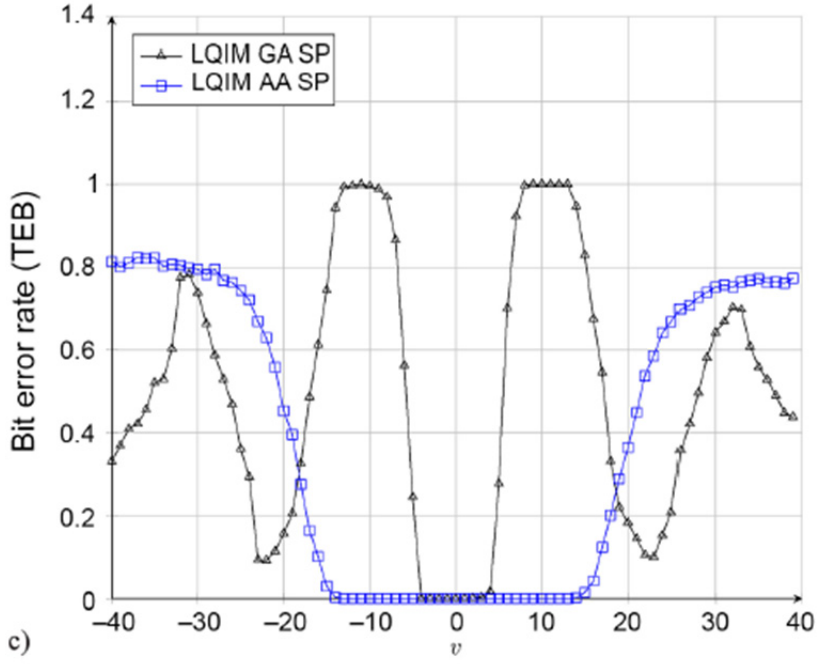


Figure 4.13. *Bit error rate for (a) hue; (b) saturation; and (c) value modification*

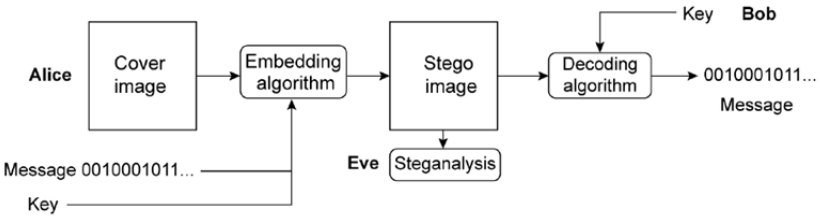


Figure 5.1. *General operation of steganography and roles of different actors (Alice, Bob and Eve)*

Figure 5.2. *Coding for steganography: the coding system generates several code words associated with the same message, m, in order to select the one that is closest to the cover content, and so minimizes the distortion between the cover content, and the stego content*

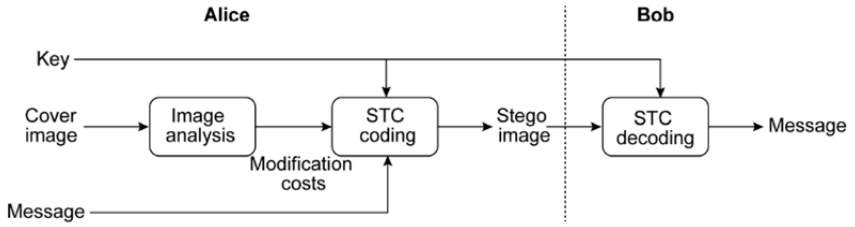


Figure 5.3. *General principle of embedding, encoding and decoding in steganography. The secret key can be used to encrypt the message, permute samples of cover content and/or configure the STC*

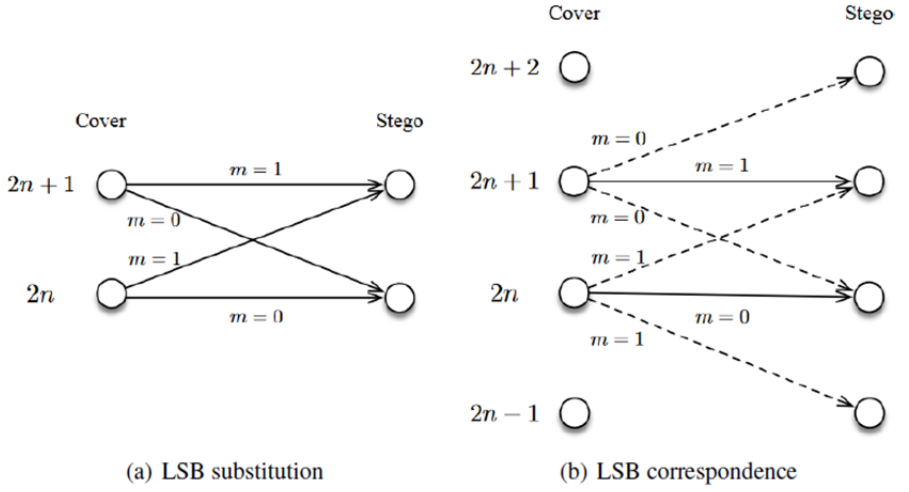
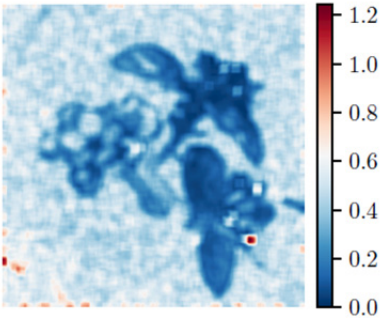
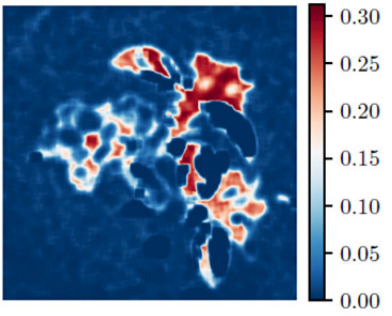


Figure 5.4. *Principles of LSB substitution and LSB correspondence, the dotted arrows represent a modification performed with a probability of 0.5,* $n \in \mathbb{N}^+$

(a) cover

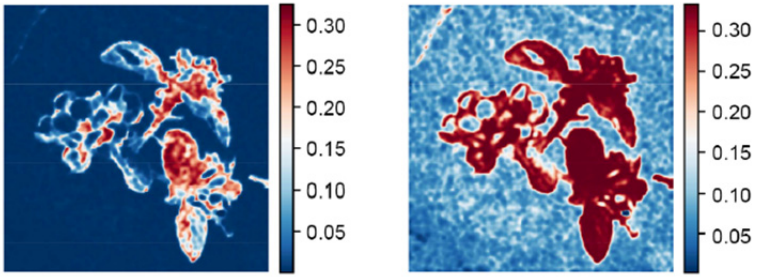


(b) Embedding cost



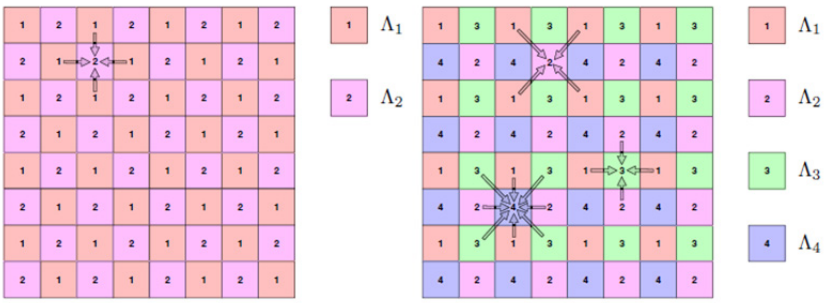
(c) Embedding probabilities (0.3 bit per pixel)

Figure 5.5. *a) Cover Image. b) Associated cost map for the HILL algorithm. Image textures are associated with the lowest costs. c) Modification probability map, the maximum probability being equal to 1/3 for a ternary embedding and the maximal embedding rate at* $\log_2(3) \simeq$ *1.6 bit per pixel*



(a) 0.3 bits per pixel (b) 1.0 bit per pixel

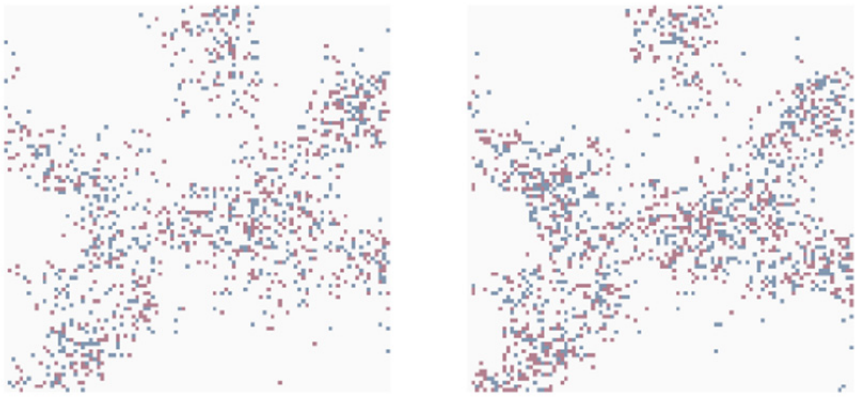
Figure 5.6. *Map of associated modification probabilities for the MiPod algorithm for two different embedding rates*



(a) Embedding on 2 lattices (dependency on 4 connections)

(b) Embedding on 4 lattices (dependency on 8 connections)

Figure 5.7. *Principle of synchronized embedding using two or four lattices*



(a) HILL (b) CMD

Figure 5.8. *Visualization of modifications (− 1, + 1) made to the image without synchronization using a) the HILL method and b) the CMD strategy*

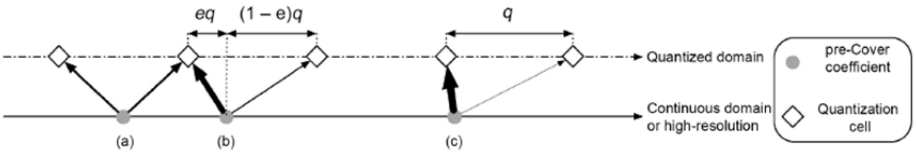
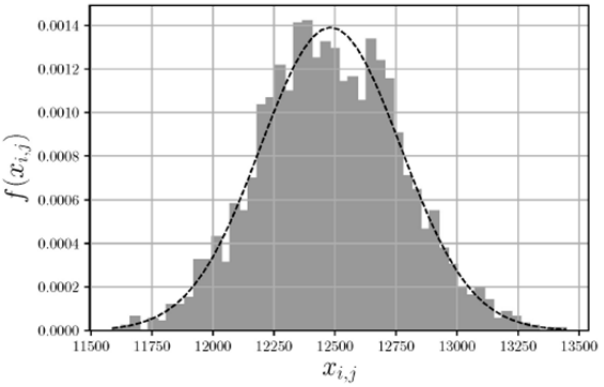
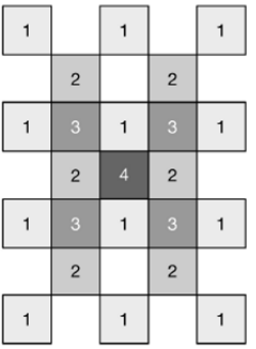


Figure 5.9. *Principle of weighting by the quantization error: situation (a), where the unquantized value is equidistant from two quantized values, will be more favourable to the modification of the value toward the surrounding quantized area than situation (c), where the unquantized value is very close to the quantized value. Situation (b) is intermediate*



(a) Standardized histogram of photonic noise (Leica Monochrome, ISO 4 000) in the RAW domain and associated normal distribution (dotted lines)



(b) JPEG intervening blocks for an embedding carried out in the central block

Figure 5.10. *Natural steganography seeks to mimic the normal distribution of the photonic noise a); this distribution in the JPEG domain becomes multivariate, and it must use a lattice decomposition to take the correlations between the DCT coefficients b) into account*

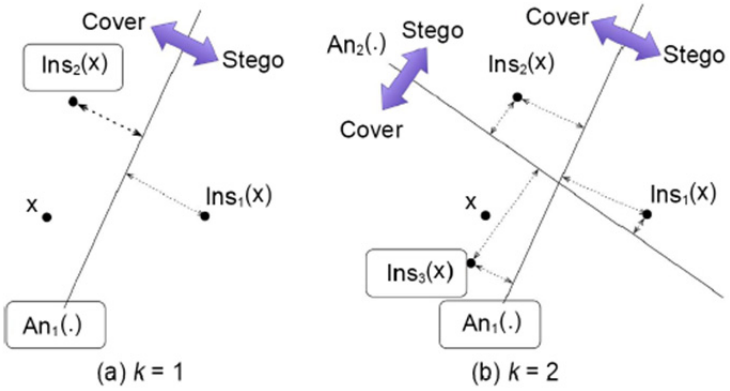


Figure 5.11. *The first two iterations of the strategy proposed by Bernard* et al. *(2019) on a toy example, where the enemies are linear classifiers and the marked distance at the decision boundary represents the output value of the functions* $\mathrm{An}_k()$

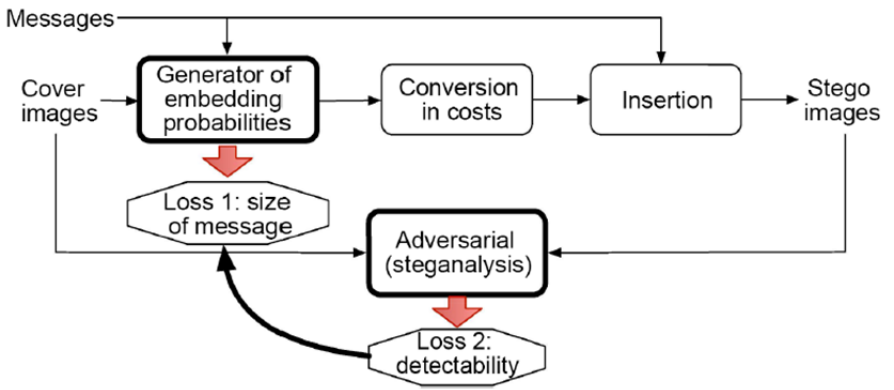


Figure 5.12. *Principle of steganography based on adversarial generators*

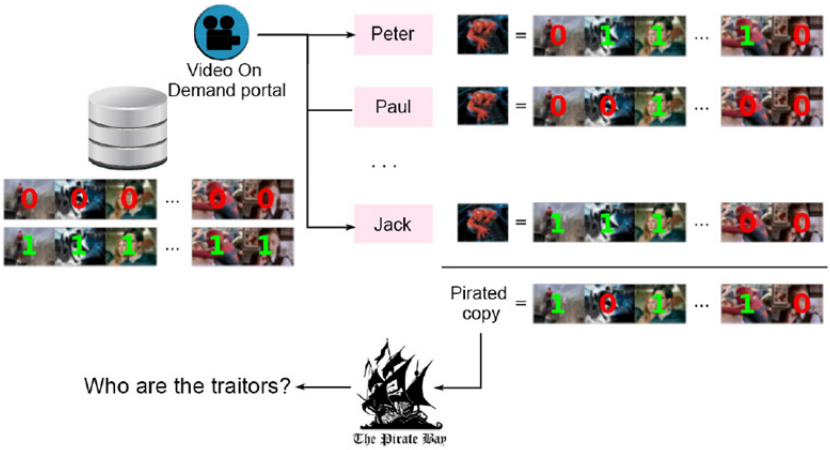


Figure 6.1. *Sequential watermarking of a movie. A thumbnail image represents a video block, watermarked to hide the symbol "1" or "0". The traitors sequentially form a pirated movie by selecting one of their blocks*

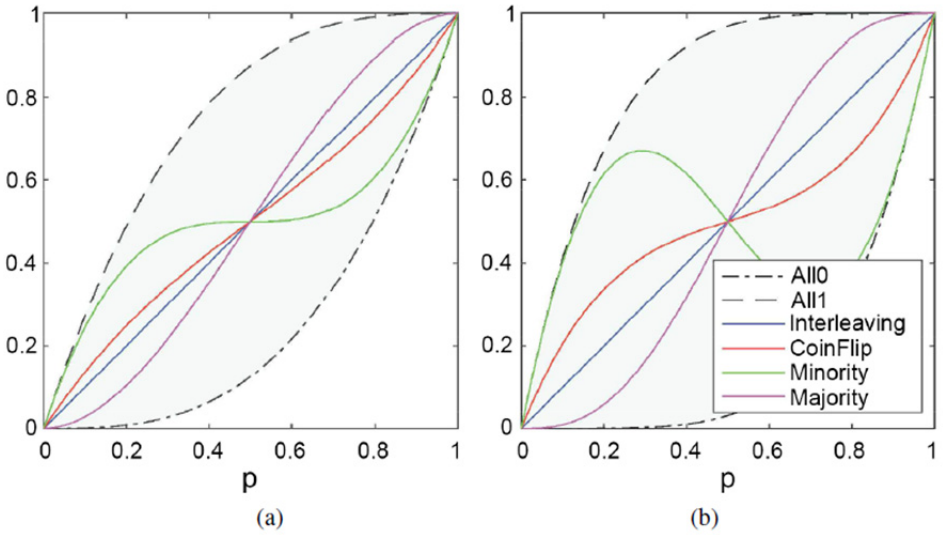


Figure 6.2. $\Pi(p) := \mathbb{P}(Y = 1|p)$ *for* $c = 3$ *a) and* $c = 5$ *b). All 1 attack:* $\Pi(p) = 1 - (1-p)^c$*, all 0 attack:* $\Pi(p) = p^c$*, interleaving:* $\Pi(p) = p$*, coin flip:* $\Pi(p) = (1 - (1-p)^c + p^c)/2$*, minority and majority do not have a simple formula. This function remains in the blue zone, marked by the "All-1" and "All-0" strategies*

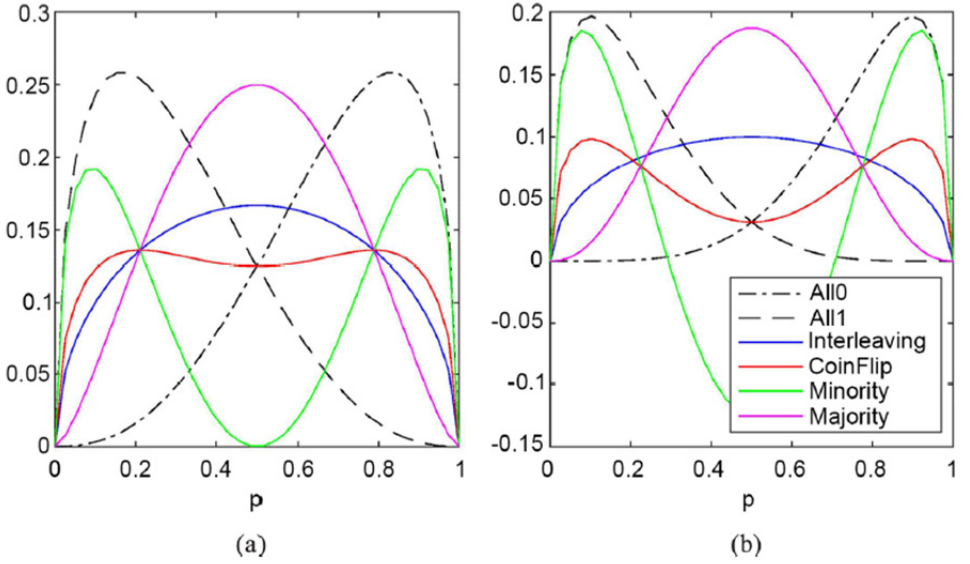


Figure 6.3. *The function* $p \to \mathbb{E}(\bar{U}(X_{\text{tra}}, Y, p))$ *for c = 3 (a) and c = 5 (b)*

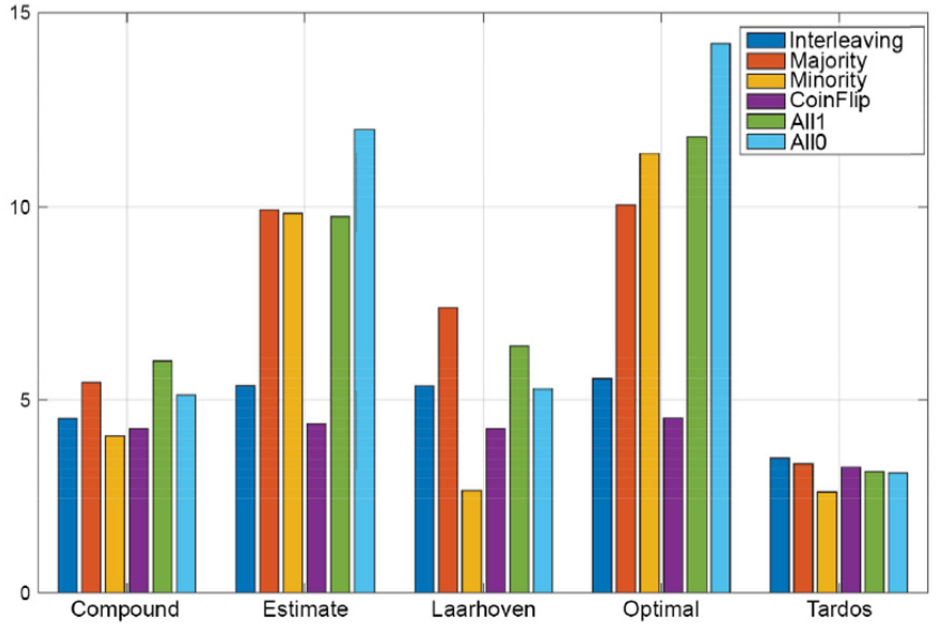


Figure 6.4. *The quantity* $|\log_{10}(\mathbb{P}_{\text{fp}})|$ *for* $\mathbb{P}_{\text{fn}} \approx 1/2$ *for the six collusion strategies in section 6.1.4 and the five following score functions: "Compound" [6.41], "Estimate" (section 6.4.3.2), "Laarhoven" [6.42], "Optimal" [6.40], and "Tardos" [6.24]. With m = 768, c = 5,* $c_{\max} = 10$

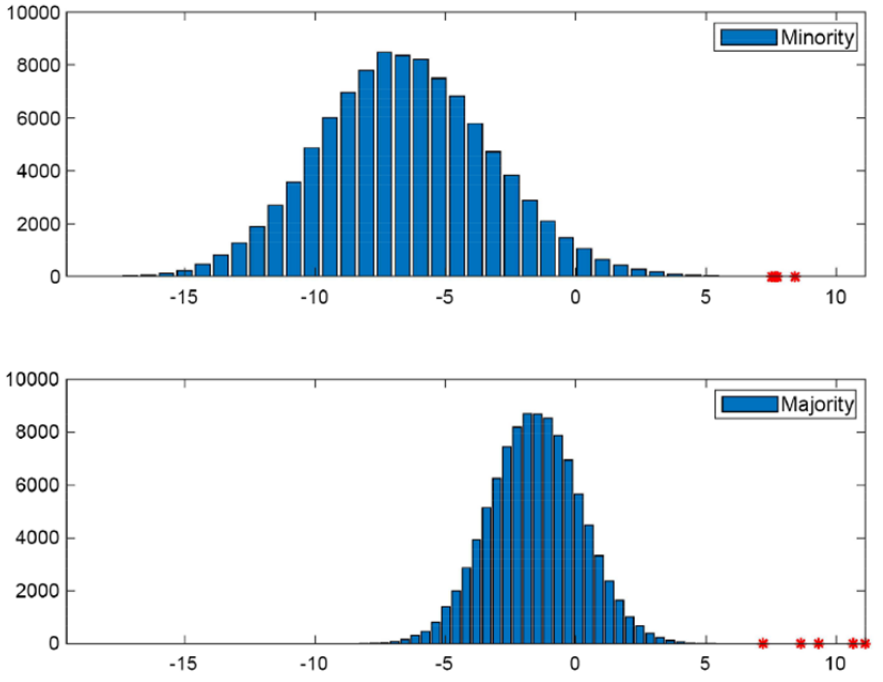


Figure 6.5. *Histograms of innocent people's scores for the "Laarhoven score function" [6.42] and two collusion strategies, "minority vote" and "majority vote", m = 768, c = 5, c_{max} = 10, n = 10^5. Traitor scores are displayed with red asterisks*

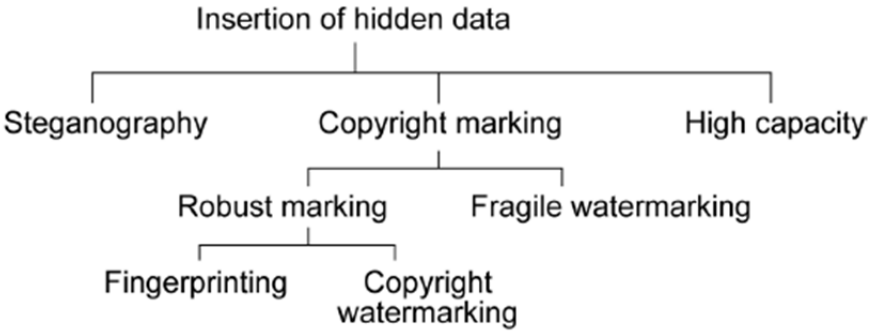


Figure 7.1. *Classification of data hiding methods, based on the work of Petitcolas* et al. *(1999) and Cheng and Wang (2007)*

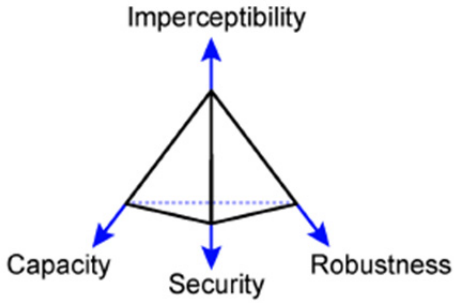


Figure 7.2. *Data hiding trade-off*

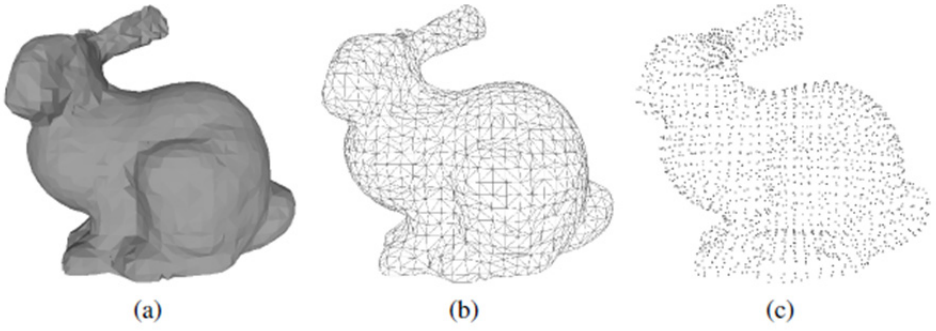


Figure 7.3. *The Stanford Bunny in low-resolution, 1,889 vertices and 3,851 faces, a) making the mesh of the 3D object, b) faces and c) point cloud*

Figure 7.4. *a) 3006 3D point cloud, and b) MST built on the point cloud*

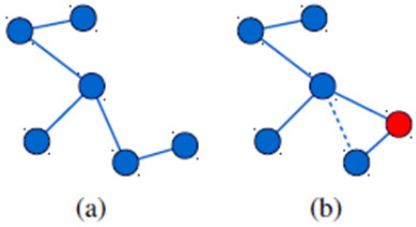


Figure 7.5. *The problem of sensitivity of EMSTs*

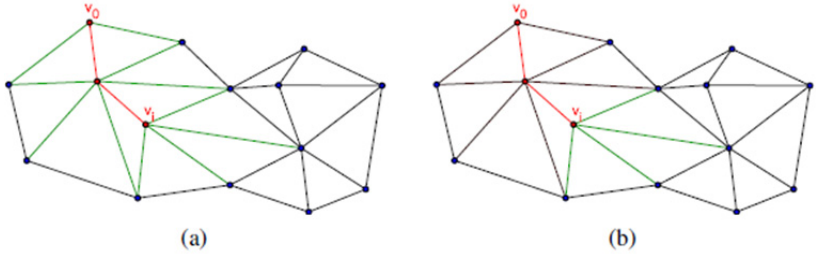


Figure 7.6. *Structure at the i stage. The vertices and edges in red are already covered, the edges in green are compared in order to add the next vertex, for a) the EMST built with Prim's algorithm, b) a Hamiltonian path*

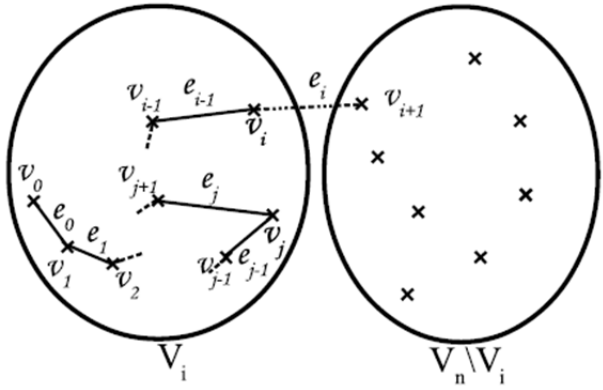


Figure 7.7. *State of the sets at step i: v_i the current vertex*

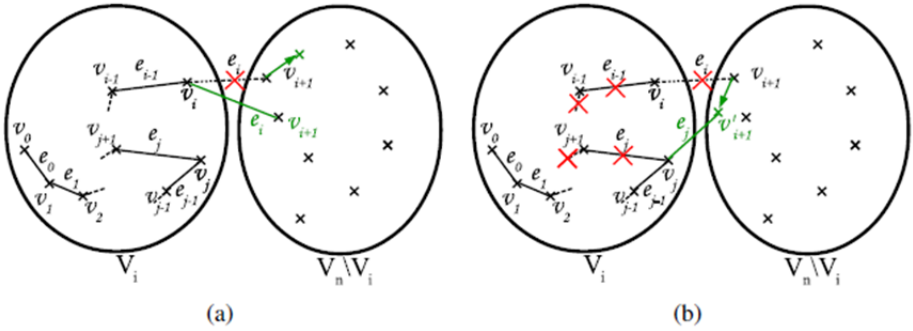


Figure 7.8. *a) v_{i+1} is moved too close to the sub-path, and b) v_{i+1} is placed too far from its predecessor*

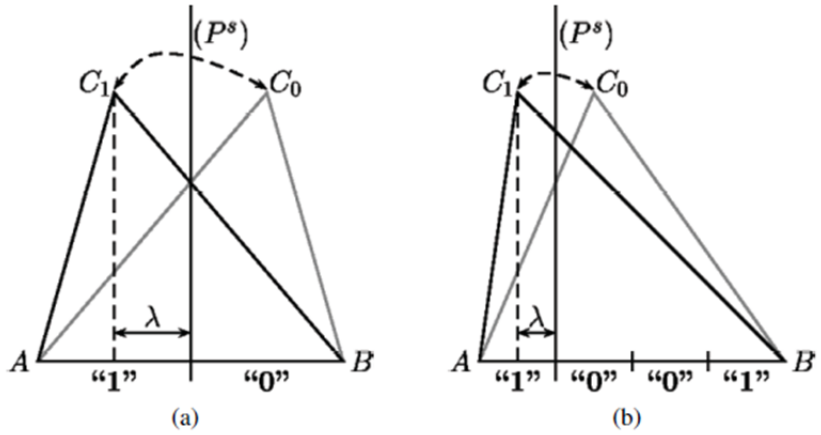


Figure 7.9. *Embedding method of the algorithm of Cayre and Macq (2003), the bit "1" is embedded by moving the vertex C_0 to the position C_1, so that its projection on the opposite edge corresponds to an interval that encodes the correct value. The opposite edge can be divided into a) two intervals and b) four intervals*

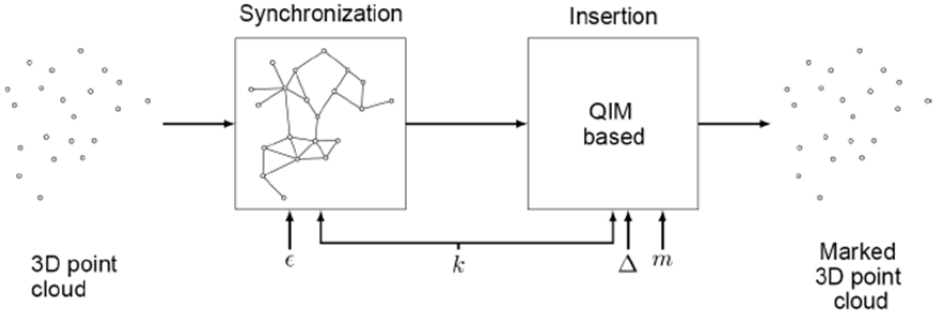


Figure 7.10. *Overview of a data hiding method in a point cloud in the spatial domain, where ϵ is the synchronization parameter, k is the secret key, Σ is the quantization step and m is the message to be embedded*

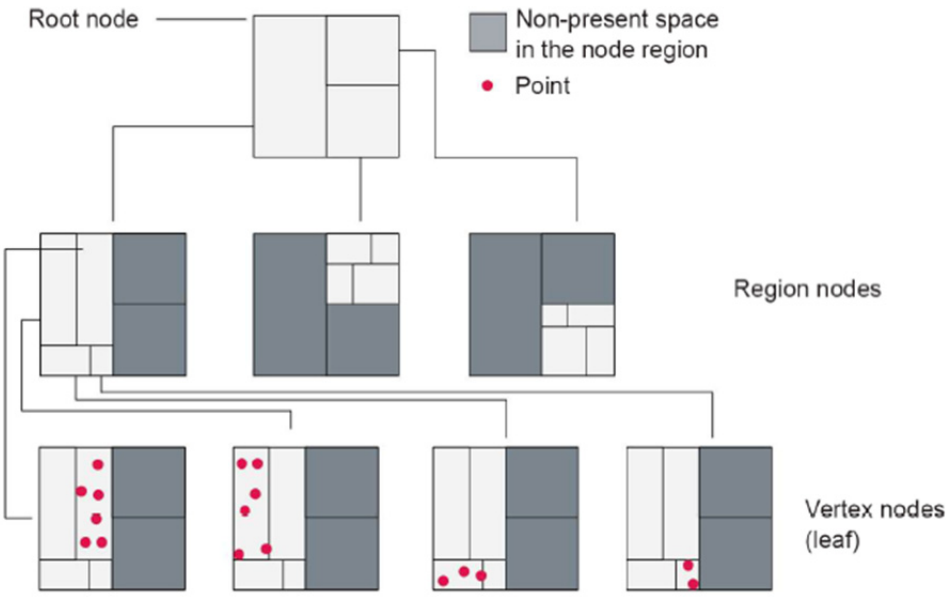


Figure 7.11. *Representation of a K-d-B-Tree*

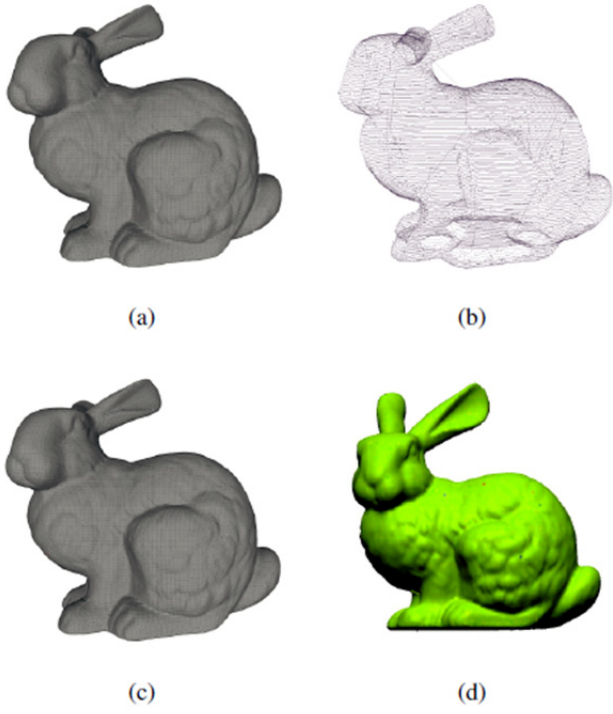


Figure 7.12. *Example of data hiding in a 3D Bunny object with $\Delta = 10^{-4}$: a) original 3D object, b) Hamiltonian path, c) marked 3D object and d) geometric distortions between the original 3D object and the marked one*

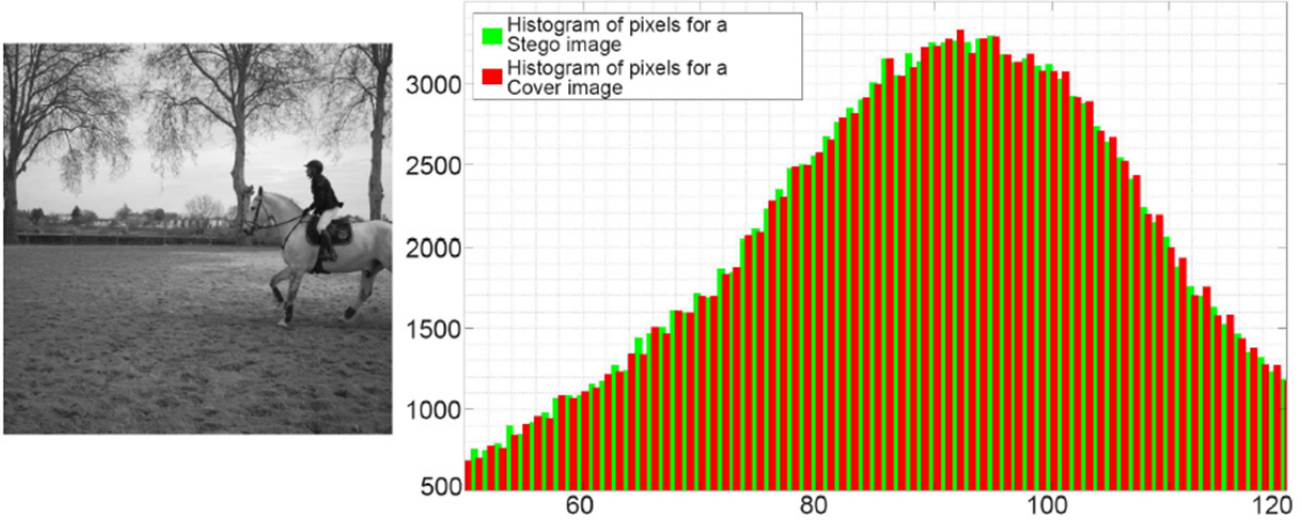


Figure 8.1. *Illustration of the impact of steganography by LSBR and its detection by the χ^2 test*

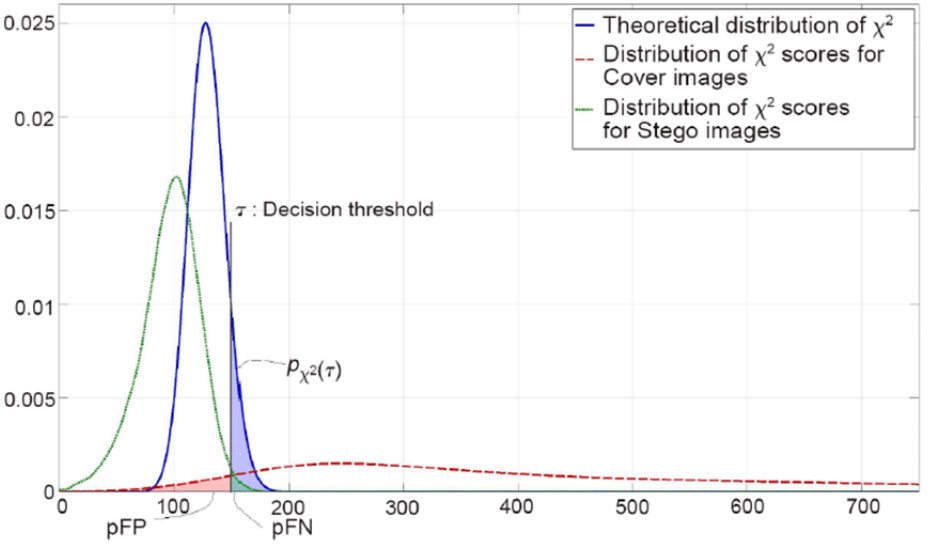
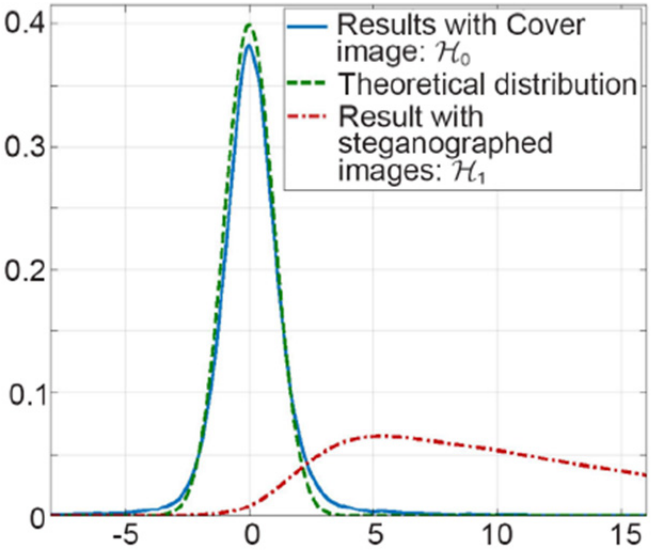


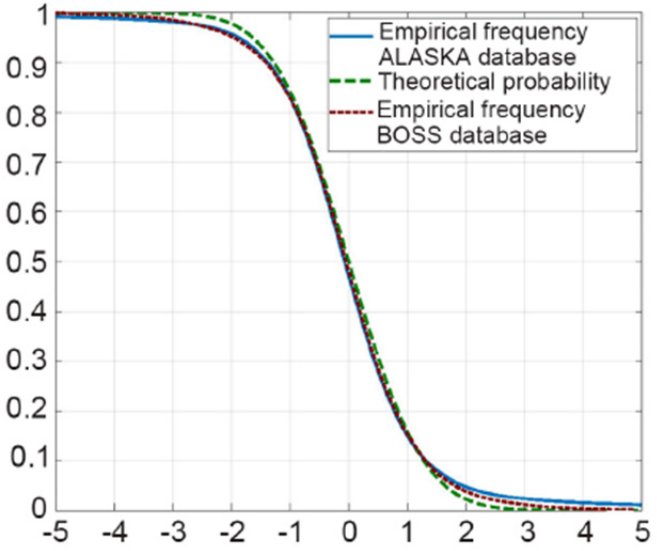
Figure 8.2. *Illustration of probability distributions (empirical and theoretical) for the result of the χ^2 test and the resulting error probabilities*

Result \ Truth	Hypothesis 0: (Cover image)	Hypothesis 1: (Stego image)
Accept Hypothesis 0	Correct decision	False-negative (missed detection) (pFN: $1-\varsigma$)
Accept Hypothesis 1	False-positive (false alarm) (pFP: $\alpha=\mathbb{P}\,[\delta(\mathbf{X})=\mathcal{H}_1\,|\,\mathcal{H}_0]$)	Correct decision ($\varsigma=\mathbb{P}\,[\delta(\mathbf{X})=\mathcal{H}_1\,|\,\mathcal{H}_1]$)

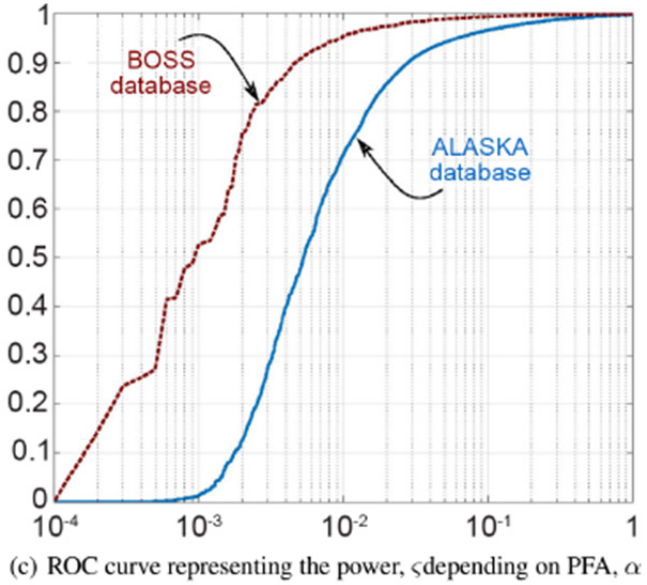
Table 8.1. *The different possibilities of good and bad detection*



(a) Distribution of LR-log for Cover and Stego images



(b) Probability of false positive (PFA), α, depending on the decision threshold ; theory and empirical results



(c) ROC curve representing the power, ς depending on PFA, α

Figure 8.3. *Results of the application of the LRG test, equation [8.14]*

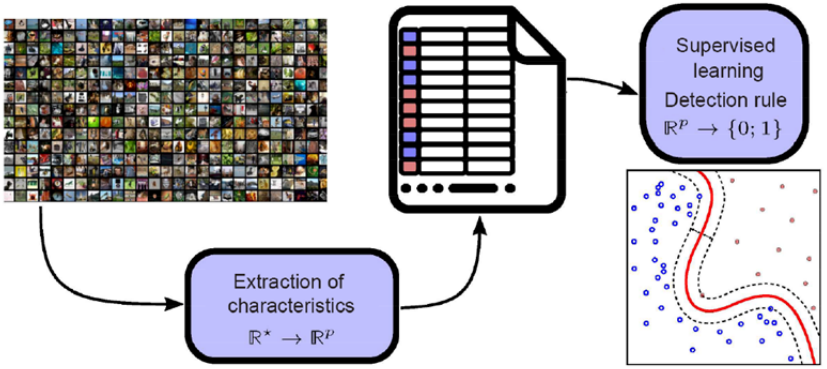


Figure 8.4. *Illustration of the supervised learning principal, which aims to determine a detection rule from a labeled database (right image licensed under CC BY-SA 4.0, produced by Zirguezi)*

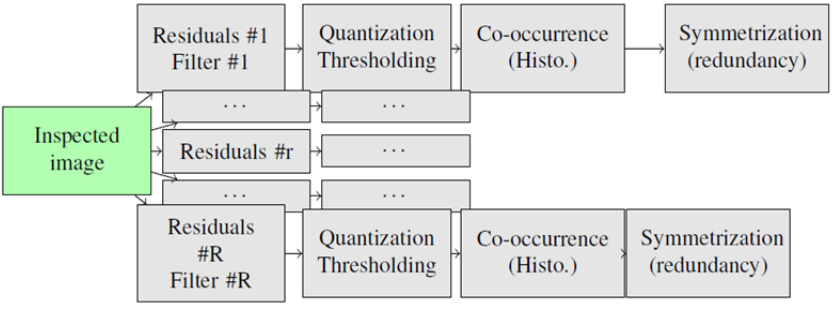


Figure 8.5. *Illustration of the principle of extraction of RM characteristics (Spatial Rich Model)*

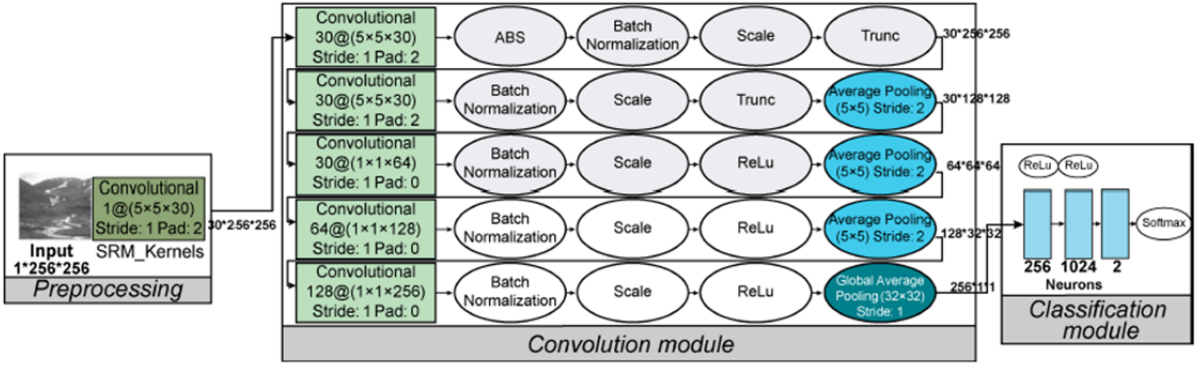


Figure 8.6. *The Yedroudj-Net (Yedroudj* et al. *2018b) network*