06.04.23

Types of modulation

- Robust DWM
- Steganography
- Tamper detection

$\underline{m} = (0101\ldots)^T$ $\begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}$

① $\underline{x} \rightarrow$ Enc $f(\cdot)$ $\rightarrow \underline{y}$; $\underline{K} = (100101)^T$

$D_{xy} = \|\underline{x} - \underline{y}\|_2^2 \leq D_{emb}$

$= (\underline{x}-\underline{y})^T(\underline{x}-\underline{y})$

min

- Note: l_2-norm $\neq$ HVS;

φ-mask

$\varphi = H$: 1 → wm visible; 0 → wm is invisible

$\|\underline{x} - \underline{y}\|_\varphi^2 = (\underline{x}-\underline{y})^T \varphi^{-1} (\underline{x}-\underline{y}) = \sum_{i=1}^{H}\sum_{j=1}^{W} \varphi(i,j) \cdot (x(i,j) - y(i,j))^2$

$\underline{y} = f_{\underline{K}}(\underline{x}, \underline{m})$ — modulation or WM embedding function.

② Types of $f_K(\cdot)$ —
- (1) Direct domain $\underline{y} = f_{\underline{K}}(\underline{x}, \underline{m})$
- (2) Transform domain: T, T^{-1}, s.t: $T\,T^{-1} = I$
  - (a) $\tilde{\underline{x}} = T(\underline{x}) = T\underline{x}$ (direct)
  - (b) $\tilde{\underline{y}} = f_K(\tilde{\underline{x}}, m)$ (embedding)
  - (c) $\underline{y} = T^{-1}(\tilde{\underline{y}}) = T^{-1}\underline{y}$ (inverse)

Hand-crafted transform (defined by humans)

- non-learnable (not ML)
- data independent (security)

(a) T → $\tilde{x}$ → (b) $f_K(\tilde{x}, m)$ → $\tilde{y}$ → (c) T^{-1}

- DCT
- DFT
- DWT
- RP (key → ...) DP DP ...

public / key-based

min D_{xy}

T: $f_\phi(\cdot)$

T^{-1}: $g_\theta(\cdot)$

▷ fixed transforms

- data agnostic → $\perp\!\!\!\perp \{\underline{x}\}$; indep.

Machine-learning based transform domain modulations

$\underline{x}$ → Neural Network $f_\phi(\underline{x}) = \tilde{\underline{x}}$ → $\tilde{x} \in C$ → $f_K(\tilde{x}, m)$ → $\tilde{y}$ → $g_\theta(\tilde{y})$ → $\underline{y}$

data $\{\underline{x}\}$

min D_{xy}

$\{\underline{x}_i\}_{i=1}^N$ → $f_\phi(x)$, $g_\theta(\tilde{y})$

Training data

- domain specific:
  - medical
  - astro
  - biometrics
  - ⋮

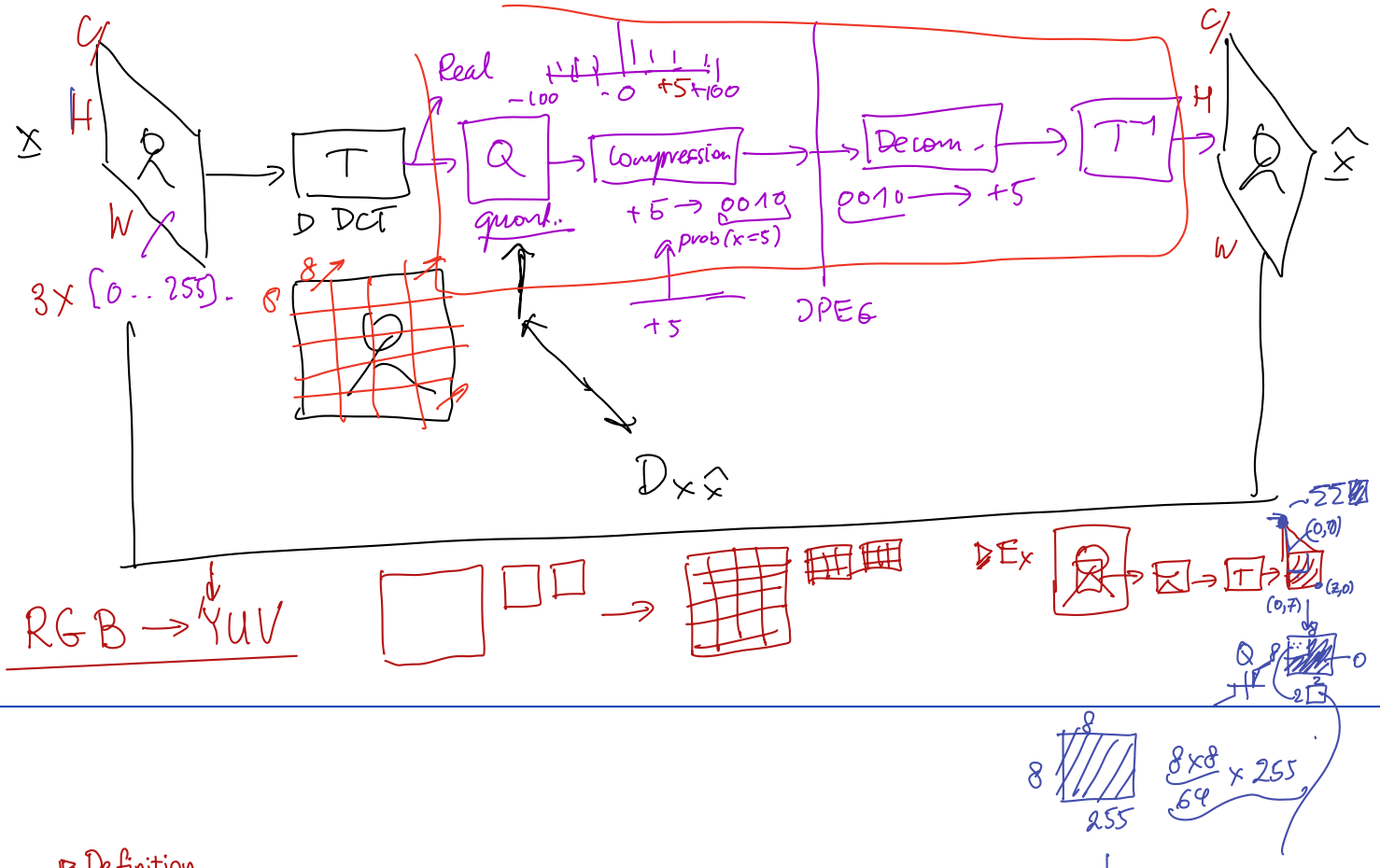
→ ML(T) ↑

- generic

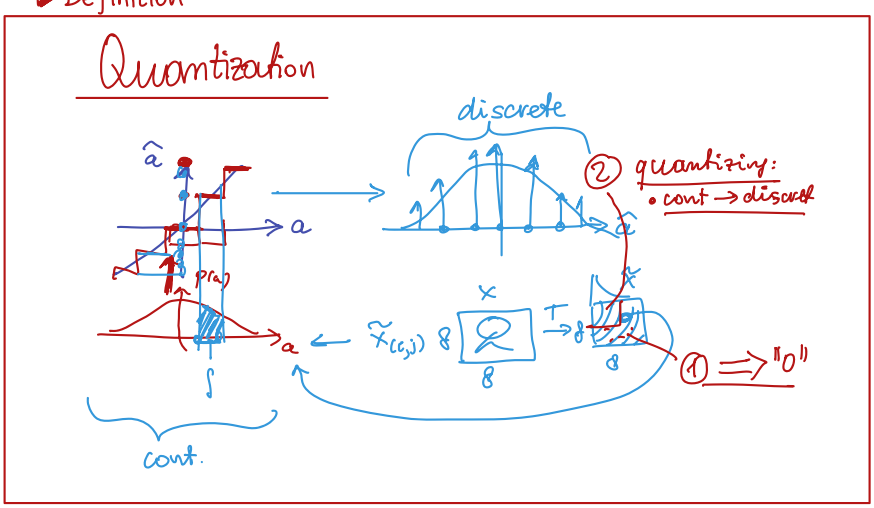
$T_{HC} \approx T_{ML}$

▷ This transform is learned in a function of data $\{\underline{x}\}$.

Link to lossy (JPEG) image compression



▷ Definition



▷ $f_K(\tilde{x}, m) = Q_K(\tilde{x}, m)$ - quantization based embedding.

• Dither quantizer
$Q(\tilde{x} + d)$
$\Pi \leftarrow (K, m)$

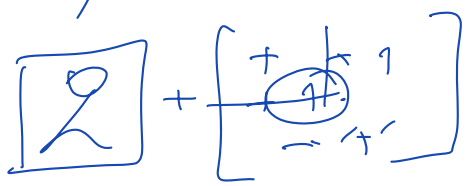
~~+0|-0|0|-0~~

Recall:

$\underline{y} = \underline{x} + \underline{w}$ $\quad \underline{m}, \underline{K}$

in the direct /image/ domain.

- direct domain ($T=I$)
- additive modulation
- quantization modulation



$\underline{m} = [0\ 1\ 0\ 1\ \dots\ L]$

$\underline{K} \rightarrow (x_1, y_1)$ $\quad 0 \rightarrow -1$, $1 \rightarrow +1$

Table: general classification of DM modulation techniques.

$\underline{y} = f_{\underline{K}}(\underline{x}, \underline{m})$

- Direct domain (I) $= T$
  - Additive modulation
  - Quantization modulation
- Transform domain (T)
  - HC ($T \neq X$)
    - Additive modulation
    - Quantization modulation
  - ML ($T \neq X$)
    - Additive modulation
    - Quantization modulation

$f_K\left(\tilde{\underline{x}}, \underline{m}\right) \leftarrow \tilde{\underline{x}} = T(\underline{x}) \leftarrow \underline{x}$

$\rightarrow T^{-1}(\cdot) \rightarrow \underline{y}$

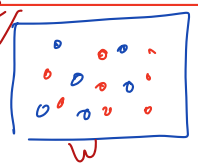
$\underline{y} = \underline{x} + \underline{w}$ $\quad \underline{m}, \underline{K}$

1) Additive modulation.

1.1. Direct domain:

$\underline{y} = f_{\underline{K}}(\underline{x}, \underline{m}) = \underline{x} + \underline{\gamma} \odot \underline{w}$

masking, $\underline{w}(\underline{m}, \underline{K})$



$\underline{w}(\underline{m}, \underline{K})$

- $y = x + w$ $\quad = x + w$
- $y = T^{-1}\left(T(x + w)\right)$ $\quad I$

$= T^{-1}\left(T(x) + T(w)\right) = T^{-1}T(x) + T^{-1}T(w)$

$\underline{y} = T^{-1}\left[T(\underline{x})\right] = \underline{x} + T^{-1}\left(\tilde{\underline{\gamma}} \odot \tilde{\underline{w}}\right)$

$\tilde{\underline{x}}$, T^{-1}

- $\bullet$ — $+1$
- $\circ$ — -1

1.2. Transform domain:

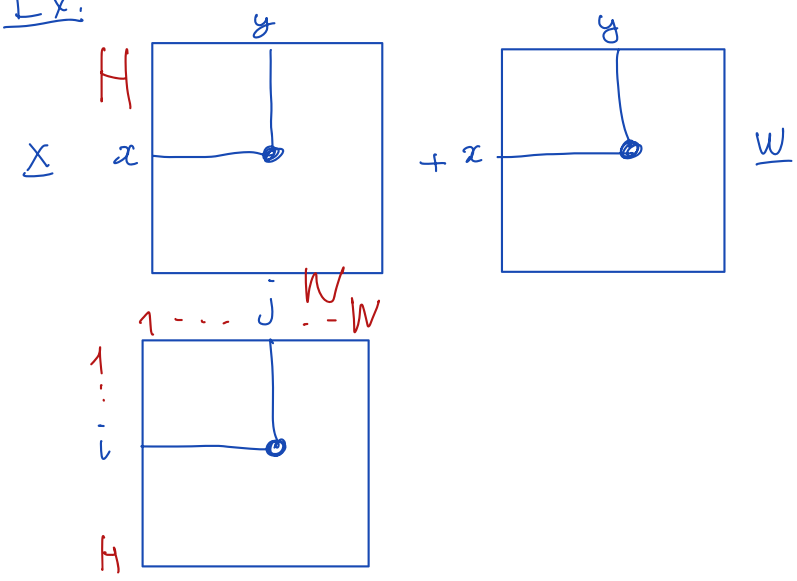
$\underline{y} = f_{\underline{K}}(\underline{x}, \underline{m}) = T^{-1}\left(T(\underline{x}) + \tilde{\underline{\gamma}} \odot \tilde{\underline{w}}\right)$

HC ML

$\underline{w}(\underline{m}, \underline{K})$

- $T(aX + bY) = aT(X) + bT(Y)$

▷ Ex:



$$\underline{y} = \underline{x} + \underline{w}$$

$$y[i,j] = x[i,j] + w[i,j]$$

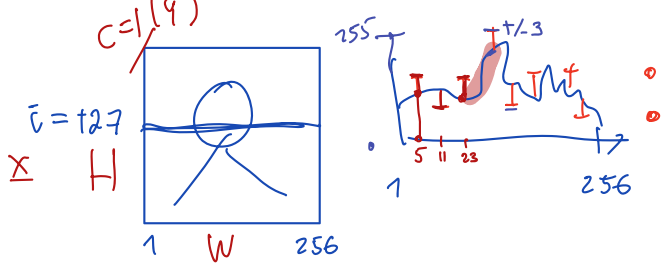
$x[i,j] \in [0 \dots 255]$; $w[i,j] \in \{\pm 1, \pm 3\}$

$\gamma \cdot \varphi[i,j]$; $\varphi[i,j] \in [0 \dots 1)$

$\gamma > 1$; ± 3 ; ± 7

① Embedding (encoding) $f_K(\underline{x}, \underline{m})$

φ < JND (Meta) / NVF (Unifa)



- position $\equiv \underline{K} \Rightarrow (5, 11, 23 \dots)$
- sign $\equiv \underline{m}$. 0 → −1, 1 → +1

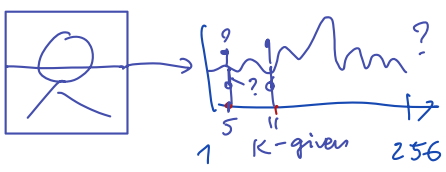
1 0 1 1 01 1 0

γ=3 ||(+1)(−1)+1+1

$y = x + w \odot \varphi$

② Extraction (decoding)

⚠ The extraction of WM is NOT a trivial problem.

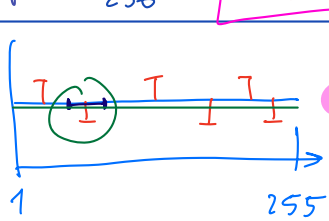


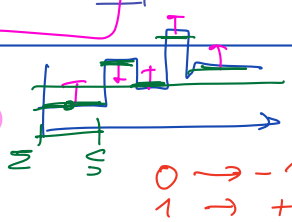
Defider (defender) / Attacker.

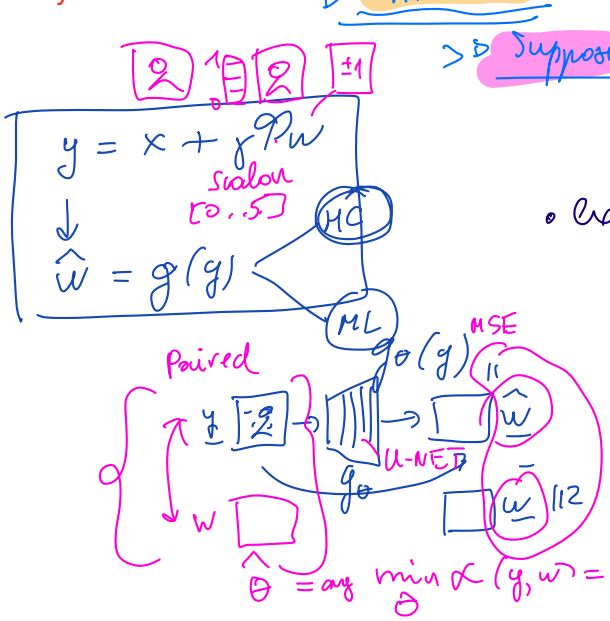
Why?

▷ Intuition.

▷▷ Suppose:

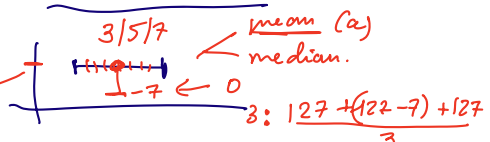






• Extraction on the window → "local mean".

Local mean (a)



$y = x + \frac{w}{\gamma}$

$\hat{x} = \frac{1}{3}\sum_{i=1}^{3} x_i$, $\frac{3 \times 127}{3} - \frac{7}{3} = \hat{x}$

$\hat{w} = y - \hat{x} = (127 - 7) - (127 - \frac{7}{3}) = -7 + \frac{7}{3} = -5 \rightarrow -1$

Local media. (5)

Watermark w faces a strong interference from the side of host image x.

1) $y = x + w$

2) $\hat{x} = pred(y)$

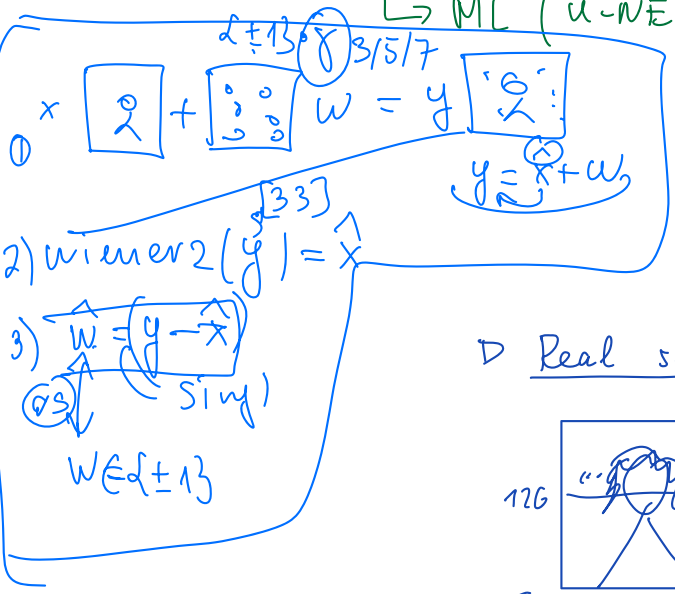
↳ denoiser

↳ wiener2 $= \bar{y} + \frac{\sigma_x^2}{\sigma_x^2 + \sigma_w^2}(y - \bar{y})$

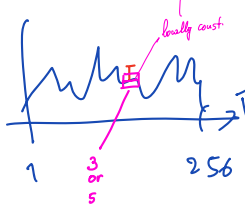
↳ $\bar{y}$ - mean value

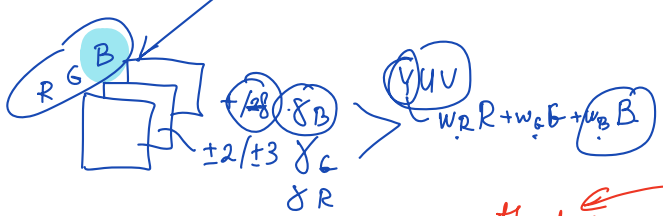
↳ median

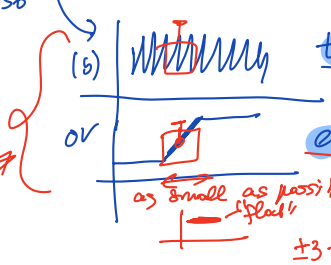
↳ ML (U-NET)



▷ Real signals







attacker cannot do the accurate prediction.

Security

*

▷ In case of image:

$\underline{y} = \underline{x} + \underbrace{\underline{w}}_{[m,K]} = \underline{x} + \underbrace{\underline{\epsilon}}_{\text{noise.}}$

O[JP]: denoiser: $w = \epsilon$ as noise.

a) $\hat{\underline{x}} = \text{denoiser}(\underline{y})$ ~~$\underline{y}$~~ ~~✗~~

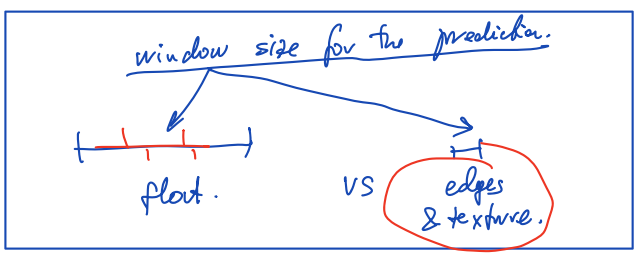
Matlab: wiener2($\underline{y}$, [3 3], σ_ϵ^2 auto.) 5×5 3×3

b) $\hat{\underline{w}} = \underline{y} - \hat{\underline{x}} = \underline{x} + \underline{w} - \hat{\underline{x}}$

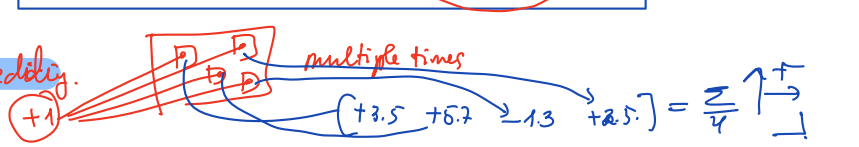
5.1. if $\hat{\underline{x}} = \underline{x}$: $\hat{\underline{w}} = \underline{w}$.

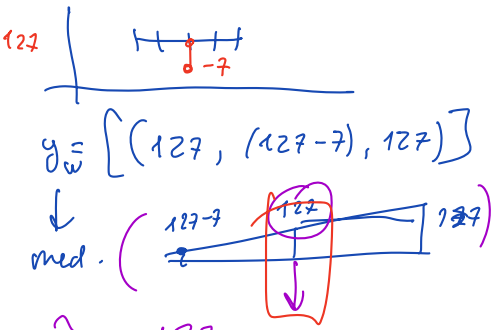
6.2: if $\hat{\underline{x}} = \underline{x} + \underline{z}_{est}$;

$\hat{\underline{w}} = \underline{y} - \hat{\underline{x}} = \underline{x} + \underline{w} - \underline{x} - \underline{z}_{est} = \underline{w} - \underline{z}_{est}$.



→ repetive embedding one bit

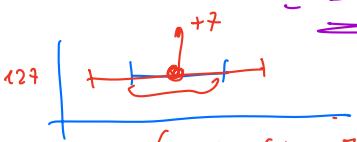




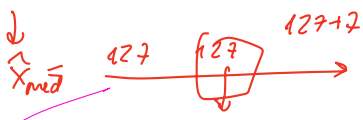
|| local mean → local median

$\hat{x}_{med} = 127.$

$\hat{w} = y - \hat{x} = (127-7) - 127 = -7.$

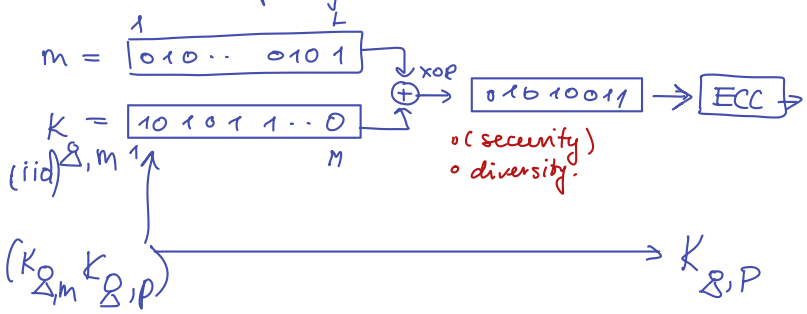


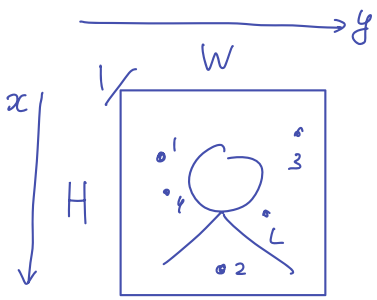
$y_w = (127, (127+7), 127)$



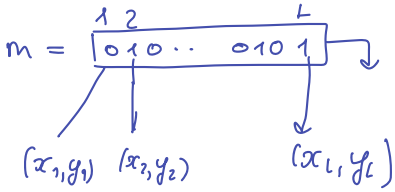
$\hat{w} = y - \hat{x}_{med} = (127+7) - 127 = +7.$

Recap of DWM





Allocation of bits



" 1 bit = 1 position "

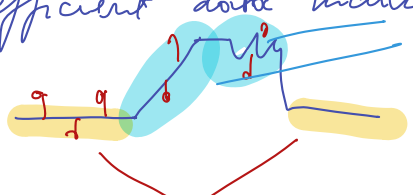
* Note: H x W vs L

3k x 4k → 12 Mio

64 or 128 bits.

⊖ • H x W "redundancy" is not used for efficient data hiding.

⊖ •



ok → predict "bits"!

▷ Solution.

① ECC

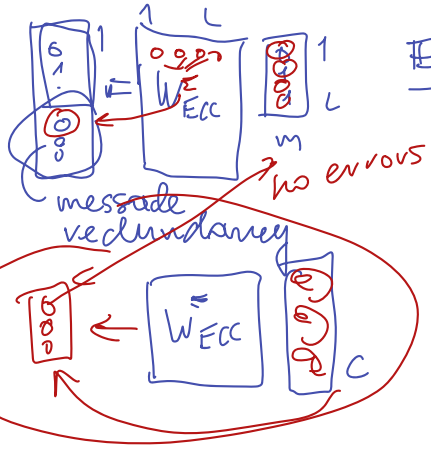
② - w ECC (R = 1/0) → complexity

- w/o ECC → repetion.

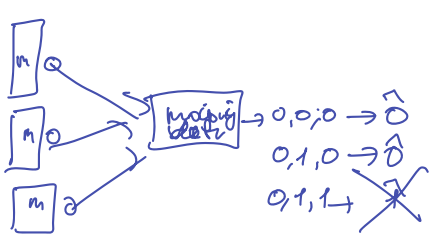
m = 0101 → ECC →

- Turbo → Polar
- RS
- BCH

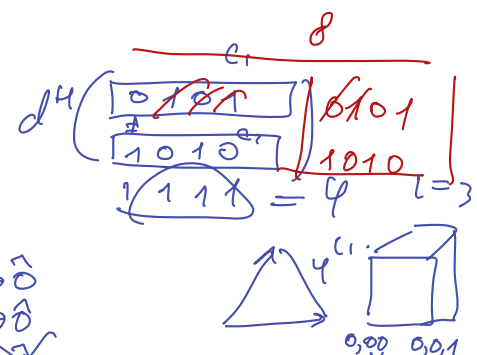
⊆



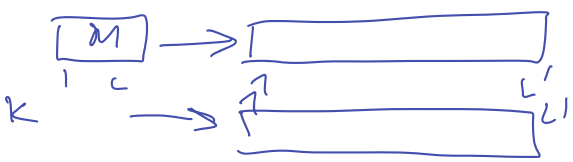
Ex:



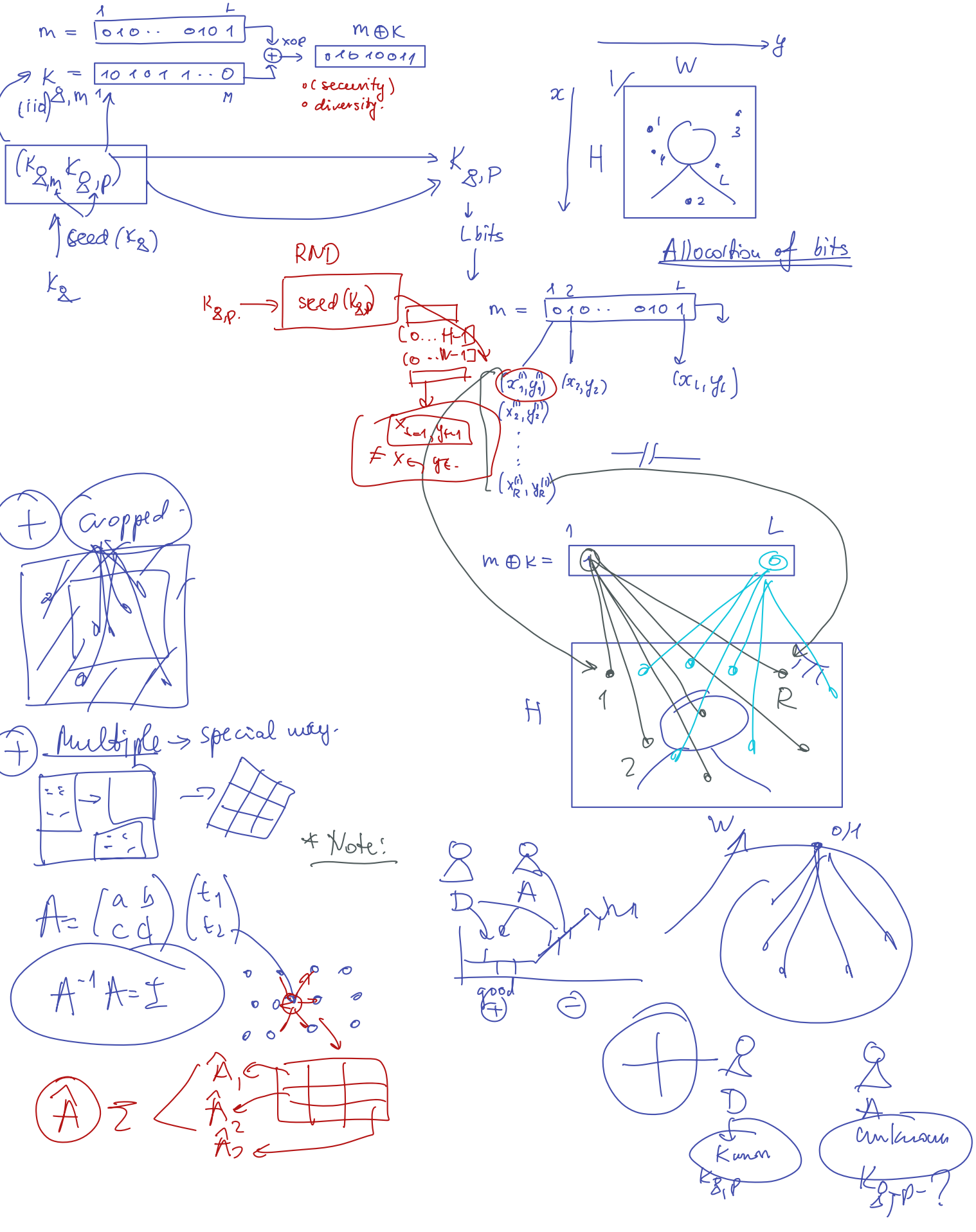
Rank 1/3:



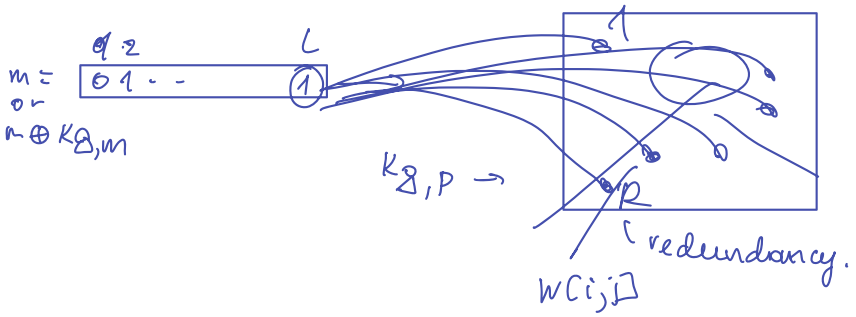
Repetitive codes → not efficient



w/o ECC. : "1 bit ⇒ multiple positions!!



Message decoding via aggregation.



▷ "Place" a bit?

1) Modulation

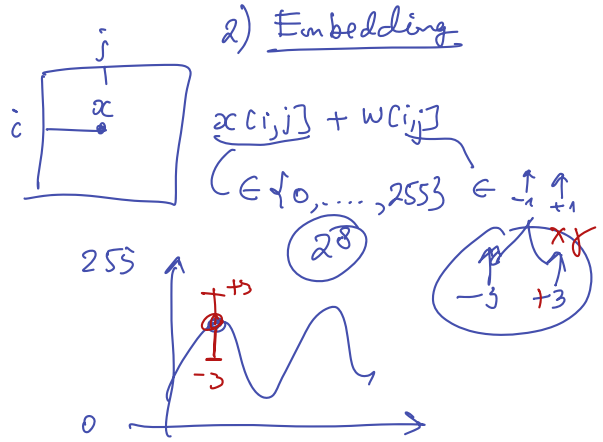
$\{m_i\}_{i=1}^{L}$ $0 \rightarrow -1$, $1 \rightarrow +1$

$c_i = 2 \cdot m_i - 1$

$m_i = 1: \; 2 \cdot 1 - 1 = 1$

$m_i = 0: \; 2 \cdot 0 - 1 = -1$

2) Embedding

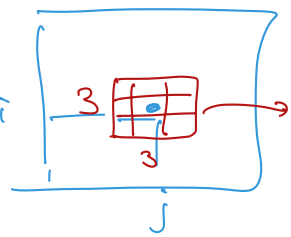


▷ Decoding:

① Bit extraction

$K_{S,P} \rightarrow (i,j)$

$y = i$



$x[i,j] + w[i,j]$

$+/- 3$

0 — 255

② Denoising:

$\hat{x}[i,j] = C_{wiener2}[y[i,j]]$

↳ $x[i,j] + w[i,j]$

③ Prediction:

$\hat{w}[i,j] = y[i,j] - \hat{x}[i,j]$

$= x[i,j] + w[i,j] - \hat{x}[i,j]$

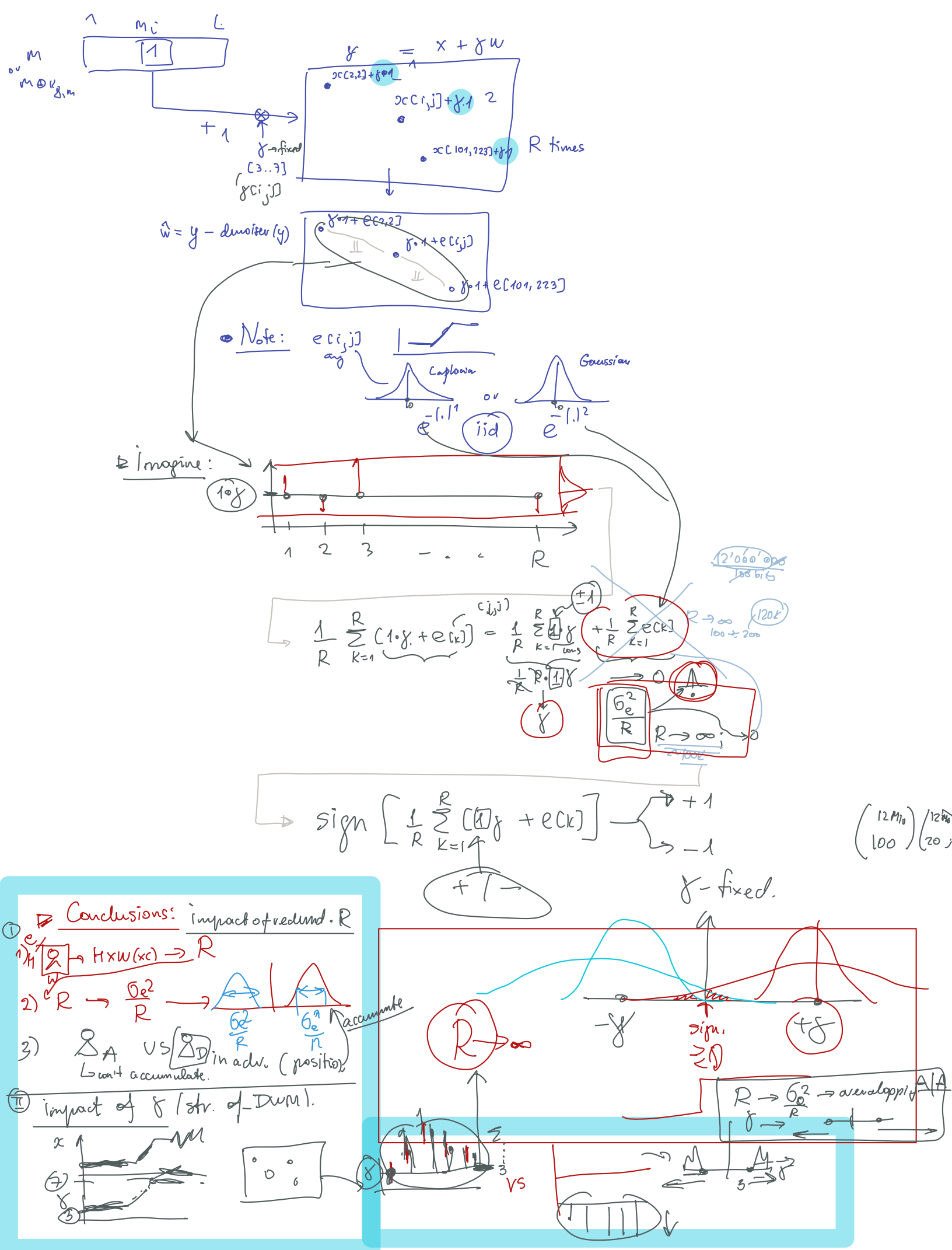
(a) if $\hat{x}[i,j] = x[i,j]$:

$\hat{w}[i,j] = w[i,j]$

(b) → real: $\hat{x}[i,j] = x[i,j] + e[i,j]$

float: $e[i,j] \rightarrow 0$

est: $e[i,j] \rightarrow$ 15.21 ≥ |w| ✗


y = x + γw
R times
ŵ = y − denoiser(y)
Note:
Laplacian
Gaussian
iid
Imagine:
1 2 3 . . . R
sign
+1
−1
γ - fixed.
Conclusions:
impact of redund. R
impact of γ (str. of Dum).
accumulate
VS

Sample mean

- Consider a sample mean of N independent and identically distributed random variables $X_1, X_2, \cdots, X_N$ with $E[X_i] = \mu_X$ and $Var[X_i] = \sigma_X^2$
- The **sample mean** is

$$M_N(X) = \frac{S_N(X)}{N} = \frac{X_1 + X_2 + \cdots + X_N}{N} = \frac{1}{N}\sum_{i=1}^{N} X_i$$

- The moments of **sample mean**
  - **Expected value:** $E[M_N(X)] = E[X] = \mu_X$
  - **Variance:** $Var[M_N(X)] = \frac{Var[X]}{N} = \frac{\sigma_X^2}{N}$

Sample mean: moments

- **Proof**
- **Expected value**: sampling mean is an unbiased estimator

$$E[M_N(X)] = \frac{1}{N}(E[X_1] + \cdots + E[X_N]) = \frac{1}{N}(E[X] + \cdots + E[X]) = E[X] = \mu_X$$

- **Variance**: suppose that $Var[X_i] = \sigma_X^2$

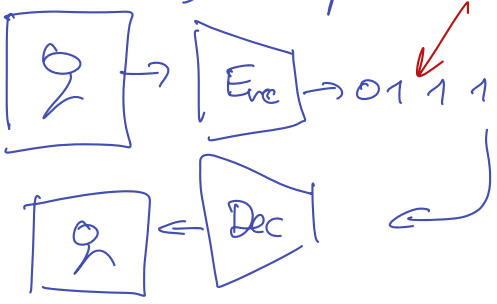
$$Var[M_N(X)] = \frac{1}{N^2}(Var[X_1] + \cdots + Var[X_N]) = \frac{1}{N^2}(Var[X] + \cdots + Var[X]) = \frac{1}{N}Var[X] = \frac{\sigma_X^2}{N}$$

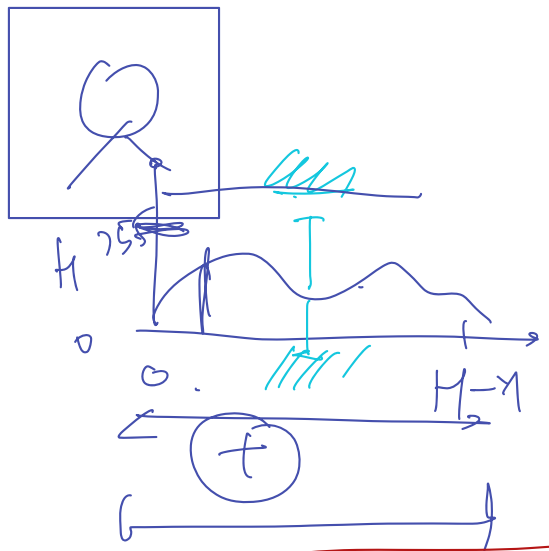
The variance of sample mean approaches zero as the number of samples increases (a.k.a. efficient estimator)

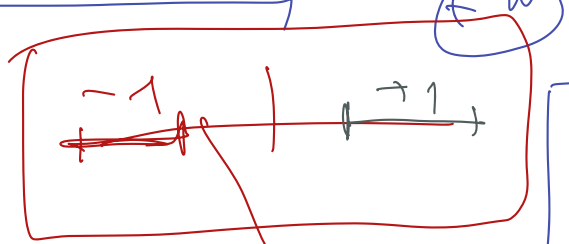
1) "Pixel" domain

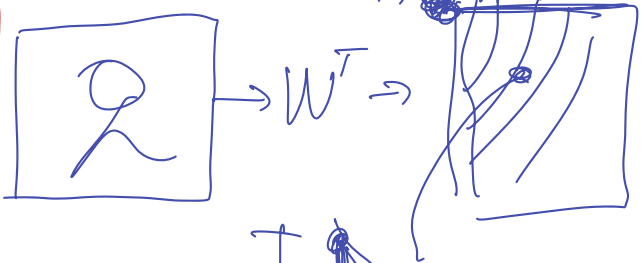
2) → Complexity
→ Compliancy with compression
→ Visibility

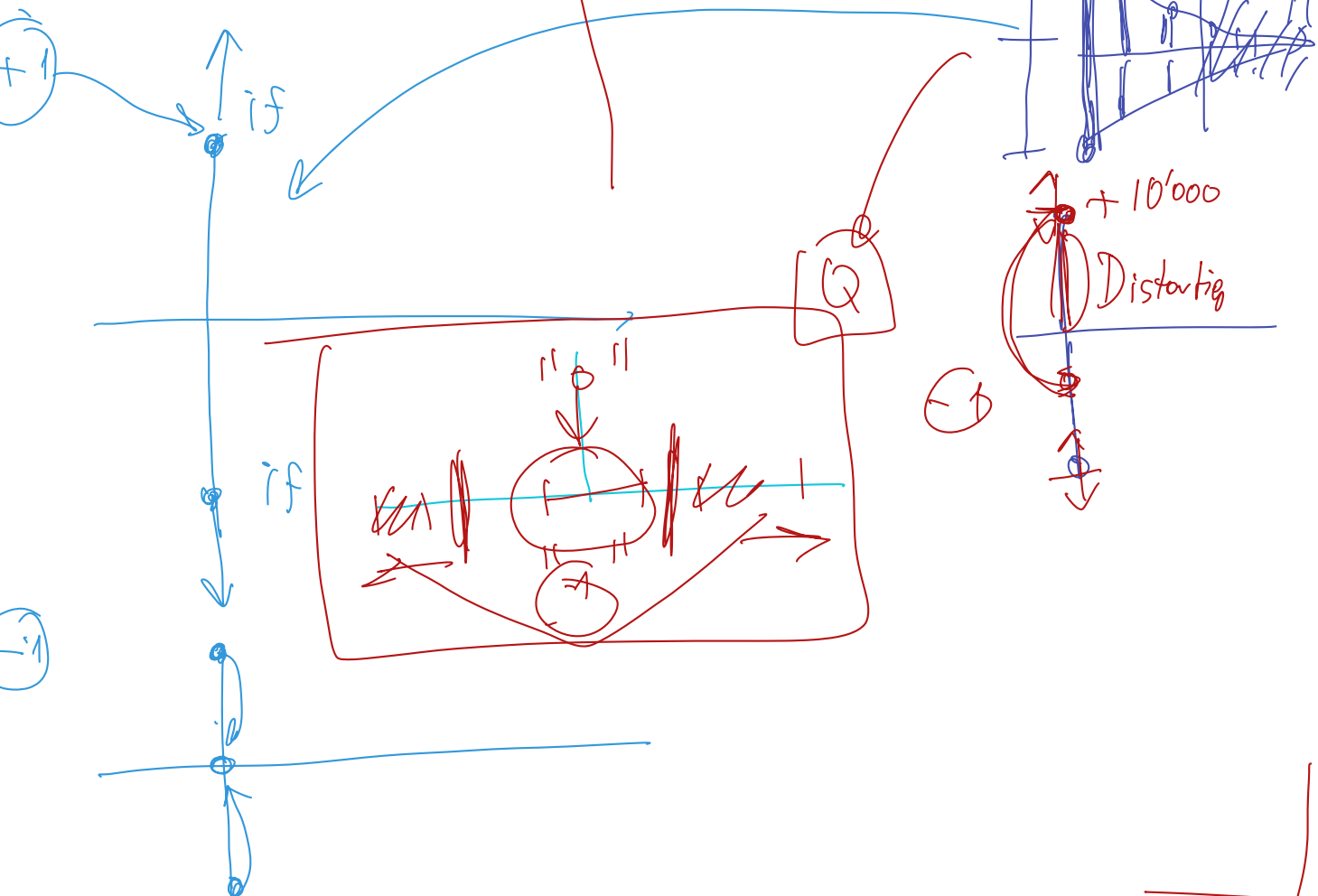
↳ "Compressed" domain











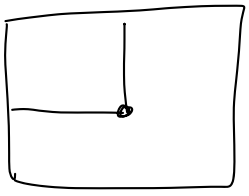
"Distributed" watermark embedding into multiple pixels: the redundancy again

$\underline{m} = [0\ 1\ 0\ \ldots\ 1]$ bits. (1, 2, L)

① $0 \to -1$, $1 \to +1$ → $[-1\ +1\ -1\ \ldots\ +1]$ bits.

②

Key $\underline{k}$ →

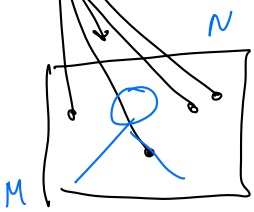


(a) $(x_1, y_1)\ (x_2, y_2)\ \ldots\ (x_L, y_L)$.

(no redundancy).

1 bit ≡ 1 location. (L)

(b). $[-1\ +1\ +1\ \ldots\ +1]$ bits (1, 2, L)



$(x_1^{(1)}, y_1^{(1)})\ (x_2^{(1)}, y_1^{(1)})\ \ldots$

R = # of repetitions.

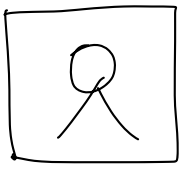
$R \times L$

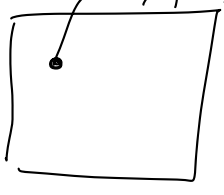
(label A)

RL positions

Attacker

$\underline{y}$





+/−1 − ? or ∅ (not modified)

- has no info where the same bit was placed.

$[+1\ \ldots\ \ldots\ ]$

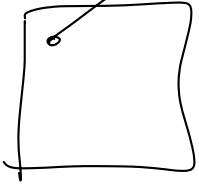
R

? → Σ → +/− → CLT → −1 / +1

$P_r = 0.5$ → +1 / −1

Defender

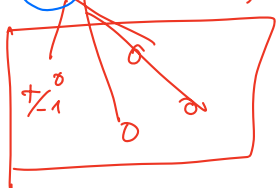
$\underline{y}$



+/−1 − ? or ∅

▷ has an info about "R times" location.

$[+1\ \ldots\ \ldots\ ]$



locations.

① ∅ → is not active.

(+/−1)

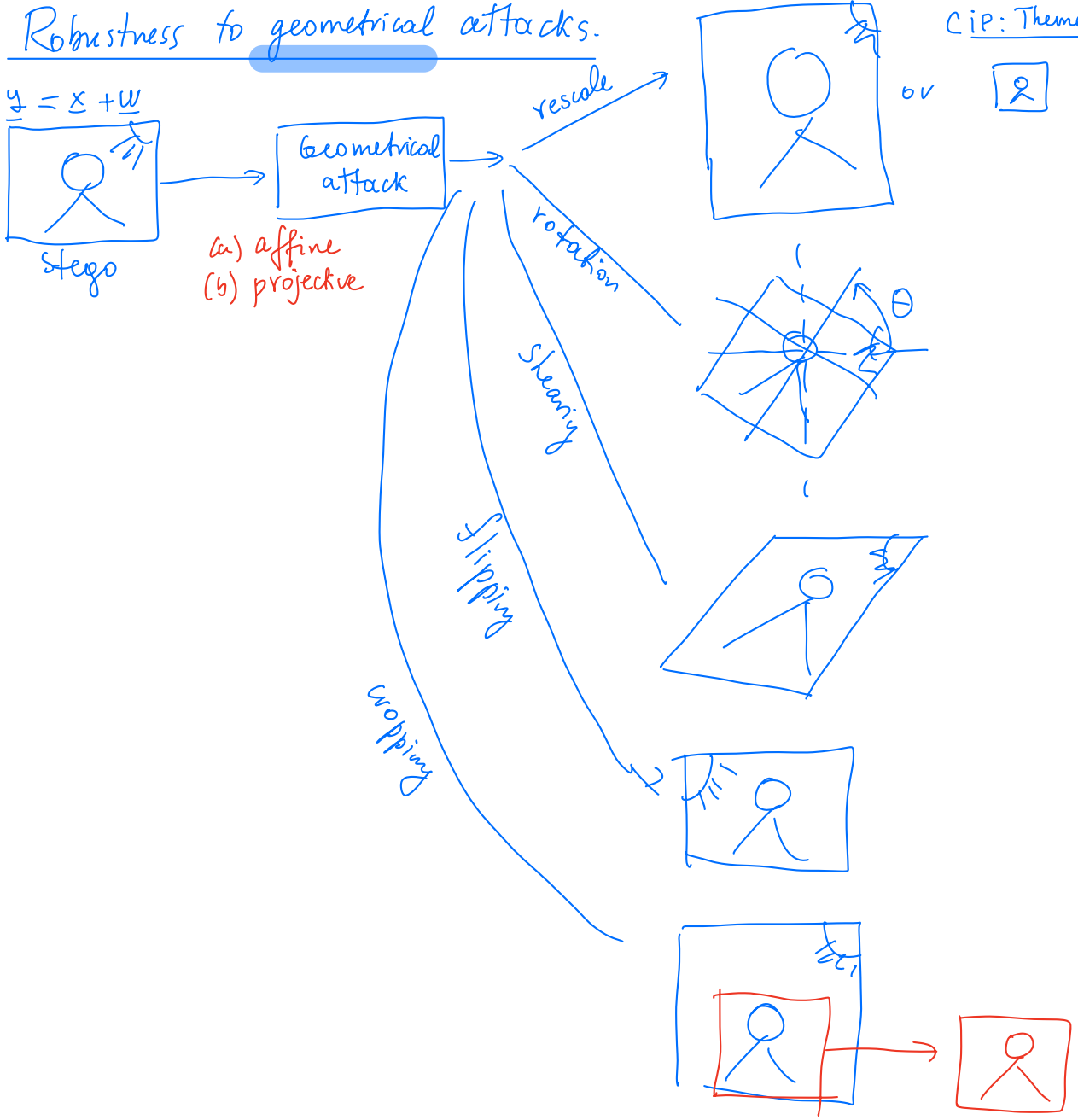
② R → Σ → ≷ → +1 / −1

Σ: $+0.2$, $+1.3$, $+0.75$, $+3.5$, -0.1, ⋮ (R)

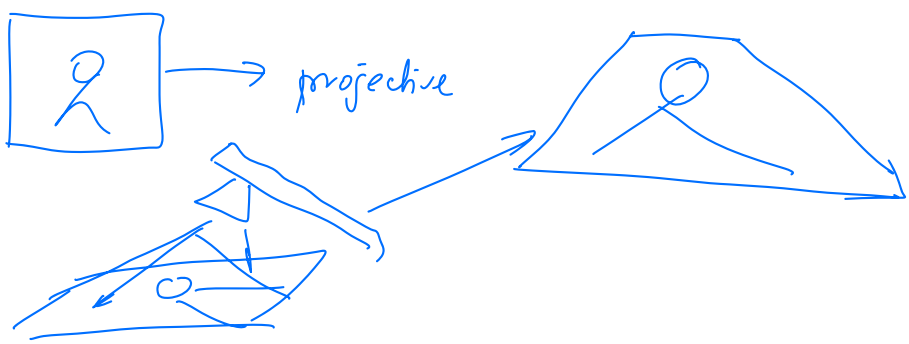
Robustness to geometrical attacks.

CIP: Theme 5

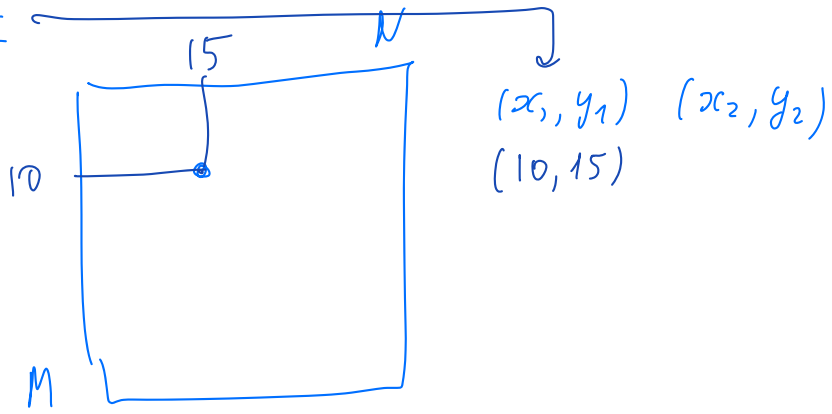
- given



Practical use case:

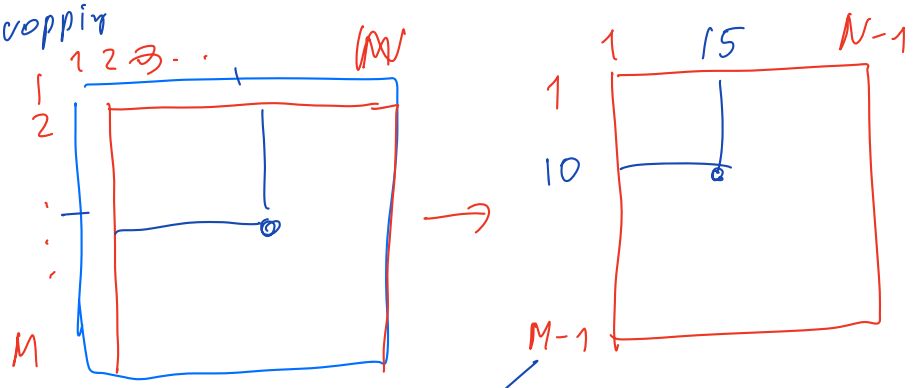


① K.



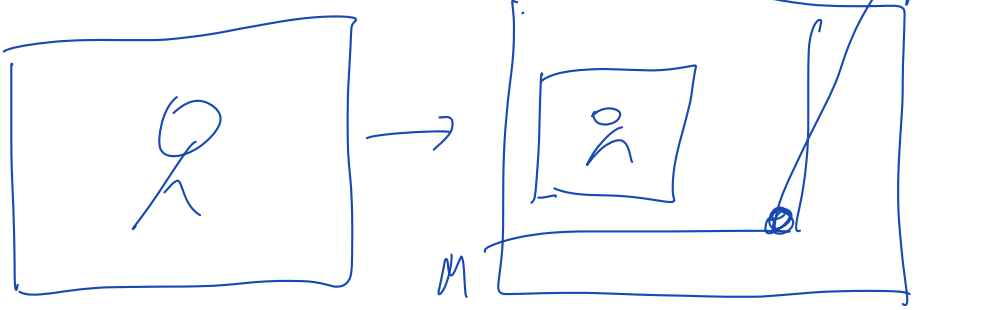
② Extraction

(a) Cropping

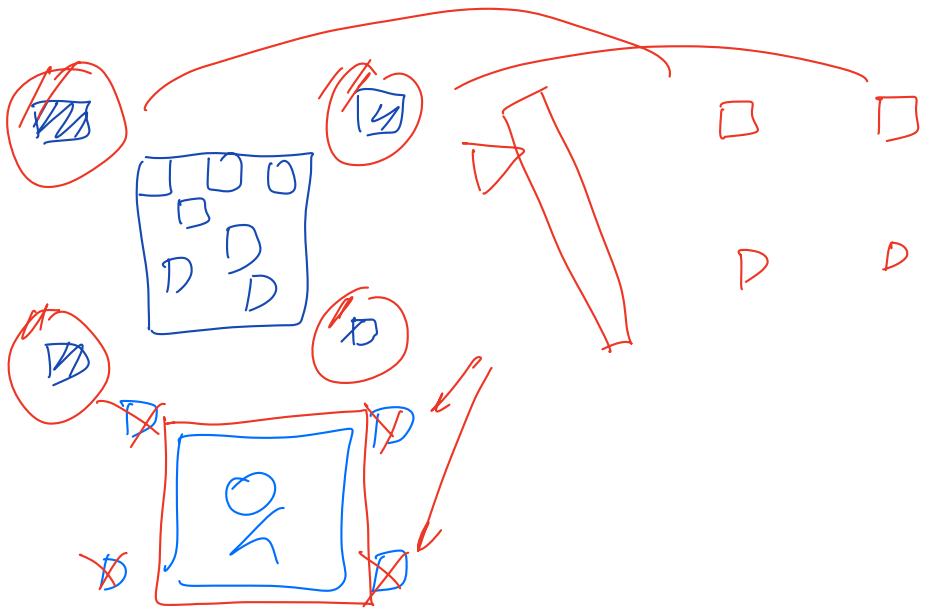


K ⟶ (x_1, y_1) (x_2, y_2) - - - -

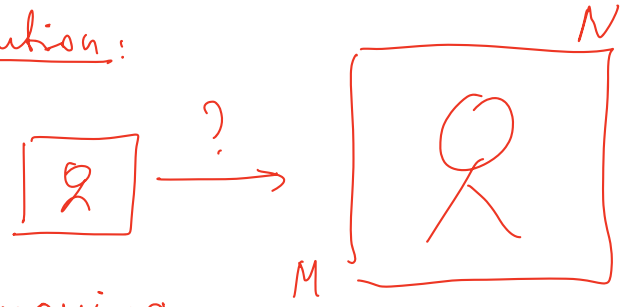
(b)



QR codes

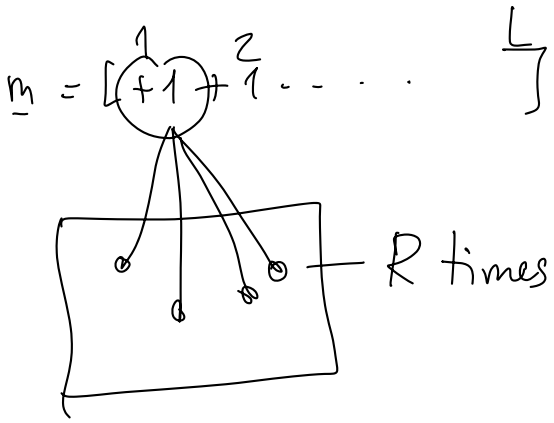


▷ Solution:



Not knowing what was done with this image?!

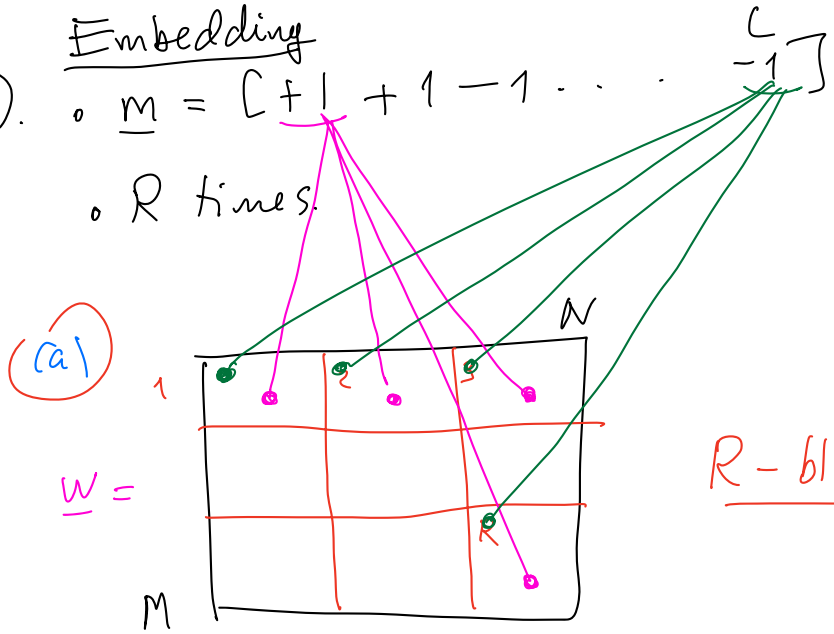
1) Label A



≡ For redundancy. (errors of prediction in view of interference with $\underline{x}$).

2) Embedding

- $\underline{m}$ = [+1 +1 −1 . . .]
- R times.



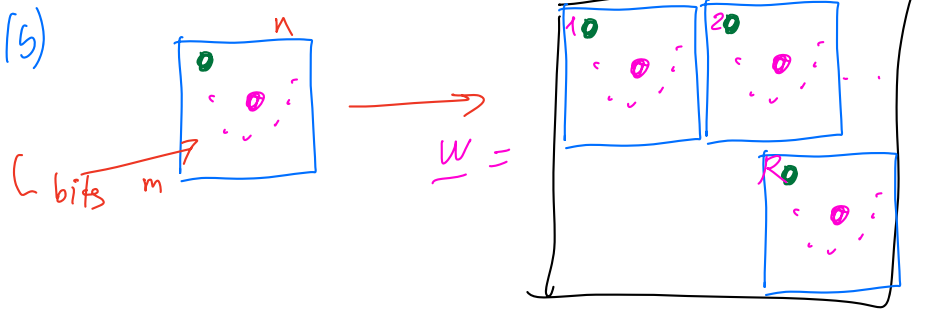
(ordered placement).

(a)

$\underline{w}$ =

R-blocks.

(b)



– repeated R times

$$\underline{y} = \underline{x} + \underline{w}$$

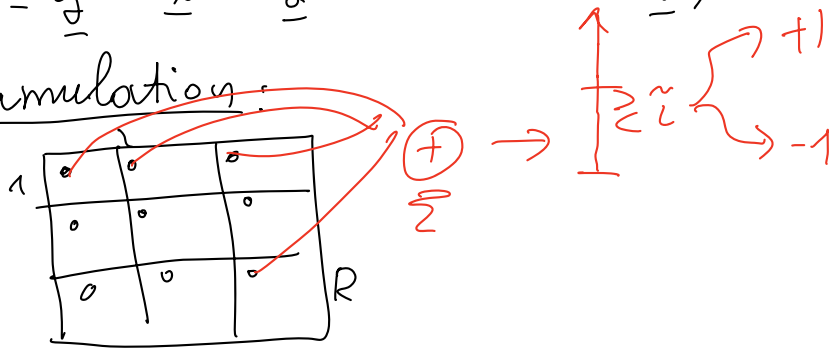
Extraction — No geom. attacks

1) ▷ $\underline{y} = \underline{x} + \underline{w}$

2) Prediction of $\hat{\underline{w}}$:

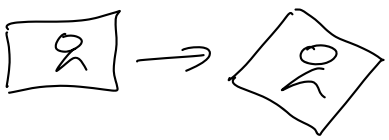
$\hat{\underline{w}} = \underline{y} - \hat{\underline{x}} = \underline{y} - \text{denoiser}(\underline{y})$ | wiener2(.)

3) Accumulation:



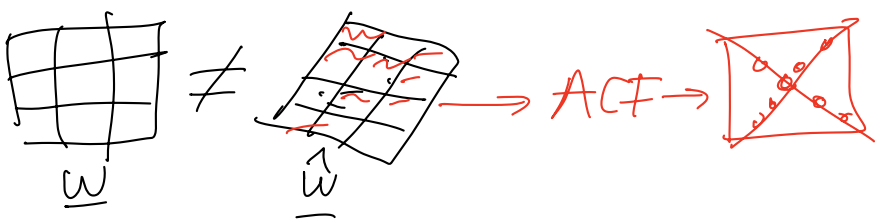
With geom. attacks-

1) $y = G(\underline{x} + \underline{w})$

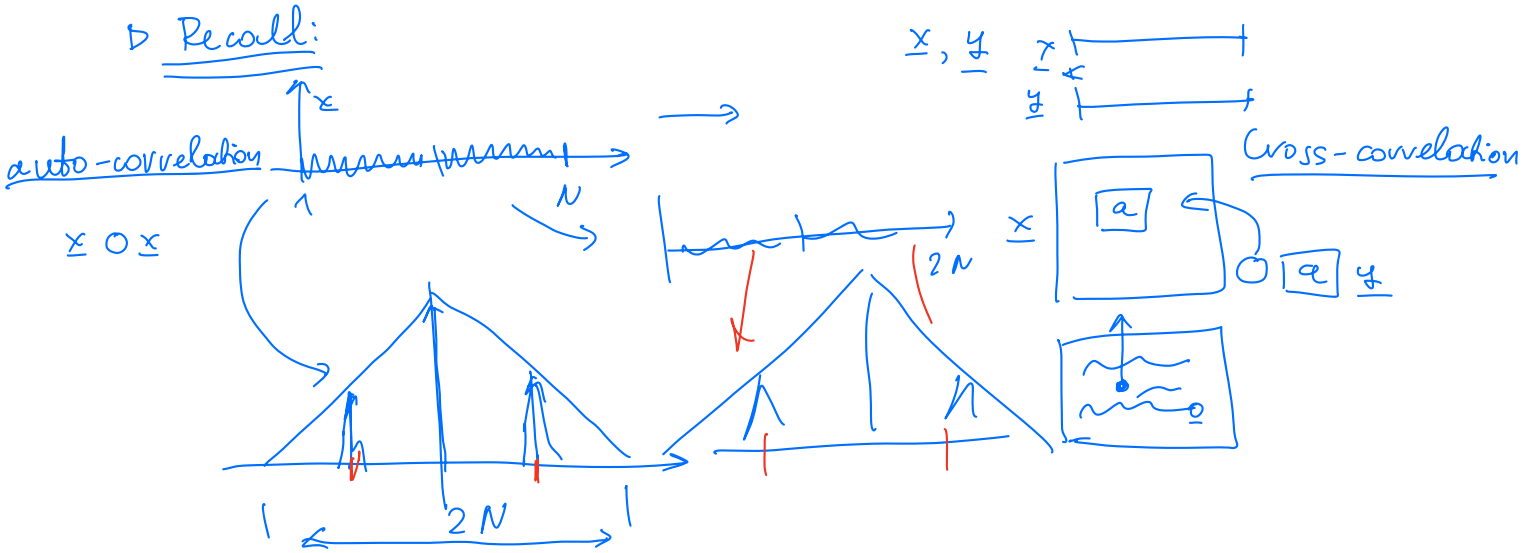


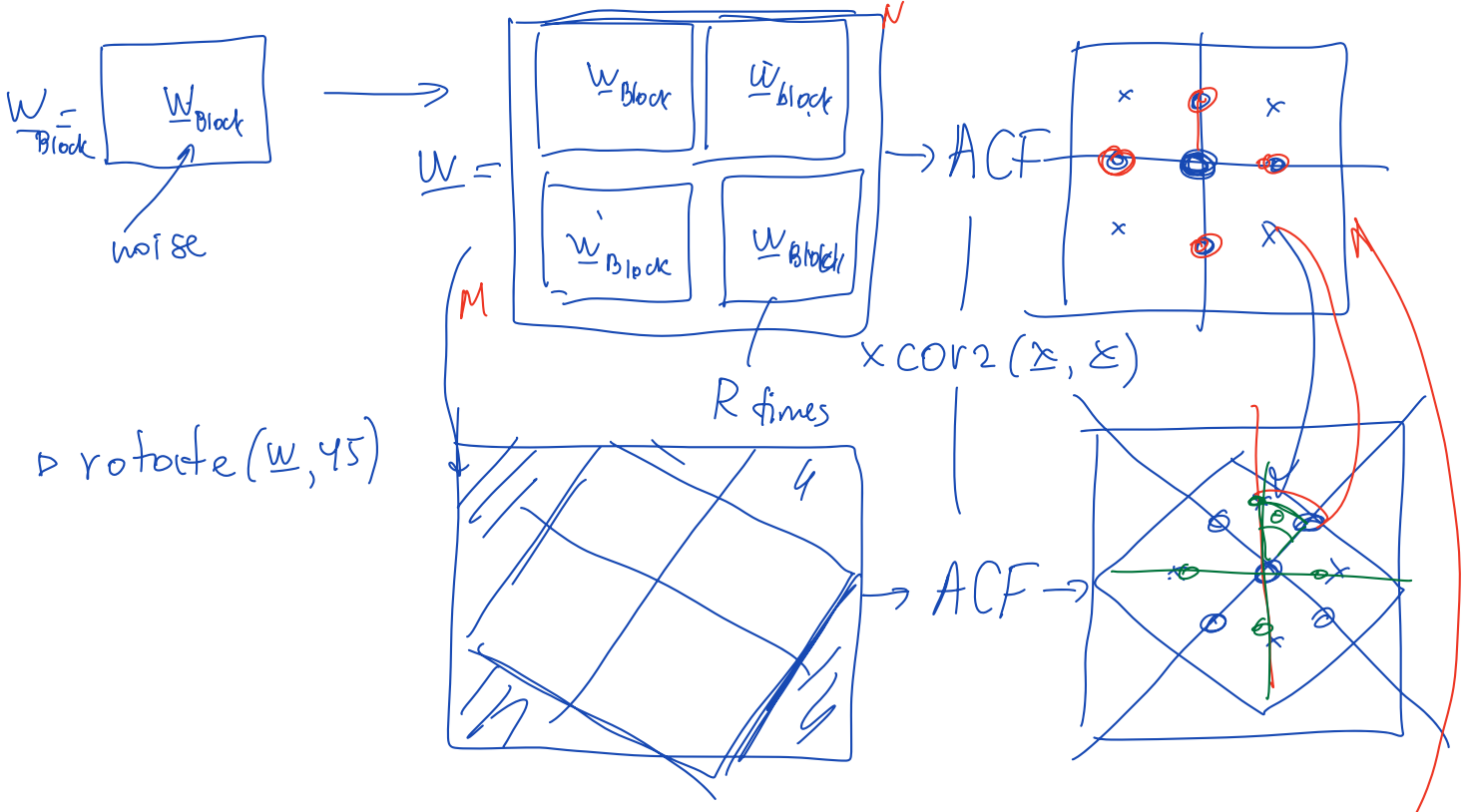
2) Prediction

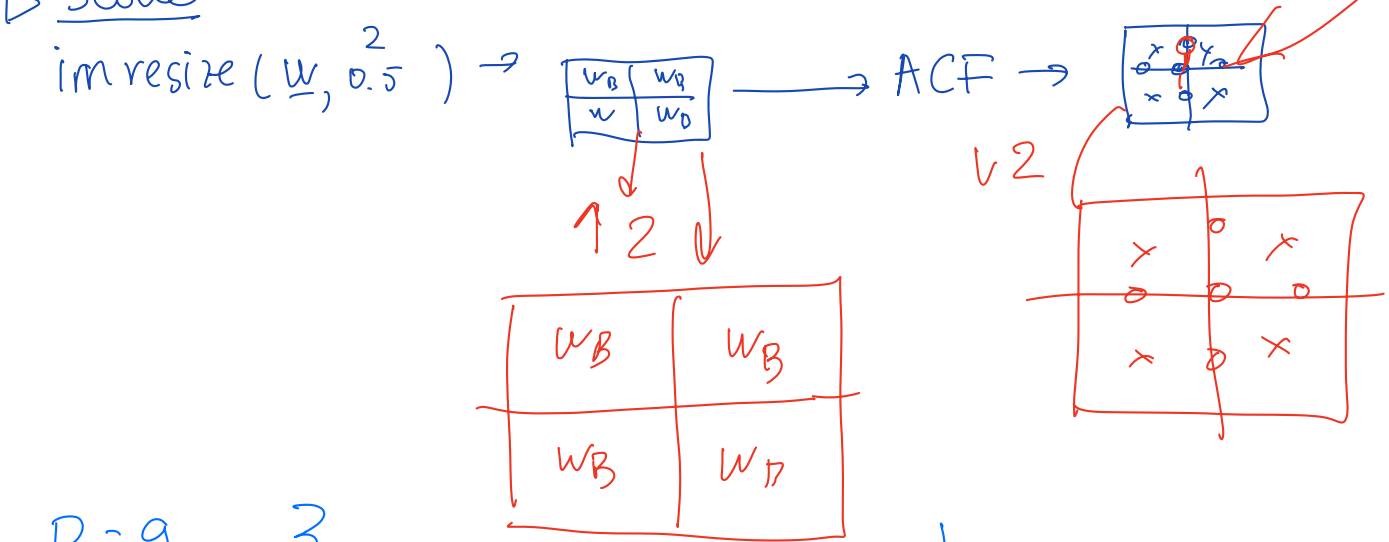
$\hat{\underline{w}} = \underline{y} - \hat{\underline{x}} = \underline{y} - \text{denoiser}(\underline{y})$ | wiener2(.)

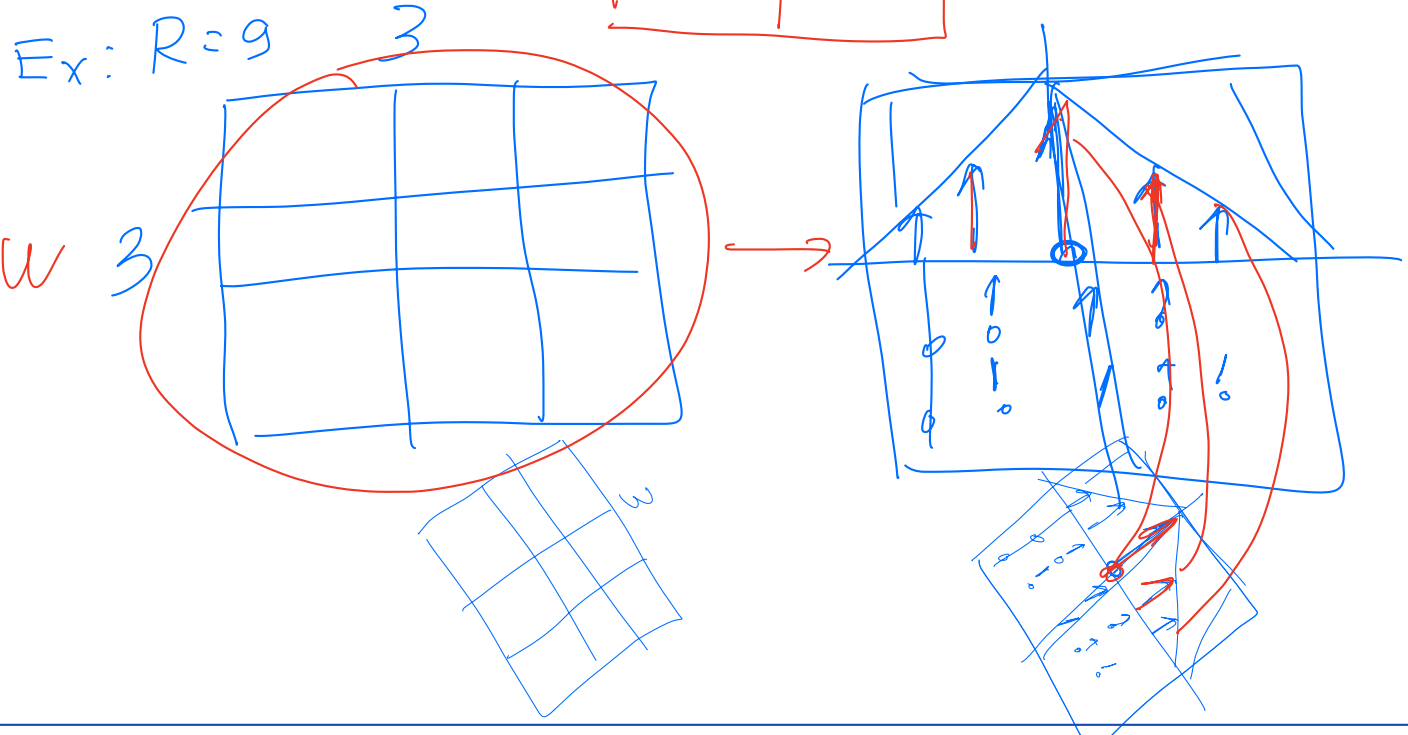


▷ Recall:

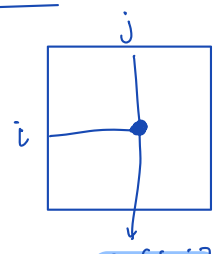



W = W_Block
noise
W_Block
W_Block
W_Block
W_Block
N
M
W =
ACF
× corr2(·,·)
R times
▷ rotate(W, 45)
4
ACF


▷ scale
imresize(W, 0.5^2)
W_B
W_B
W
W_D
ACF
V2
↑2↓
W_B
W_B
W_B
W_D


Ex: R = 9
3
W
3
W

2) Quantization modulation.



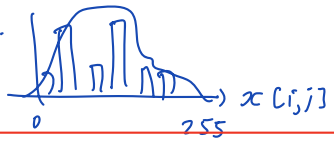
▷ additive modulation

• Direct domain:

$$y[i,j] = x[i,j] + w[i,j] \quad \hookrightarrow (M, K).$$

▷ quantization modulation

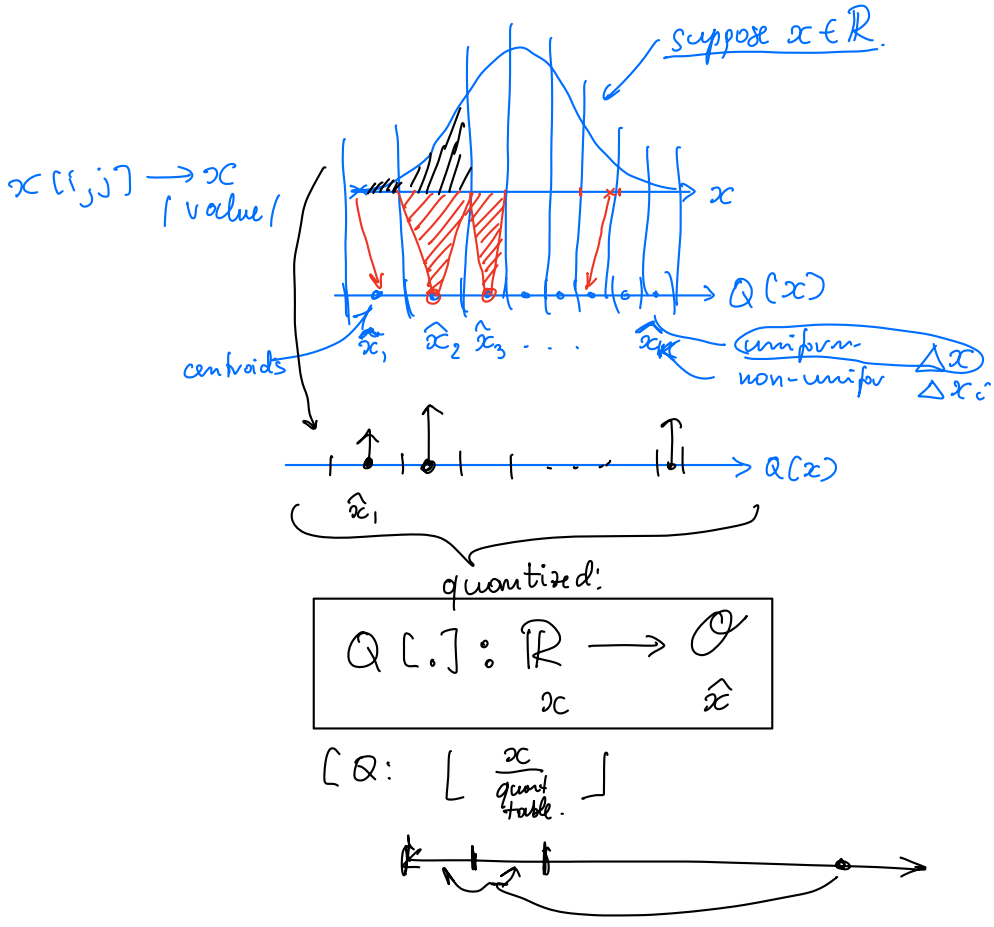
• histogram.



$$y[i,j] = \begin{cases} Q_0[x[i,j]], & \text{for } m=0, \\ Q_1[x[i,j]], & \text{for } m=1 \end{cases}$$

▷ where $Q_0[\cdot]$ for the quantizer for bit 0
$Q_1[\cdot]$ for bit 1, respectively

▷ Recall: quantization



Modulation

$m = [0\ (1)\ 0 \ldots]$

$\underline{x} \rightarrow [f_{\underline{K}}(\underline{x}, m)] \rightarrow \underline{y}$

$\underline{K}$

$m = 0$: $Q_0[x(i,j)]$

$m = 1$: $Q_1[x(i,j)]$

0/1

i, j

$x(i,j) \longrightarrow = 125$

127

x^* – value (magn.)

$Q_0[x]$ 5 10 15 20 125

$y(i,j) = Q_0[x(i,j)] = 125$

$Q_1[x]$ 5 127

$y(i,j) = Q_1[x(i,j)] = 127$

J

i

Selection of interval Δx

Robustness to distortions

visibility of modulation distortion.

Extraction

- given: $\underline{K}$

$\underline{y}$ i j

value $\equiv y = 125 \rightarrow \hat{m}$ (bit) → 0 / 1

(a) (no noise)

$j = 1, \ldots$

$|y - Q_0^j|$

$|y - Q_1^j|$

centroid y

Q_0 1 2 - - - J $m=0$

Q_1 $m=1$

$\hat{m} = 0$.

▷ Decision rule: $|y - Q_0^j| \gtrless |y - Q_1^j|$ → 0 / 1

(L.SB).

Additive:

float: → $w \Rightarrow \hat{m}$ → accurate.

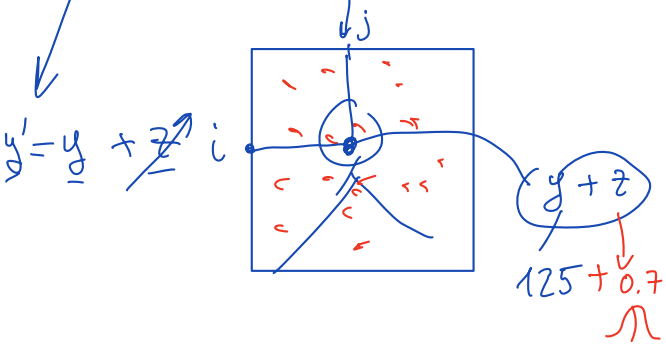
e/t → $\hat{m}$ → not very accurate

Q: is it important x is → float ? e & t. ?

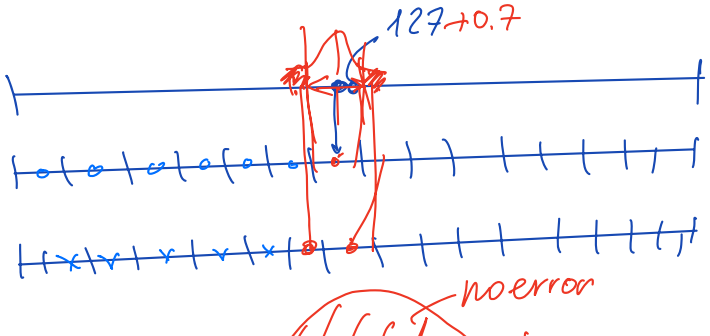
it is → $\hat{x} \rightarrow y - \hat{x} = \hat{a}$

↳ Q → 0 / 1

- given K

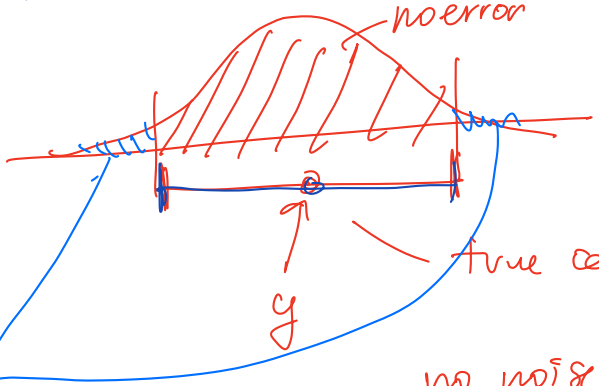


(b) with noise.

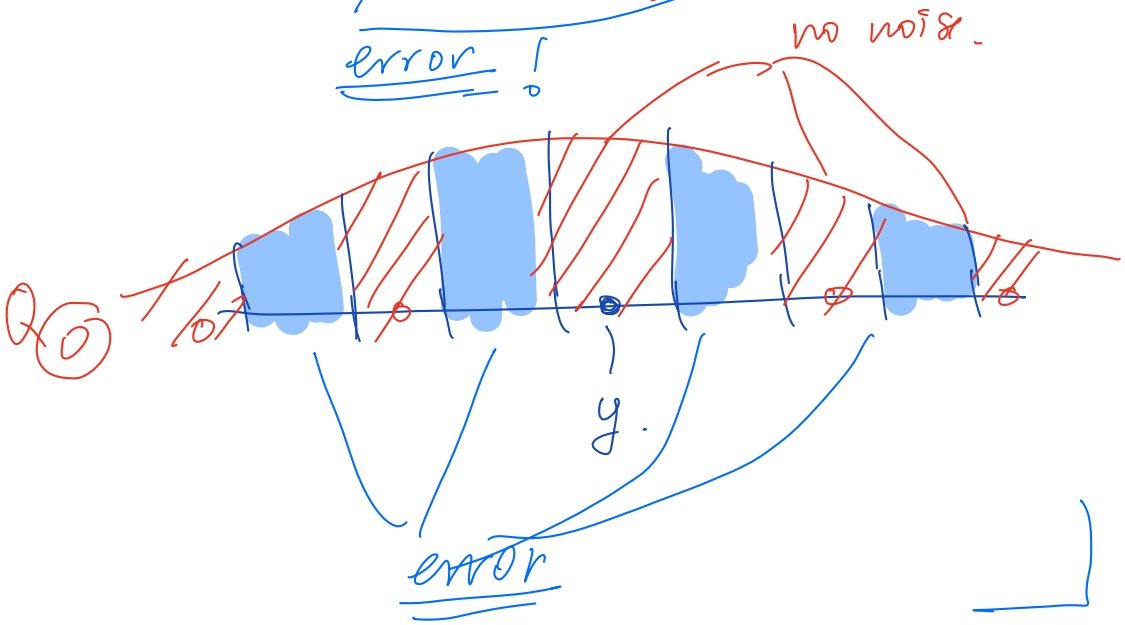


▷ Robustness.

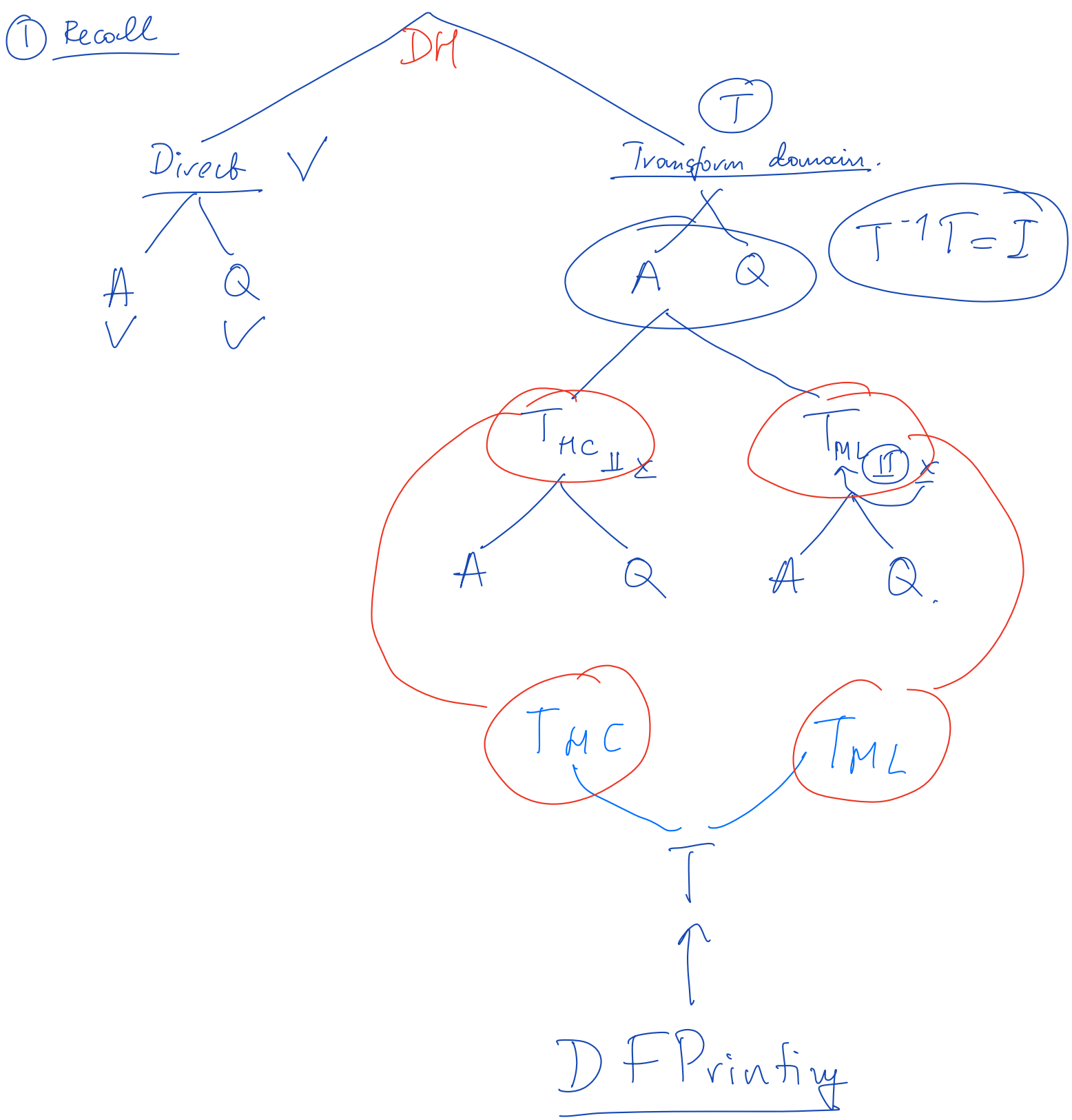
①



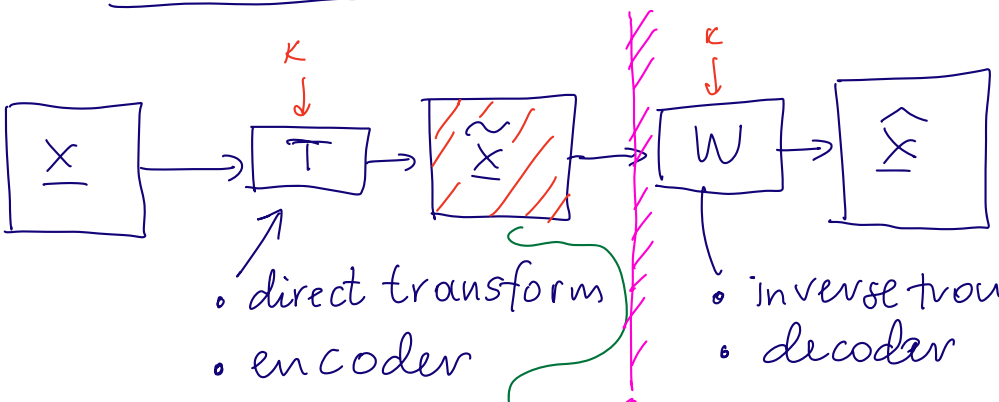
error !



▷ LSB: a part. case of quant. modulation.


① Recall
DH
Direct
A
Q
V
V
T
Transform domain.
A
Q
$T^{-1}T = I$
T_{MC}
T_{ML}
A
Q
A
Q
T_{MC}
T_{ML}
T
DFPrinting

1) Transforms (Defender)

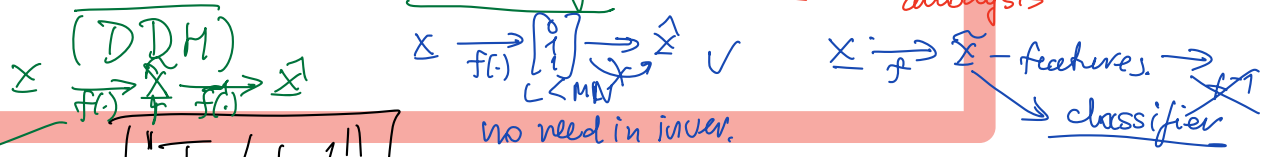


stop

▷ DWM (DRM)

▷ DFPting

[T.H. forensic analysis]



"T / W"

Yes, we need to choose T ⇒ W = T⁻¹

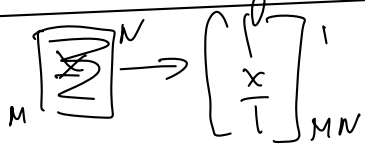
HC

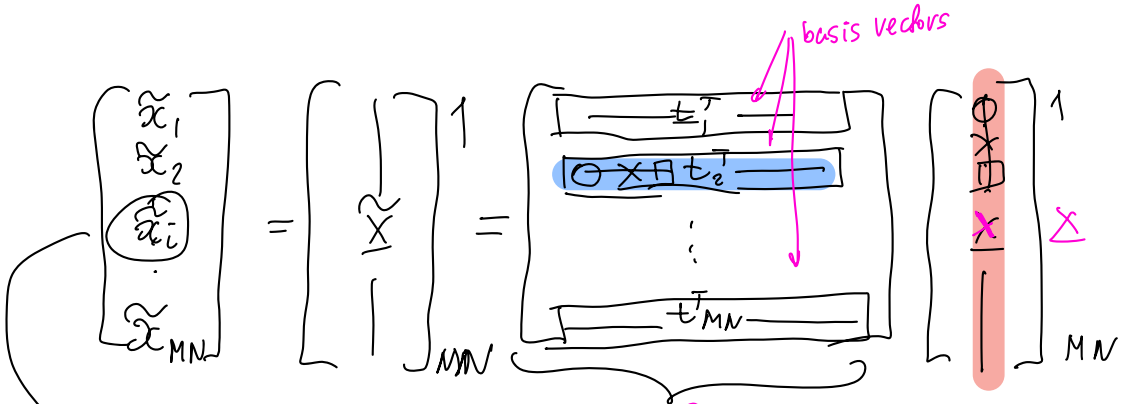
ML

Sect. A → see below

1) HC (hand-crafter ≡ engineered) transform

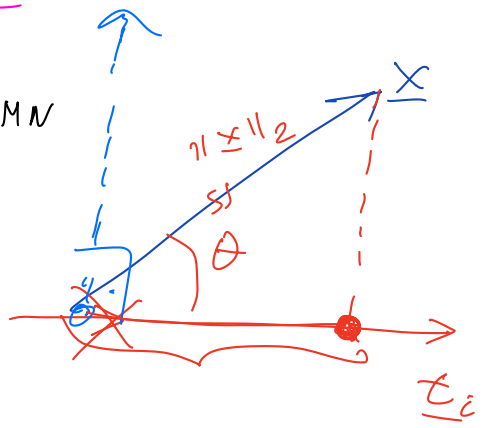
▷ Recall.





$\tilde{x} = T x$

$$\tilde{x}_i = t_i^T x = x^T t_i = \sum_{j=1}^{M.N} x_j \cdot t_{ji}$$



▷ HC Transform : a selection of T (resp. W).

▷ "manually" → expertise of person.

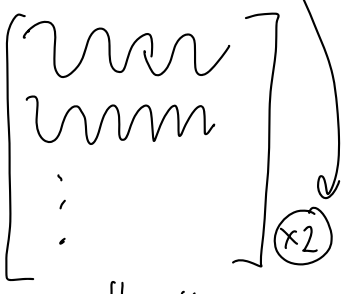
▷ selection is $\perp\!\!\!\perp$ to data X. (no ML)

- in practice:

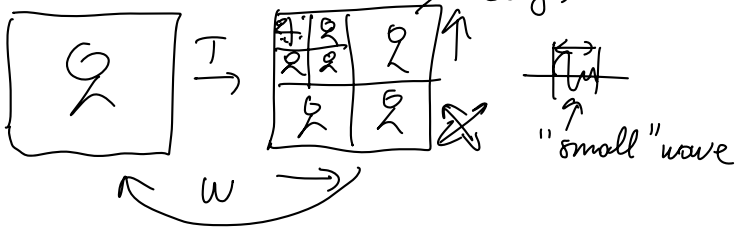
T to be:

- DFT

- DCT



- DWT ↑ wavelet



invertable

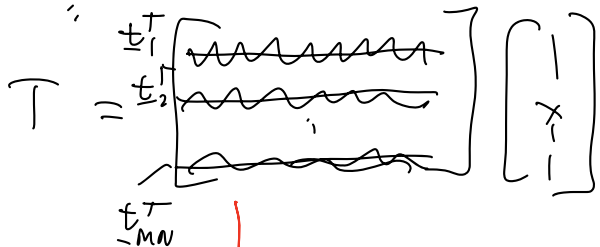
- Macdonnan.

Invertability

- it depends on the construction of matrix T. $L \times (MN)$

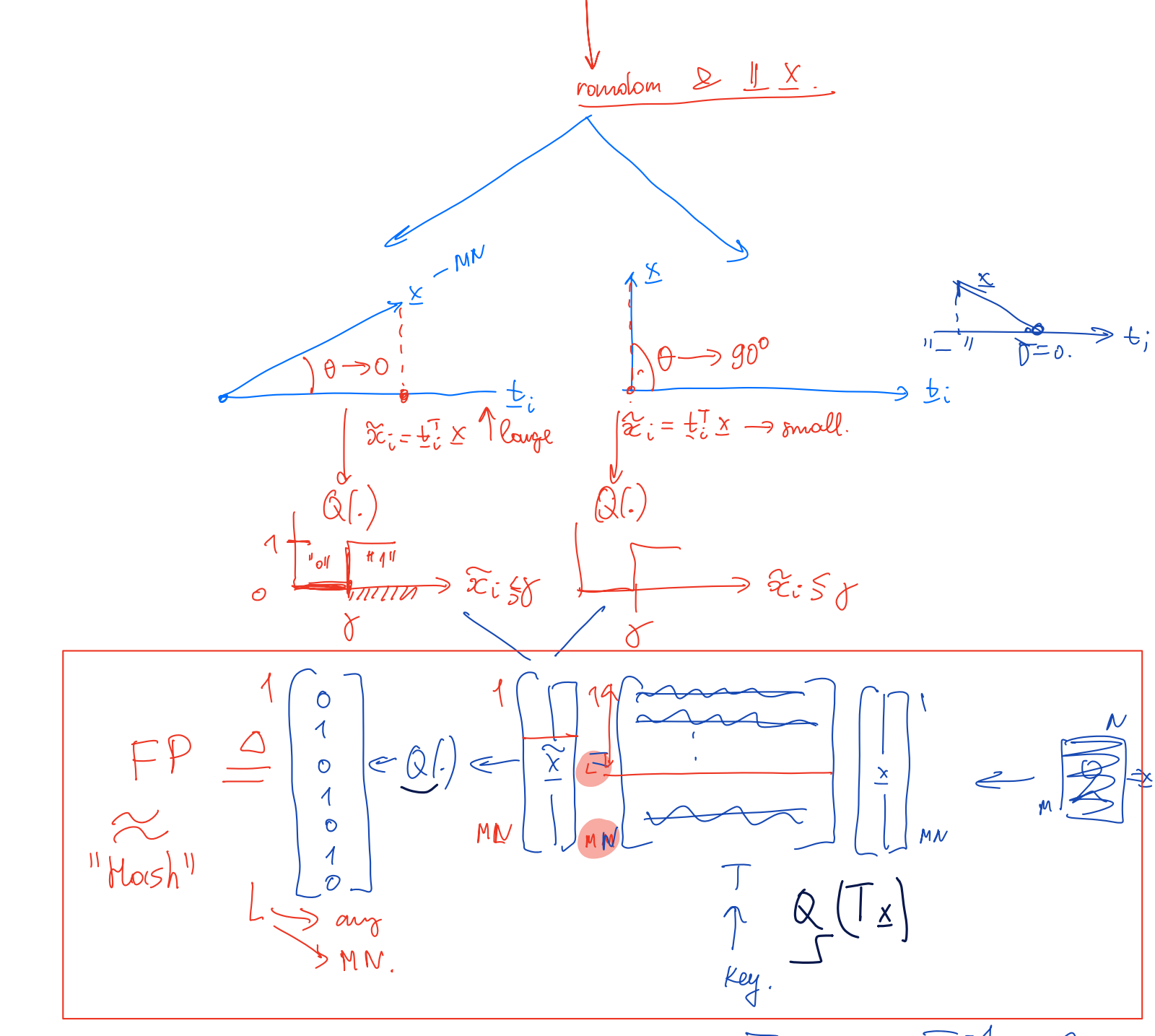
- Random projection

$\mathcal{N}(0, \frac{1}{MN} I_{MN})$



① $T \perp\!\!\!\perp X$

② $T \leftarrow$ keep it • seed (K). (security).


random $\underline{\alpha} \perp\!\!\!\perp \underline{x}$.
$\underline{x}$ - MN
$\theta \to 0$
$\underline{t}_i$
$\tilde{x}_i = \underline{t}_i^T \underline{x}$ ↑ large
$\theta \to 90^\circ$
$\tilde{x}_i = \underline{t}_i^T \underline{x} \to$ small.
$\theta = 0$.
$Q(.)$
"0" "1"
γ
FP $\triangleq$
"Hash"
$Q(.)$
MN
L
MN.
T
Key.
$Q(T\underline{x})$
N
M
$\underline{x}$
• $L < ML \to T \longrightarrow T^{-1} - ?$
SVD: $T = U S V^{-1}$
$T^{-1} = (\quad)^{-1}$
$= V S^{-1} U^{-1}$
$\lambda_i \geq 0$
$\lambda_1 \ldots$
≥ 0
MN
MN
① $L < MN$ to avoid a perfect inversability
② $\tilde{\underline{x}} \to Q(\tilde{\underline{x}}) \to 1$ (or 0.)
Defender (FP)

▷ Attacker : $Q(\tilde{x}) \longmapsto \hat{x} \longleftrightarrow x$.

• Invertability

15:13

ML (maths).

"SVD":

→ approximate solution (exact).

$\tilde{y} = T x$ – linear equations.

$\varphi(x)$ or $\mathcal{L}(x) = \frac{1}{2}\|\tilde{y} - Tx\|_2^2$

$\hat{x} = \underset{x \in \mathbb{R}^{MN}}{\operatorname{argmin}} \mathcal{L}(x)$

$= \underset{x}{\operatorname{argmin}} \frac{1}{2}\|\tilde{y} - Tx\|_2^2$.

ℓ_2

$\nabla_x \mathcal{L}(x) = 0$.

$\hat{x} = (T^T T)^{-1} T^T \tilde{y}$

$T^{-1} \to (T^T T)^{-1}$

W_θ

ML

• $\{x_i, Q(\tilde{x}_i)\}_{i=1}^N$

$y_i \quad L = T x_i$

$x_i \to T \to \tilde{x}_i \to Q \to y_i$

1 2 ⋮ N

x y

? → Key-?

$\to W_\theta \to \hat{x}$

$\mathcal{L}(\theta) = \sum_{i=1}^{N} \|x_i - W_\theta(y_i)\|_2^2$

y M N $\hat{x}$

W_θ GAN

$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \mathcal{L}(\theta)$

$L \sim MN$

① T not invertable $L \ll MN$.

② $Q(\tilde{x})$ → loss of inform. → stochas. (M.A.P.)

$\mathcal{L}(x) = \frac{1}{2}\|\tilde{y} - Tx\|_2^2 + \lambda \Omega(x)$

prior

$\Omega(x)$

ℓ_2: $\|x\|_2$ → ridge

ℓ_1: $\|x\|_1$ → Lasso.

$\nabla_x \mathcal{L}(x) = 0$..

$Q(\tilde{x}) \to$ No

for the attacker.

N M L

• Not feasible.

▷ Sec A. T/W based on ML. (Defender)

DWM /DMI

DFP

$\hat{\underline{X}} = \underline{X}$

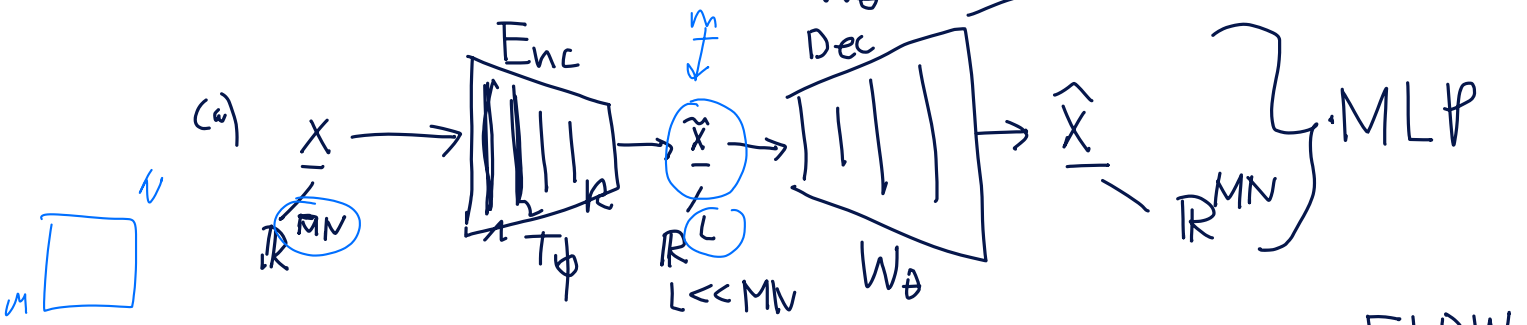
$\underline{X} \rightarrow [T] \rightarrow \underline{\tilde{X}} \rightarrow [W] \rightarrow \underline{\hat{X}}$

▷ ML: the transf: $T \longrightarrow T_\phi$

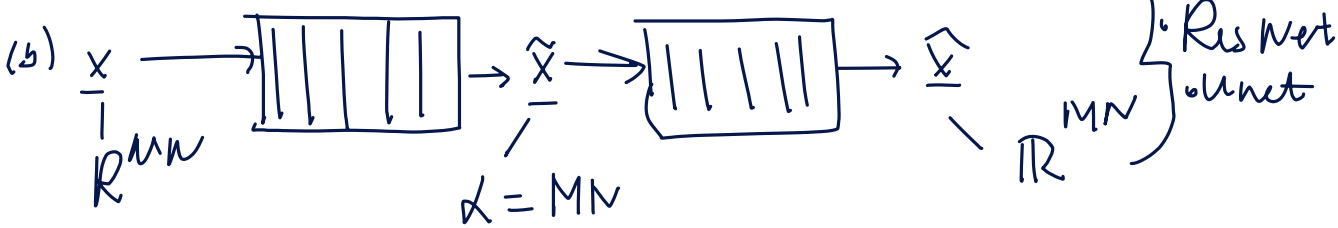
$W \longrightarrow W_\theta$

Auto-encoder (AE)

(a)



(b)



Ex: Enc:

$$\tilde{\underline{X}} = T_\phi(\underline{X}) = \sigma_R\left(T_R \cdots \sigma_2\left(T_2 \cdot \sigma_1\left(T_1 \underline{X} + \underline{b}_1\right) + \underline{b}_2\right) + \cdots \underline{b}_R\right)$$

(Q)

$\underline{X}_1$

$\phi = \{T_1, T_2, \ldots, T_k, \underline{b}_1, \ldots \underline{b}_R\}$

Dec:

$$\hat{\underline{X}} = W_\theta(\tilde{\underline{X}}) = \sigma_R(W_R - \cdots - \sigma_1(W_1 \tilde{\underline{X}} + \underline{a}_1) - \cdots + \underline{b}_R)$$

▷ Training

$$\mathcal{L}(\phi, \theta) = \sum_{i=1}^{N} \| \underline{x}_i - \underline{\hat{x}}_i \|_2 = \sum_{i=1}^{N} \| \underline{x}_i - W_\theta (T_\phi(\underline{x}_i)) \|_2^2$$

$\underline{\hat{x}}_i = W_\theta T_\phi(\underline{x})$

$\underline{\tilde{x}}$

1) $\nabla_{\phi,\theta} = 0$

2) GD: $(\hat{\phi}, \hat{\theta}) = (\phi, \theta)^K - \beta \nabla_{\phi,\theta} \mathcal{L}(\phi, \theta)$

Remarks:

① # in $\phi, \theta \longrightarrow N \uparrow$

② $\underline{x} \mapsto \triangleright \; || \; \triangleleft \mapsto \underline{\hat{x}}$

T_ϕ^* $\tilde{x}$ W_θ^*

DWM (T_ϕ^* & W_θ^*)

DFP

$\underline{x} \mapsto \triangleright \; || \; \mapsto \underline{\tilde{x}} \mapsto [\,Q(\cdot)\,] \mapsto \underline{y}$

T_ϕ^*