

Codes graphiques pour stockage d'informations

Master IMAGINA

Iuliia Tkachenko
iuliia.tkachenko@lirmm.fr

Plan du cours

- Introduction
- Théorie
- QR code
- Enrichissement de codes graphiques
- Codes graphiques pour authentification

Introduction

- Un code à barres :

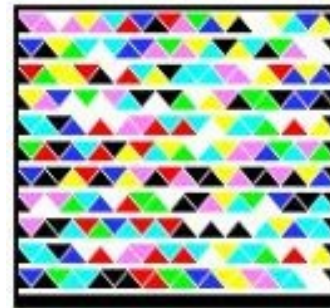
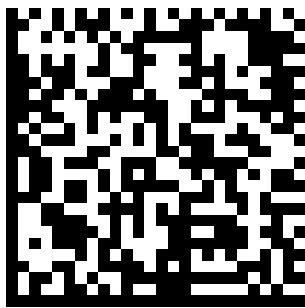
Un code à barres est la représentation d'une donnée numérique ou alphanumérique sous forme d'un symbole constitué de barres et d'espaces dont l'épaisseur varie en fonction de la symbologie utilisée et des données ainsi codées. [Wikipédia]



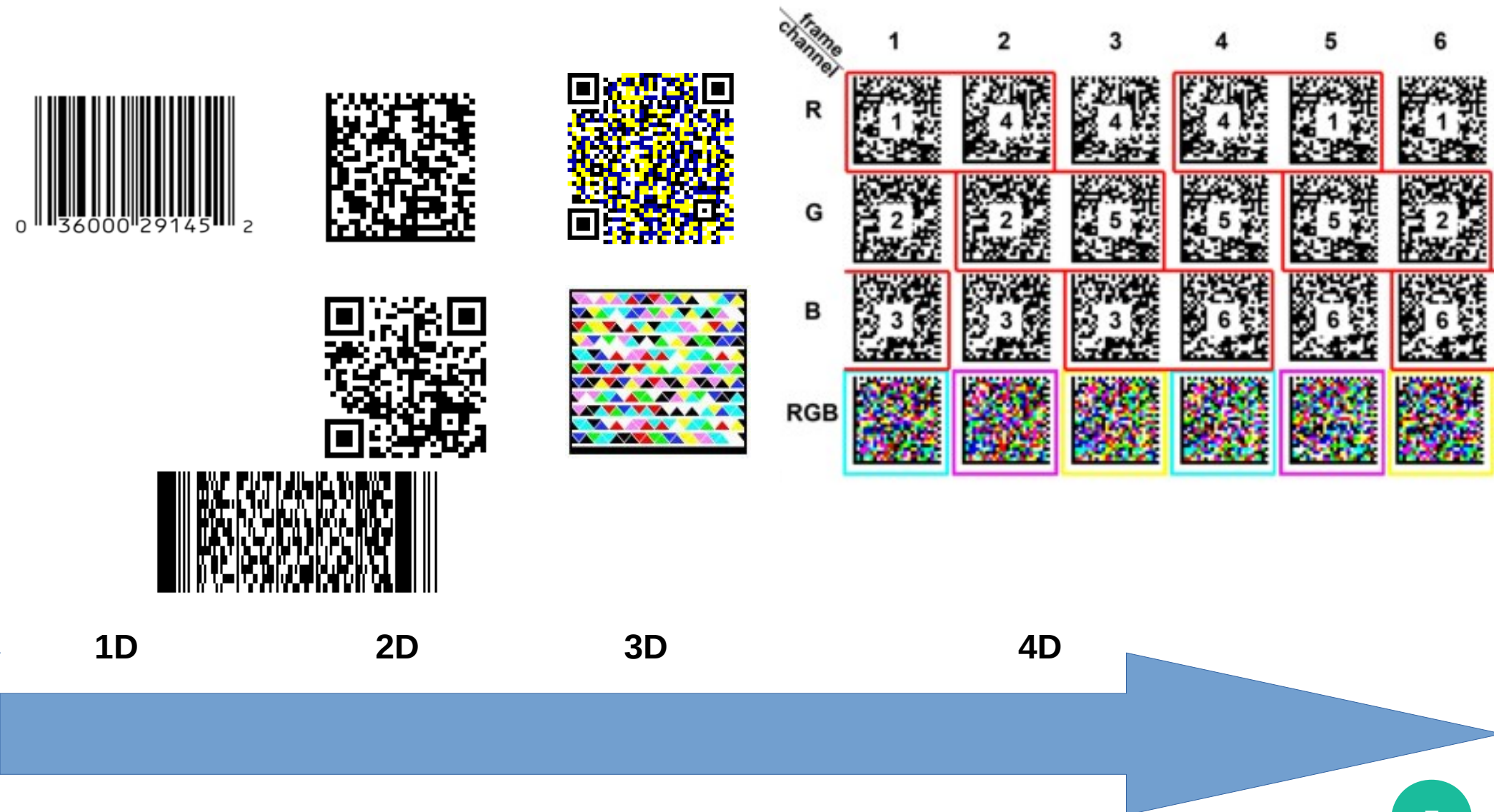
Introduction

- Un code graphique :

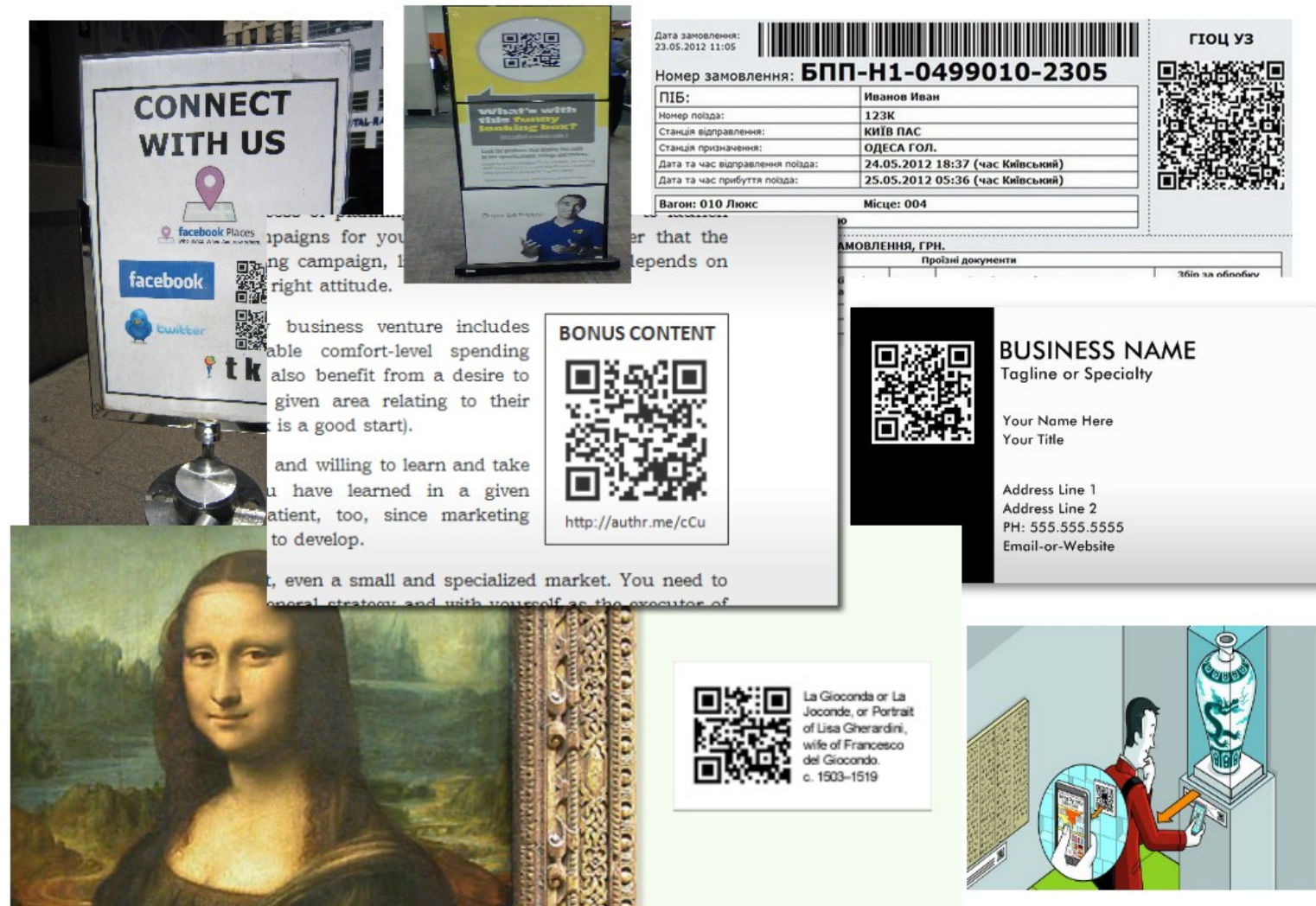
Un code graphique est la représentation d'une donnée numérique ou alphanumérique sous la forme d'un symbole où chaque bit d'information est représenté par une forme graphique (barre, carré, triangle, ...) et sa couleur (optionnel).



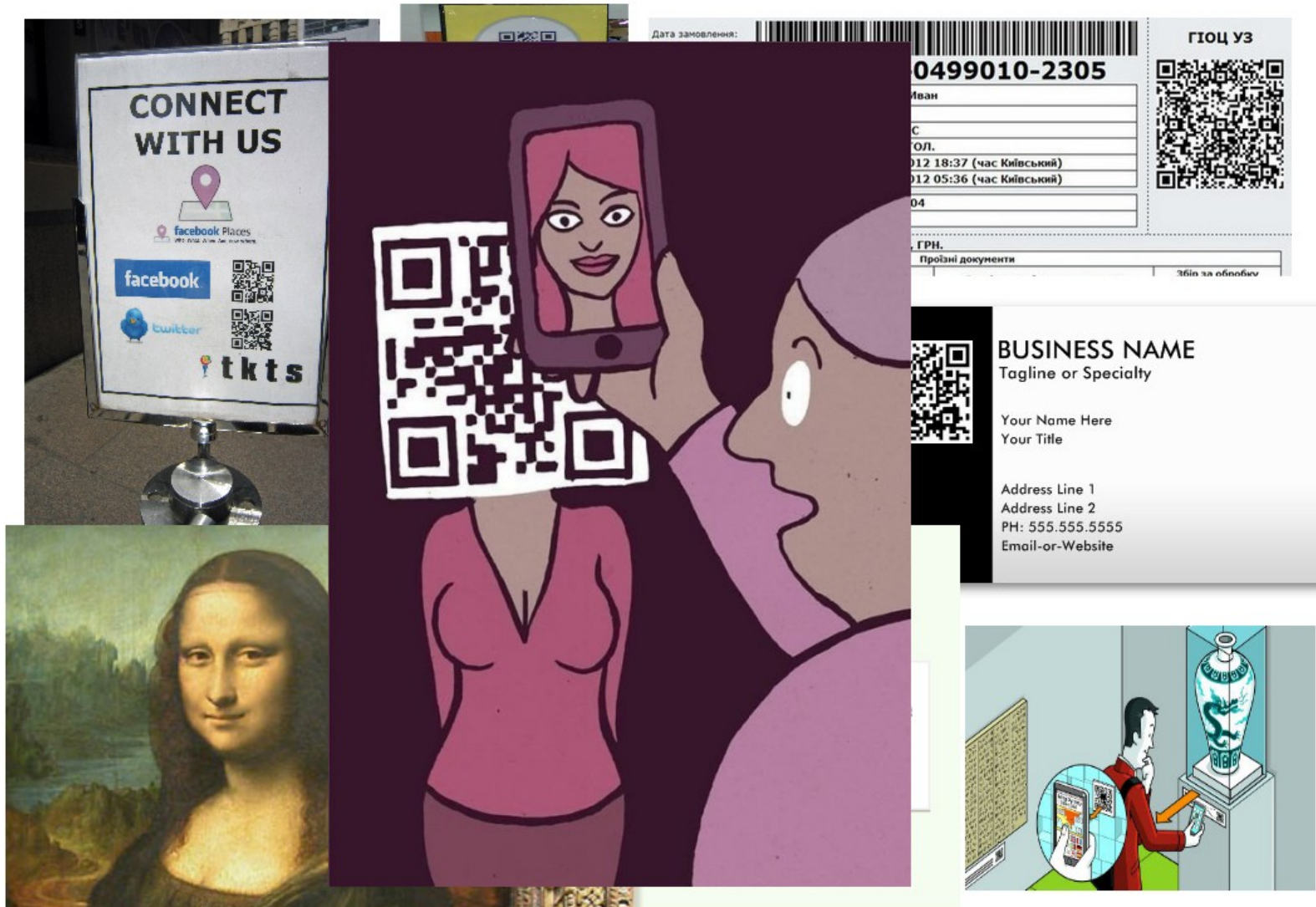
Evolution de codes graphiques



QR code dans la vie quotidienne



QR code dans la vie quotidienne



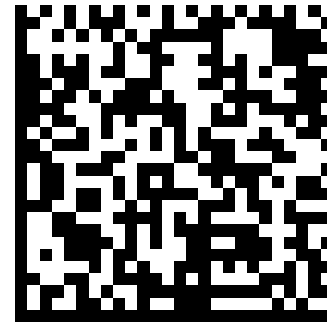
Points forts des codes graphiques

- Petite taille
- Haute capacité de stockage
- Correction des erreurs de capture
- Facilité de génération
- Rapidité de lecture

Exemples de codes graphiques 2D



QR code



DataMatrix



PDF417



Aztec code

Comparaison des codes 2D

Barcode characteristics		QR code	PDF 417	Data Matrix	Aztec code
Code type		2D barcode	Stacked barcode	2D barcode	2D barcode
Maximal	numeric data	7 089	2 710	3 116	3 832
	alphanumeric data	4 296	1 850	2 355	3 067
	binary data	2 953	1 018	1 556	1 914
	kanji data	1 817	554	778	-
Large capacity		yes	yes	yes	yes
Small printout size		yes	no	yes	yes
High speed scan		yes	yes	yes	yes
Error correction		yes	yes	yes	yes
Weak points		Reading of higher versions	Scanning device must be carefully aligned	Localization difficulties	
Applications		Ticketing Advertising	Airline ticketing Postage	Product labeling	Transport Governmental

Plan

- Introduction
- **Théorie**
- QR code
- Enrichissement de codes graphiques
- Codes graphiques pour authentification
- Conclusion

Base théorique

Pour création et lecture d'un code graphique, nous avons besoin de connaissance en

- Traitement d'images
- Théorie de codage

Base théorique

Pour la création et la lecture d'un code graphique, nous avons besoin de connaissance en

- **Traitement d'images**
 - Génération d'une image avec un code graphique
 - Pré-traitement d'une image avec un code graphique
 - Binarisation d'un code graphique
 - Correction des conditions d'éclairage et de capture

Traitement d'images

Pré-traitement d'une image avec un code graphique

- Recherche des patterns de positions
 - Détection de contours (détecteur de Canny)

1. Filtre Gaussien

2. Gradient d'intensité

3. Suppression des valeurs non-maximales

4. Filtrage d'une image du gradient



Traitement d'images

Pré-traitement d'une image avec un code graphique

- Recherche des patterns de positions
 - Détection de contours (détecteur de Canny)
 - Technique de reconnaissance de formes (transformée de Hough)

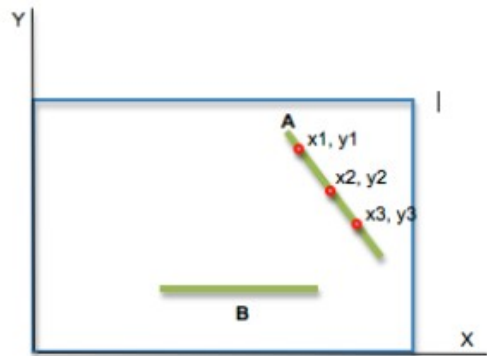


Fig 1

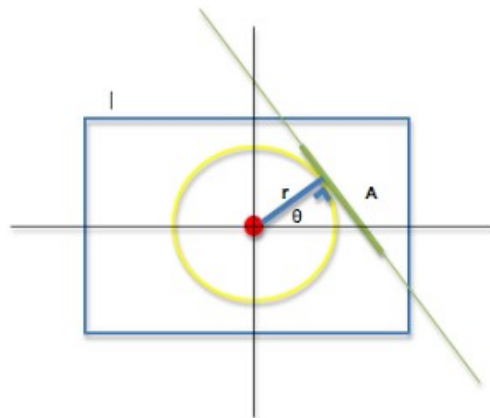


Fig 2

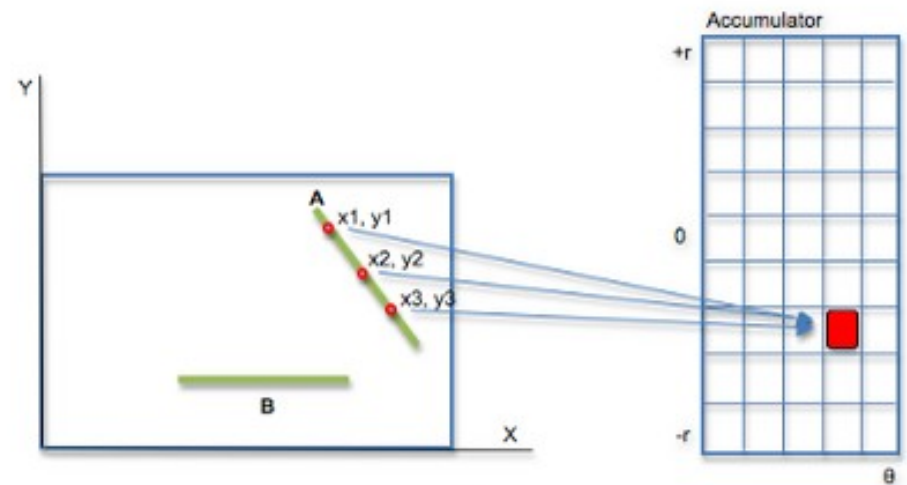
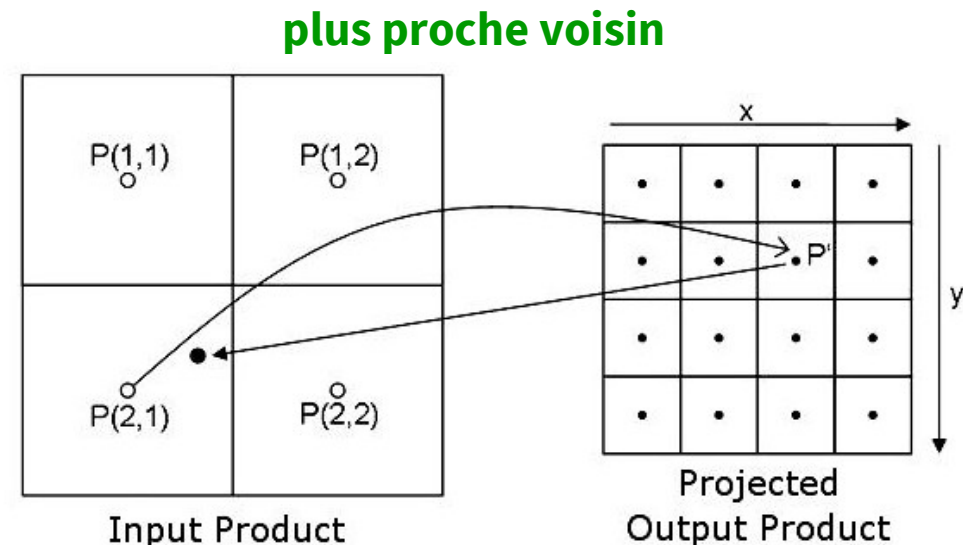


Fig 3 - Illustration

Traitement d'images

Pré-traitement d'une image avec un code graphique

- Recherche des patterns de positions
- Échantillonnage d'une image
 - Interpolation (plus proche voisin, bilinéaire, bicubique, ...)



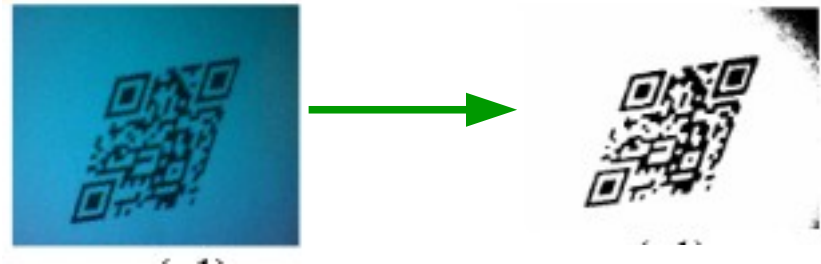
Traitement d'images

Binarisation d'un code graphique

- **Binarisation globale :** Un seuil global est fixé pour toute l'image

Méthode classique :

- $Th = (max - min)/2$



Méthode d'Otsu :

- définit le seuil qui minimise la variance intra-classe, qui est recherché à partir de tous les seuillages possibles.

Traitement d'images

Binarisation d'un code graphique

- **Binarisation globale** : Un seuil global est fixé pour toute l'image

Méthode classique : $Th = (max - min)/2$

Algorithm d'Otsu

1. Calculer l'histogramme et les probabilités de chaque niveau d'intensité.
2. Définir les $\omega_i(0)$ et $\mu_i(0)$ initiaux
(ω_i représente la probabilité d'être dans la i -ème classe, μ_i – moyenne d'une classe i).
3. Parcourir tous les seuils possibles $t = 1 \dots intensité\ max$
 - 1) Mettre à jour $\omega_i(t)$ et $\mu_i(t)$
 - 2) Calculer $\sigma_b^2(t)$
4. Le seuil désiré correspond au $\sigma_b^2(t)$ maximum.

Traitement d'images

Binarisation d'un code graphique

- **Binarization globale** : Un seuil global est fixé pour toute l'image

Méthode classique : $Th = (max - min)/2$

Méthode d'Otsu définit le seuil qui minimise la variance intra-classe est recherché à partir de tous les seuillages possibles.



Otsu



classique



La méthode d'Otsu est plus efficace.
Plusieurs d'améliorations de cette methode ont été proposées.

Traitement d'images

Binarisation d'un code graphique

- Binarisation locale : Un seuil local est défini dans une fenêtre pré-déterminée.

Algorithme de Niblack calcule un seuil pixel par pixel à l'aide d'une fenêtre glissante rectangulaire sur l'image.

$$Th_{Niblack} = \mu + k * \sigma,$$

ou $k = -0.2$, μ et σ est une moyenne et une écart-type



Niblack



Otsu

Traitement d'images

Binarisation d'un code graphique

- Binarisation locale : Un seuil local est défini dans une fenêtre pré-déterminée.

Algorithme de Sauvola (peut améliorer la méthode de Niblack) calcule un seuil en utilisant la plage dynamique de la variance d'une image.

$$Th_{\text{Niblack}} = \mu * (1 - k * (1 - \sigma/R)),$$

ou $k = 0.5$, $R = 128$, μ et σ est une moyenne et une écart-type

Traitement d'images

Binarisation d'un code graphique

- **Binarisation mixte : Un seuil global est calculé en utilisant des seuils locaux.**

Un seuil global est calculé en utilisant 9 parties d'image de taille 60x60 pixels [1]. Dans chaque partie un seuil classique est calculé ($Th = (max-min)/2$). Le seuil globale est défini comme la valeur minimale des seuils locaux.

[1] E. Ohbuchi, H. Hanaizumi, and L. A. Hock. Barcode readers using the camera device in mobile phones. In Cyberworlds, 2004 International Conference on, pages 260-265. IEEE, 2004.
[2] Y. Liu and M. Liu. Automatic recognition algorithm of quick response code based on embedded system. In Intelligent Systems Design and Applications, 2006. ISDA'06. Sixth International Conference on, volume 2, pages 783-788. IEEE, 2006.

Traitement d'images

Binarisation d'un code graphique

- **Binarisation mixte : Un seuil global est calculé en utilisant des seuils locaux.**

Méthode de seuillage “multi-niveaux” [2] basée sur l'analyse de l'histogramme.

- Pour un histogramme bi-modal, un seuil est choisi comme une valeur minimale de creux.
- Pour un histogramme uni-modal, nous utilisons une méthode de seuillage itératif, qui calcule seuil d'une valeur moyenne parmi le vieux seuil global et le centre des zones sombres ou claires.
- Pour un histogramme multi-modal, un algorithme de seuil local a été utilisé.



Otsu



Classique



“Multi-niveaux”



[1] E. Ohbuchi, H. Hanaizumi, and L. A. Hock. Barcode readers using the camera device in mobile phones. In Cyberworlds, 2004 International Conference on, pages 260-265. IEEE, 2004.

[2] Y. Liu and M. Liu. Automatic recognition algorithm of quick response code based on embedded system. In Intelligent Systems Design and Applications, 2006. ISDA'06. Sixth International Conference on, volume 2, pages 783-788. IEEE, 2006.

Base théorique

Pour la création et la lecture d'un code graphique, nous avons besoin de connaissance en

- **Traitement d'images**
- **Théorie de codage**
 - Encodage d'une information (=un message)
 - Détection et correction d'erreurs après capture
 - Décodage d'un message

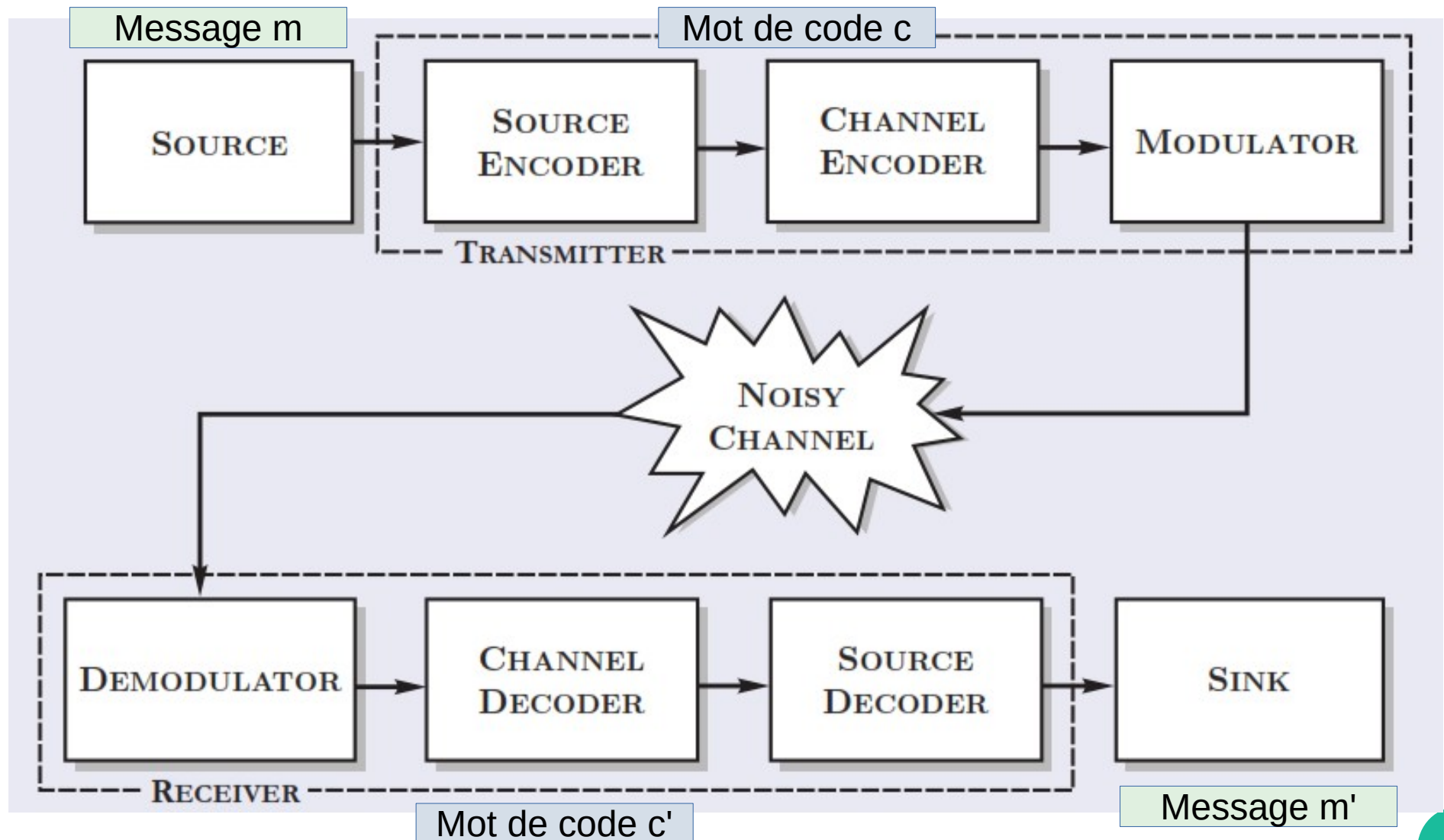
Théorie de codage

- Les codes pour la cryptographie,
- Les codes pour la compression,
- **Les codes pour la correction d'erreur (CCE).**

Un code correcteur d'erreurs est une technique de codage basée sur la redondance de l'information, destinée à détecter, voire corriger, les erreurs qui peuvent-être générées par un canal de communication peu fiable.

Théorie de codage

Système de commutation numérique



Exemple de CCE

Le code de répétition

- Encodage:

Pour un bit d'information – 3 bits sont envoyés

0 → 000

1 → 111

Exemple de CCE

Le code de répétition

- Encodage:

Pour un bit d'information – 3 bits sont envoyés

0 → 000

1 → 111

- Décodage:

Se fait par vote majoritaire.

Par exemple: 010 → 0

011 → 1

Types de CCE

- les codes en bloc :

Ils traitent chaque bloc d'information indépendamment les uns des autres. Chaque mot de code est indépendant des autres mots de code.

- les codes convolutifs :

La sortie d'un codeur convolutif dépend de l'information courante à coder ainsi que de l'information précédente et de l'état du codeur.

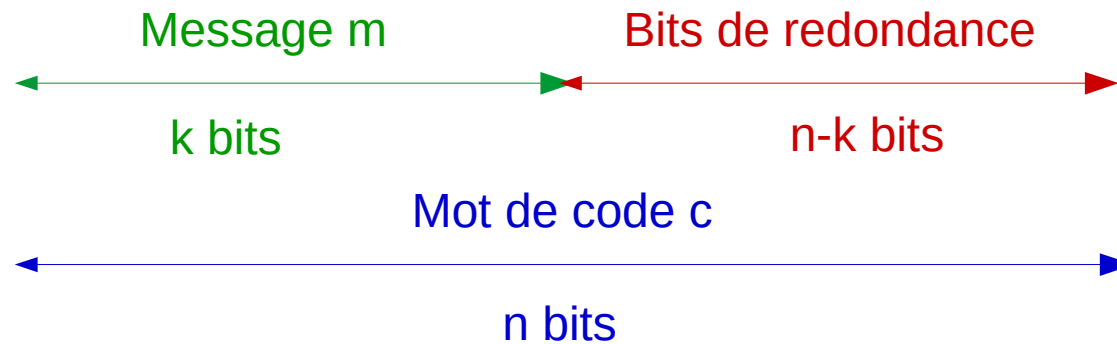
Principe général du codage en blocs

Un message est découpé en blocs de k bits , et un même algorithme est appliqué sur chaque blocs :

- on ajoute des bits de contrôle à la fin de chaque blocs,
- ou on modifie complètement les blocs mais on évite que deux blocs différents soient transformés en un même bloc.

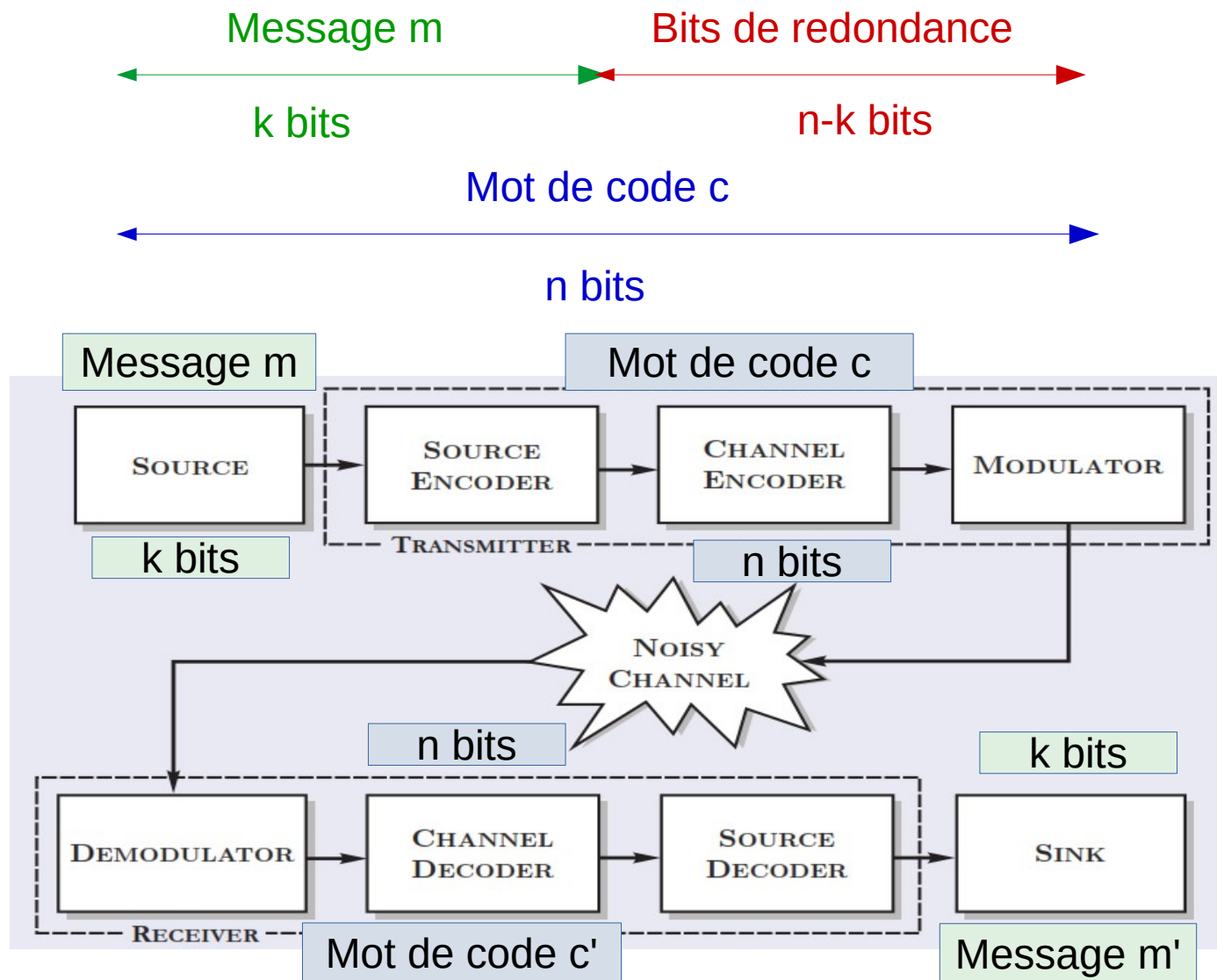
Formation d'un mot de code

Code en block



Formation d'un mot de code

Code en block



Definitions

Un code est une application injective (tout élément de l'ensemble d'arrivée a au plus un antécédent dans l'ensemble de départ)

$$\Phi : \{0,1\}^k \rightarrow \{0,1\}^n.$$

La distance de Hamming (dans le cas binaire (F_2))

Soit deux vecteurs x et y de dimension n .

La distance de Hamming correspond au nombre de bits pour lequel ces deux vecteurs diffèrent : $d_H(x,y) = |\{i: x_i \neq y_i, 0 \leq i \leq n\}|$.

Exemple: La distance de Hamming entre les mots de code (000) et (111) est 3.

Paramètre de code en bloc

- $[n, k, d_{\min}]$ – n est la longueur du code (# bits de mot de code),
 k est la dimension de code (# bits de message),
 d est la distance minimale de Hamming d'un code.
- $(d_{\min} - 1)$ est le nombre d'erreur maximum détectable.
- $t = [(d_{\min} - 1)/2]$ indique le nombre d'erreurs corrigeable.
- $r = k/n$ est le taux d'information d'un code:
 - $r < 1$: ajout d'informations,
 - $r = 1$: pas d'informations supplémentaires,
 - $r > 1$: compression de données.

Exemple de CCE

Le code de répétition

Le code binaire à répétition $[3,1,3]$

- La longueur du code est $n = 3$
- La dimension du code est $k = 1$
- La distance de Hamming $d_{\min} = 3$
- $d_{\min} - 1 = 2$ erreurs détectables
- $t = [(d_{\min} - 1)/2] = 1$ erreur corrigeable
- Le taux d'information est $r = 1/3$

Détection d'une erreur de transmission

Soit $y = (c'_1, \dots, c'_n)$ un mot reçu par un récepteur. La détection d'erreurs est alors simple :

- Soit $\omega = \Psi(c'_1, \dots, c'_k)$ la redondance calculée par le récepteur sur les k premiers symboles reçus.
- Si la redondance calculée ω est égale à la redondance reçue (c'_{k+1}, \dots, c'_n) le mot **c' appartient au code**, sinon il **a détection d'une erreur**.

Détection d'une erreur de transmission

Soit $y = (c'_1, \dots, c'_n)$ un mot reçu par un récepteur. La détection d'erreurs est alors simple :

- Soit $\omega = \Psi(c'_1, \dots, c'_k)$ la redondance calculée par le récepteur sur les k premiers symboles reçus.
- Si la redondance calculée ω est égale à la redondance reçue (c'_{k+1}, \dots, c'_n) le mot **c' appartient au code**, sinon il **c' a détection d'une erreur**.

Le syndrome correspond à l'erreur entre la redondance calculée et la redondance reçue.

Dans le cas binaire le syndrome vaut $s = \omega - (c'_{k+1}, \dots, c'_n) \bmod 2$.

Un syndrome nul indique qu'il n'y a pas eu d'erreurs de transmission.

Code linéaire

- Un code $C [n,k,d_{\min}]$ est **linéaire** s'il existe une matrice G de dimension $k \times n$ dont les coefficients sont dans F tels que l'ensemble des mots de code soient obtenus par le produit matriciel entre le message m et la matrice G .

Pour F_2^n : $C = \{ c' \mid c' = m * G, m \in \{0,1\}^k \}$

- On appelle **G une matrice génératrice**.

Code linéaire

- Étant donné une matrice génératrice G d'un code linéaire avec k lignes et N colonnes, on peut former une autre matrice H , avec $n - k$ lignes et n colonnes, de sorte que l'espace des lignes de G est orthogonal à H :

$$GH^t = 0.$$

- On note **H une matrice de vérification de parité (ou une matrice de contrôle)**.
- Quelque soit le mot de code c : $cH^t = Hc^t = 0$.

Décodage par le syndrome

- Soit G une matrice génératrice et H une matrice de vérification de parité.
- Soit c un mot de code envoyé et $c' = c + b$ le mot de code reçu.

$$c'H^t = H(c+b)^t = Hc^t + Hb^t = Hb^t$$

Décodage par le syndrome

$$c'H^t = H(c+b)^t = Hc^t + Hb^t = Hb^t$$

- Ainsi, le vecteur $s = Hb^t$ ne dépend que du bruit b et non pas du mot de code envoyé.
- Ce vecteur $s = Hb^t$ est appelé le syndrome associé à c' .

Décodage par le syndrome

- À partir de la connaissance de H et de c' , on peut donc espérer déterminer le bruit b , car $s = c'H^t = Hb^t$.
- De plus, $s = Hb^t = H_j$, où H_j est la $j^{\text{ème}}$ colonne de la matrice H .
- On connaît donc la position de l'erreur, et il n'y a plus qu'à corriger le mot de code reçu, noté c .
- Une fois le bruit connu, c se déduit simplement par $c = c' \oplus b$

Où \oplus est la fonction OU exclusif (la valeur 1 seulement si les deux opérandes ont des valeurs distinctes).

Example

Code binaire de Hamming [7,4,3]

Matrice génératrice:

$$G = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Matrice de vérification de parité:

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

Example

Code binaire de Hamming [7,4,3]

Message: $m = (1011)$

Encodage:

$$G.m^t = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

Mot de code c

Example

Code binaire de Hamming [7,4,3]

Message: $m = (1011)$

Encodage:

$$G.m^t = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

Bits de parité

Example

Code binaire de Hamming [7,4,3]

Mot de code: $c = (0110011)$

Décodage:

$$H.c^t = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \boxed{\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}}$$

Syndrome

Pas d'erreur!

Example

Code binaire de Hamming [7,4,3]

Mot de code: $c' = (0110111)$


Décodage:

$$H.c^t = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \boxed{\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}} \text{ erreur}$$

Syndrome non nul


Example

Code binaire de Hamming [7,4,3]

$$\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \quad H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$


**Erreur à la
5ème position**

Correction d'erreur:

$$c = c' + b = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$


Erreur corrigé

Liste de CCE

- Codes de Hamming (ont été construits en 1946)
 - Code de Hamming $[7,4,3]$
 - Code de Hamming $[15,11,3]$
 -
- Code de Golay (ont été construits en 1949)
 - Binaire $[23,12,7]$ (code binaire étendu $[24,12,8]$)
 - Ternaire $[11,6,5]$ (code ternaire étendu $[12,6,6]$)
- Code de Reed-Solomon

Liste de CCE

- **Code de Reed-Solomon (ont été construit en 1960)**

$$n=k+2e$$

$$n=2^m-1$$

- Avec : m : nombre de bits par symbole ;
 k : nombre de symboles d'information, appelé charge utile ;
 n : nombre de symboles transmis (charge utile et correction d'erreur) ;
 $2e$: nombre de symboles de contrôle.
- En général, un symbole sera un octet et donc $m=8$ (un octet fait huit bits). Souvent, on a $e = 8$ (un octet), $n = 255$ et $k = 239$.
- Stockage de données (en DVD, en codes à barres), transmission par satellite, utilisation dans un modem avec codeur convolutif (ADSL).

Plan

- Introduction
- Théorie
- **QR code**
- Enrichissement de codes graphiques
- Codes graphiques pour authentification

L'histoire de QR code

- Quick Response code = QR code
- QR code été créé par l'entreprise japonaise Denso-Wave en 1994
- QR code est rendu public en 1999
- QR code a 40 versions:

V1: 21x21 modules → V40: 177x177 modules

Version de QR code



Version 1, 21×21, 10-25 caractères.



Version 2, 25×25, 20-47 caractères.



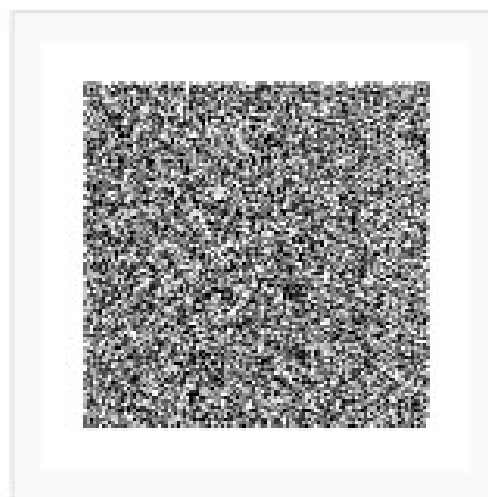
Version 3, 29×29, 35-77 caractères.



Version 4, 33×33, 67-114 caractères.



Version 10, 57×57, 174-395 caractères.

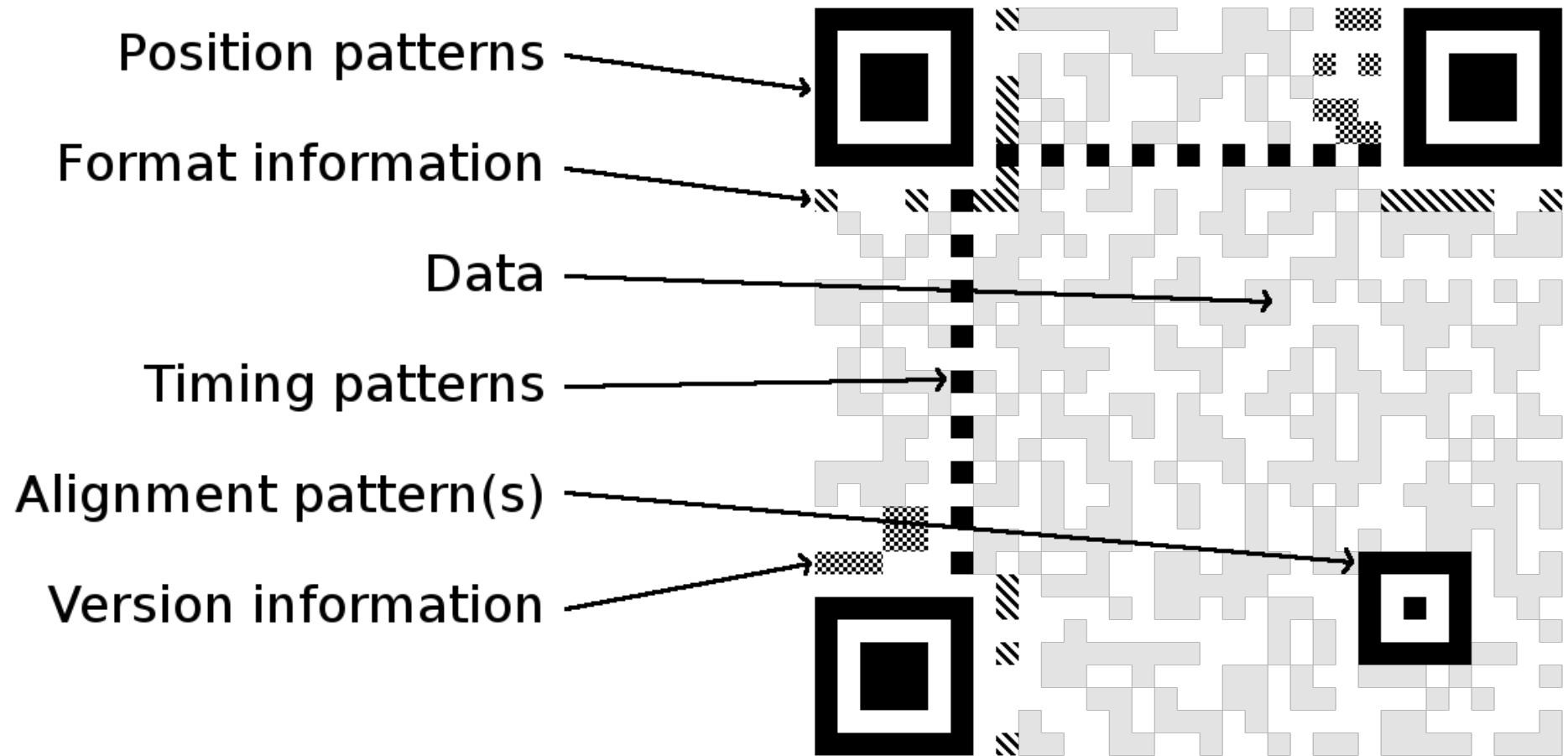


Version 40, 177×177,
1 852-4 296 caractères.

Caractéristique de QR code

- Encodage de données en utilisant le code correcteur d'erreurs de Reed-Solomon.
- 4 niveaux de correction d'erreurs:
 - Niveau L récupère 7% du mot de code;
 - Niveau M récupère 15% du mot de code;
 - Niveau Q récupère 25% du mot de code;
 - Niveau H récupère 30% du mot de code.
- La capacité de stockage la plus élevée correspond au niveau le plus bas de correction d'erreurs.

Structure de QR code

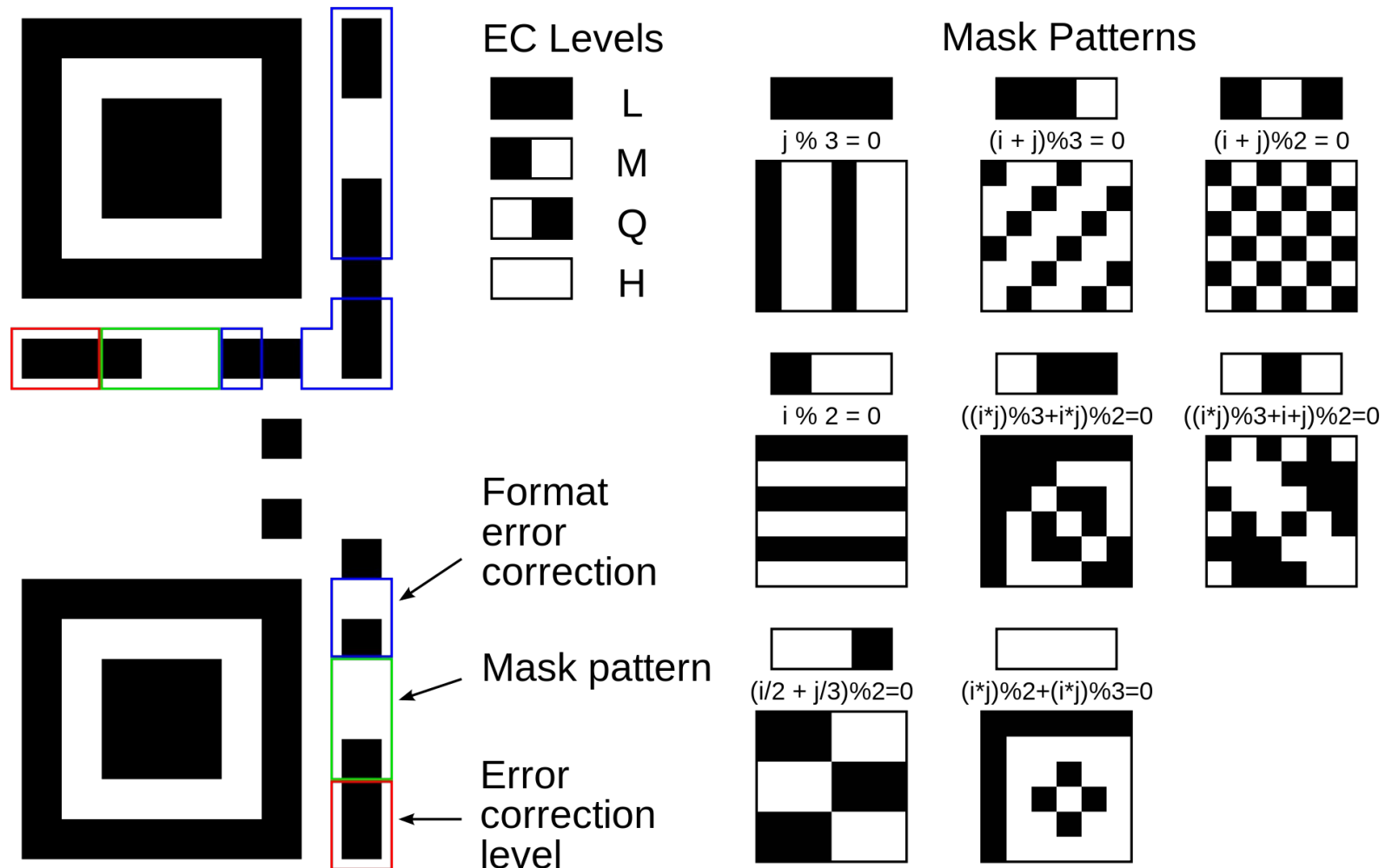


Génération de QR code

- **Encodage des données**
 - des données sont divisées en blocs par 8 bits,
 - des bits de code de correction d'erreur sont ajoutées à chaque bloc (code de RS).
- **Application de masque**
- **Placement des mots de code de droite à gauche dans un motif en zigzag**
 - Dans les versions les plus hautes de code QR, le placement des mots de code est plus complexe en raison de la présence de motifs d'alignement.
- **Ajout des motifs fonctionnels**

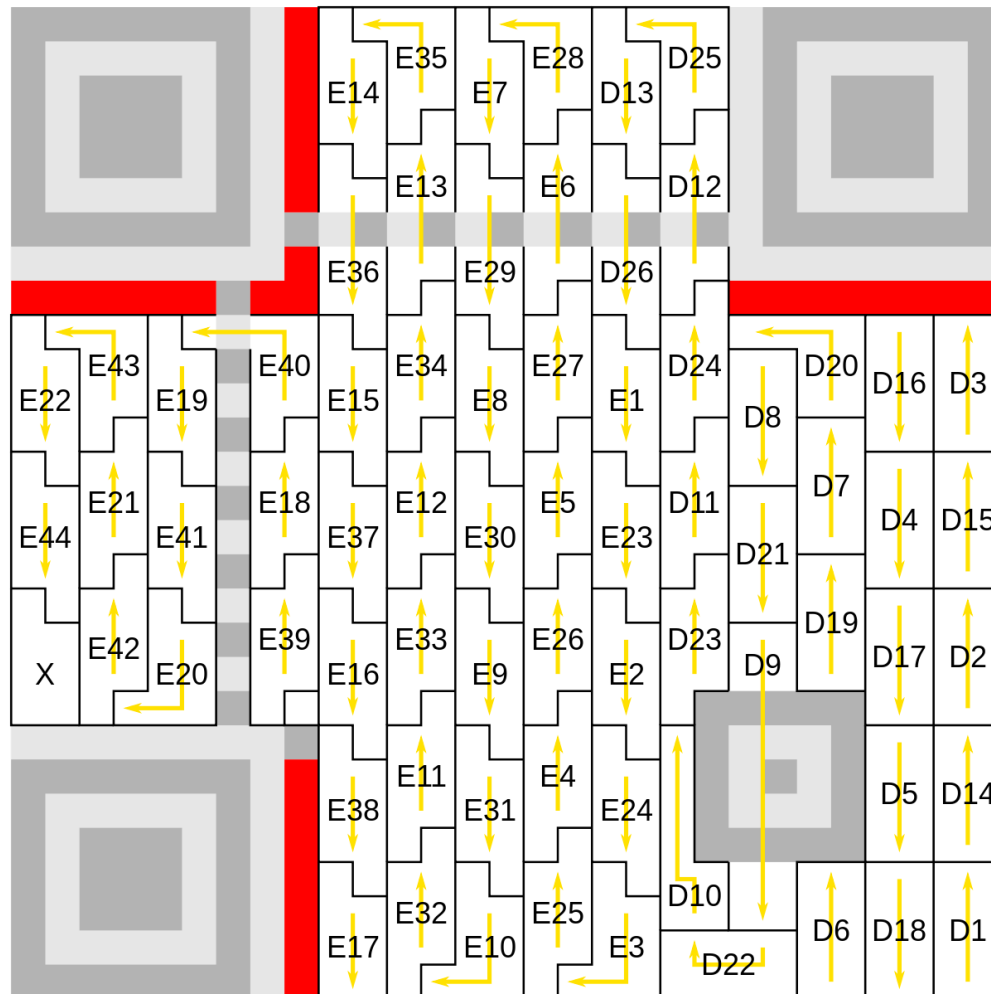
Génération de QR code

Information de format

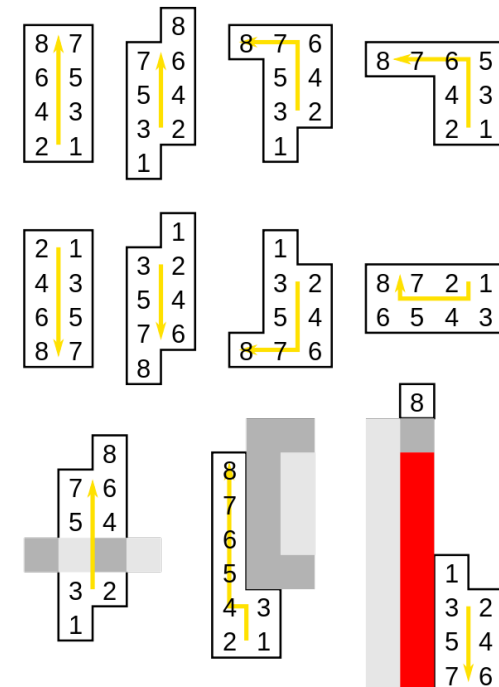


Génération de QR code

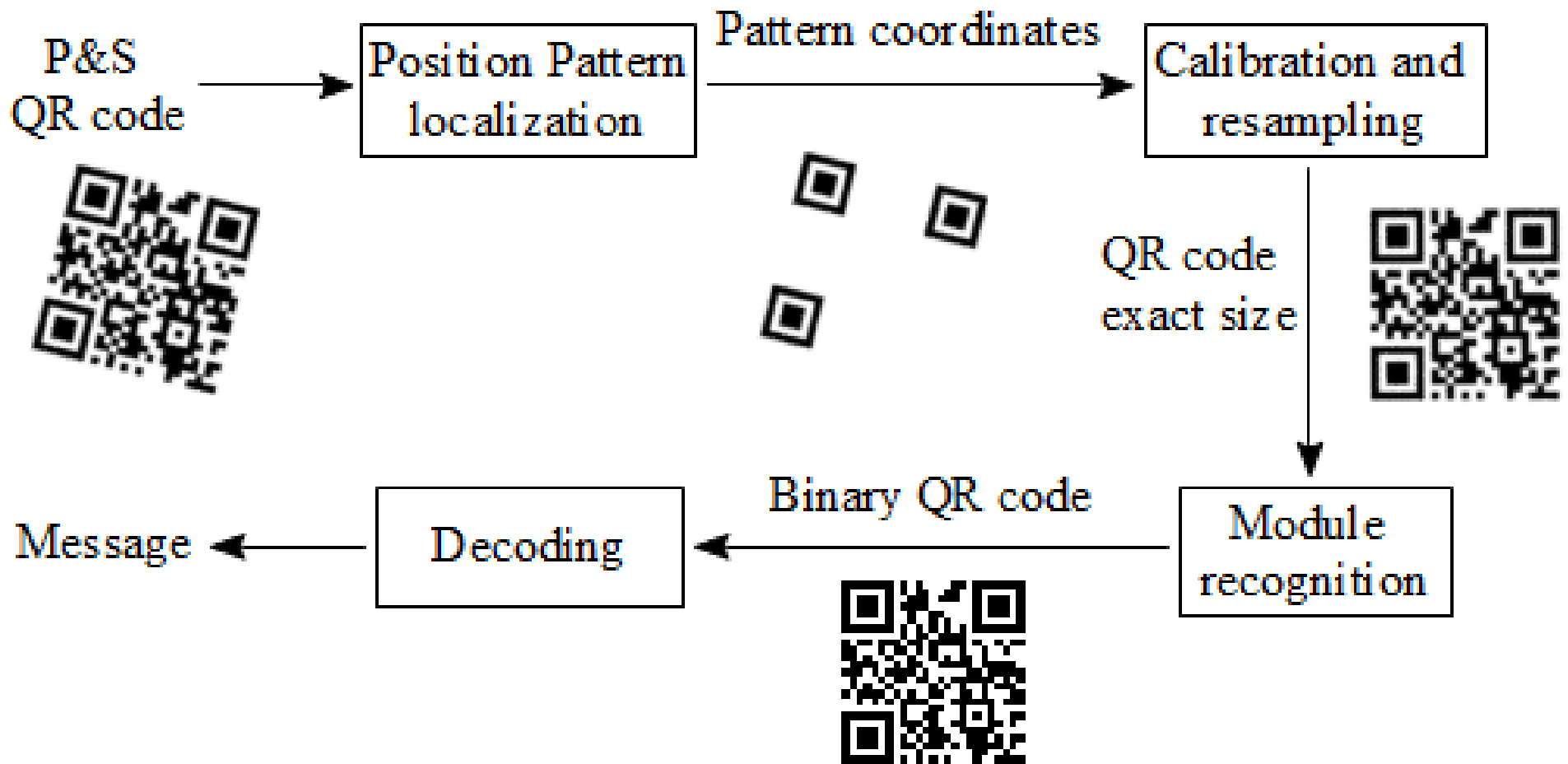
Emplacement des mots de codes



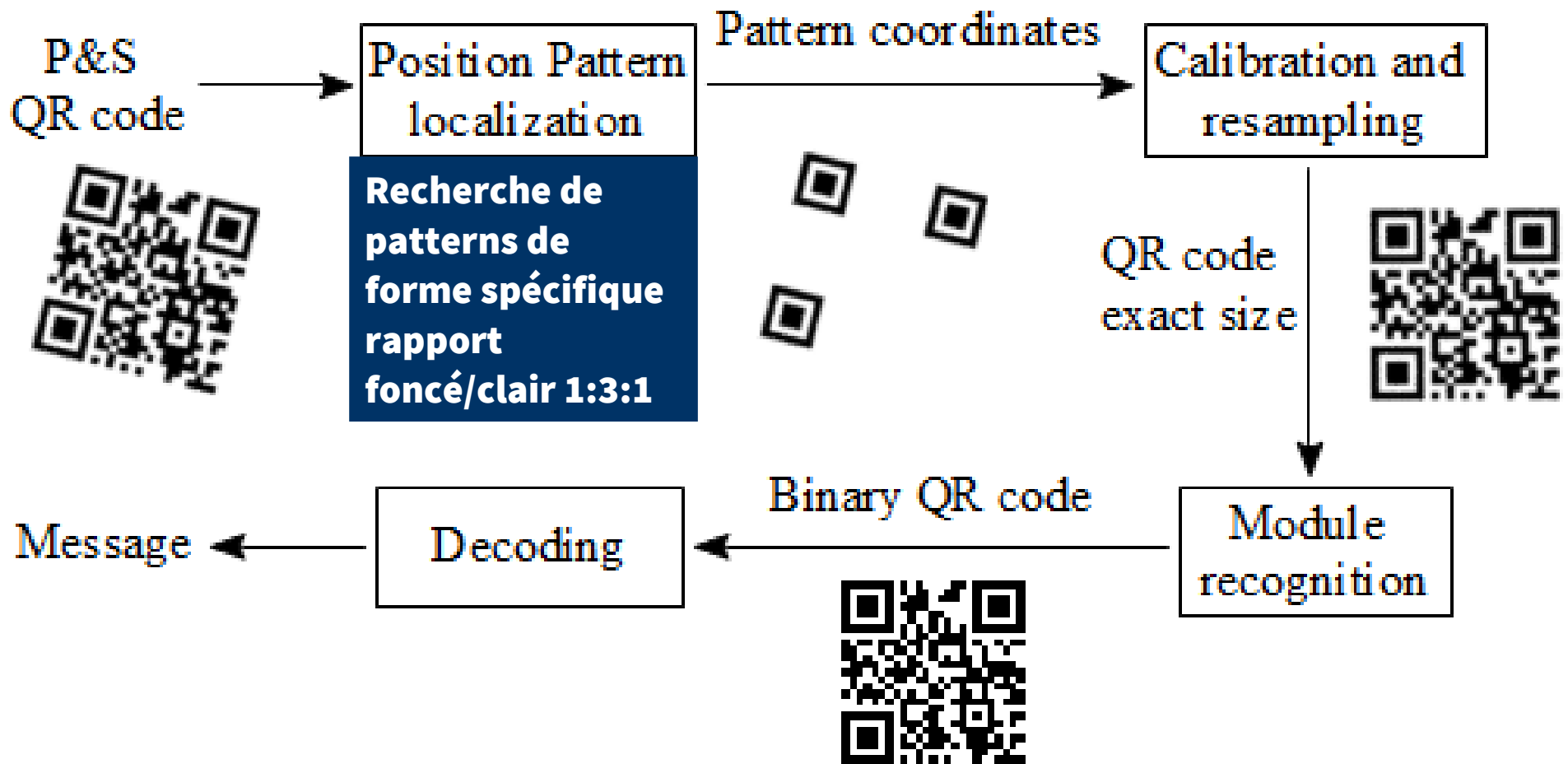
Fixed Patterns Format Info
 D: Data, E: Error Correction, X: Unused
 Error Correction Level H is shown
 Block 1 Codewords: D1–D13, E1–E22
 Block 2 Codewords: D14–D26, E23–E44
 Message Data: D1–D13, D14–D26
 Bit order (1 is the most significant bit):



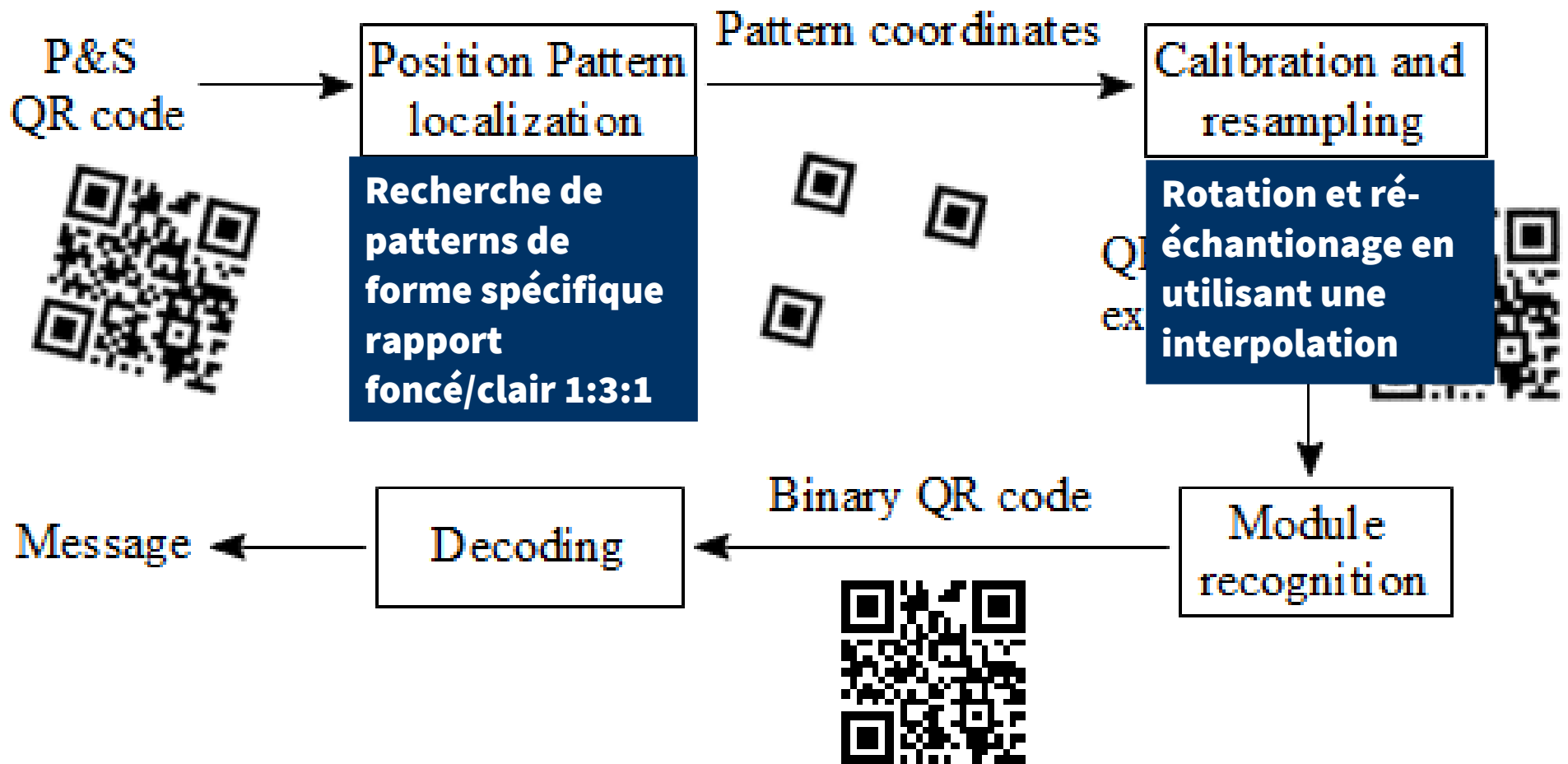
Lecture de QR code



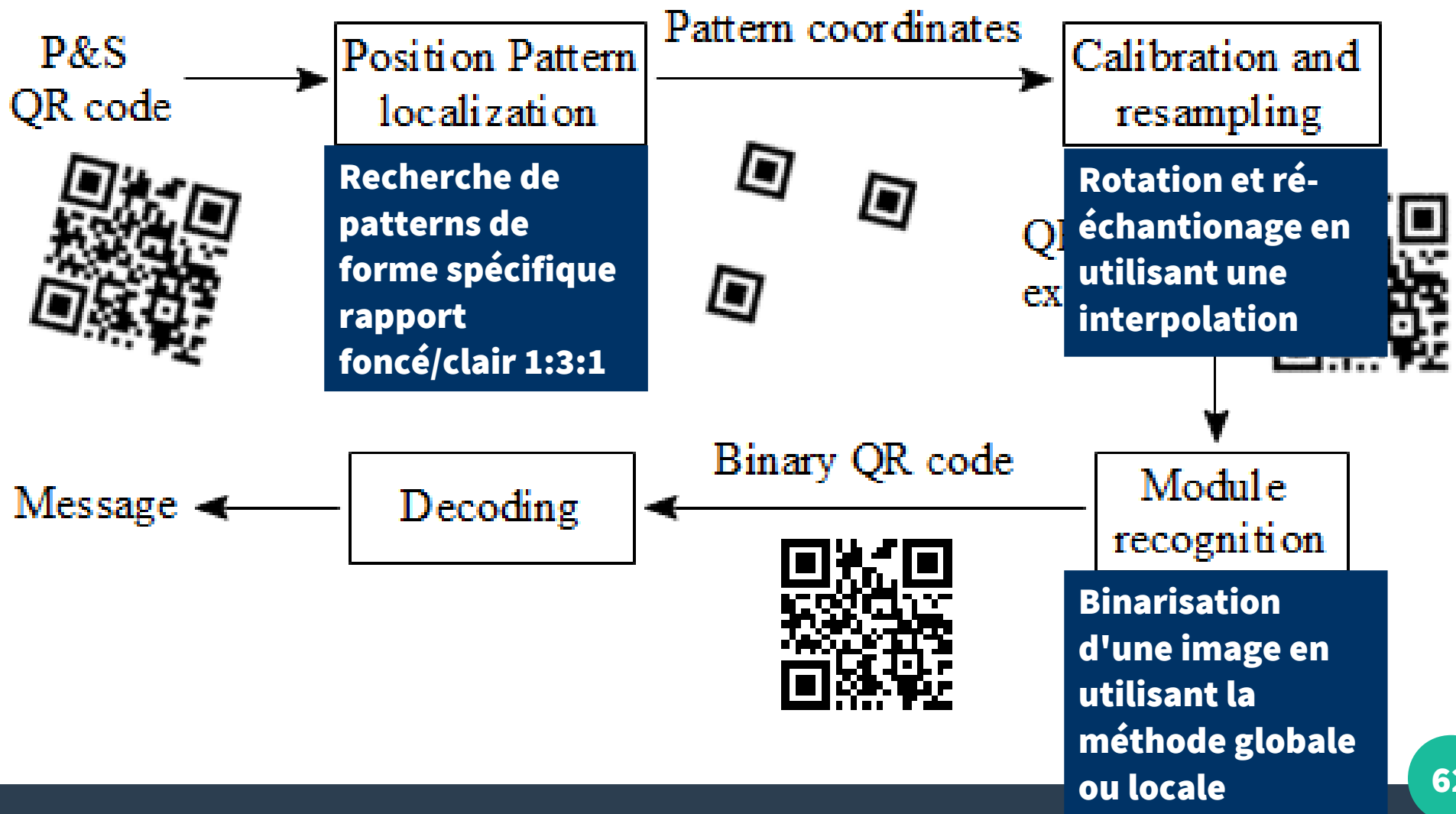
Lecture de QR code



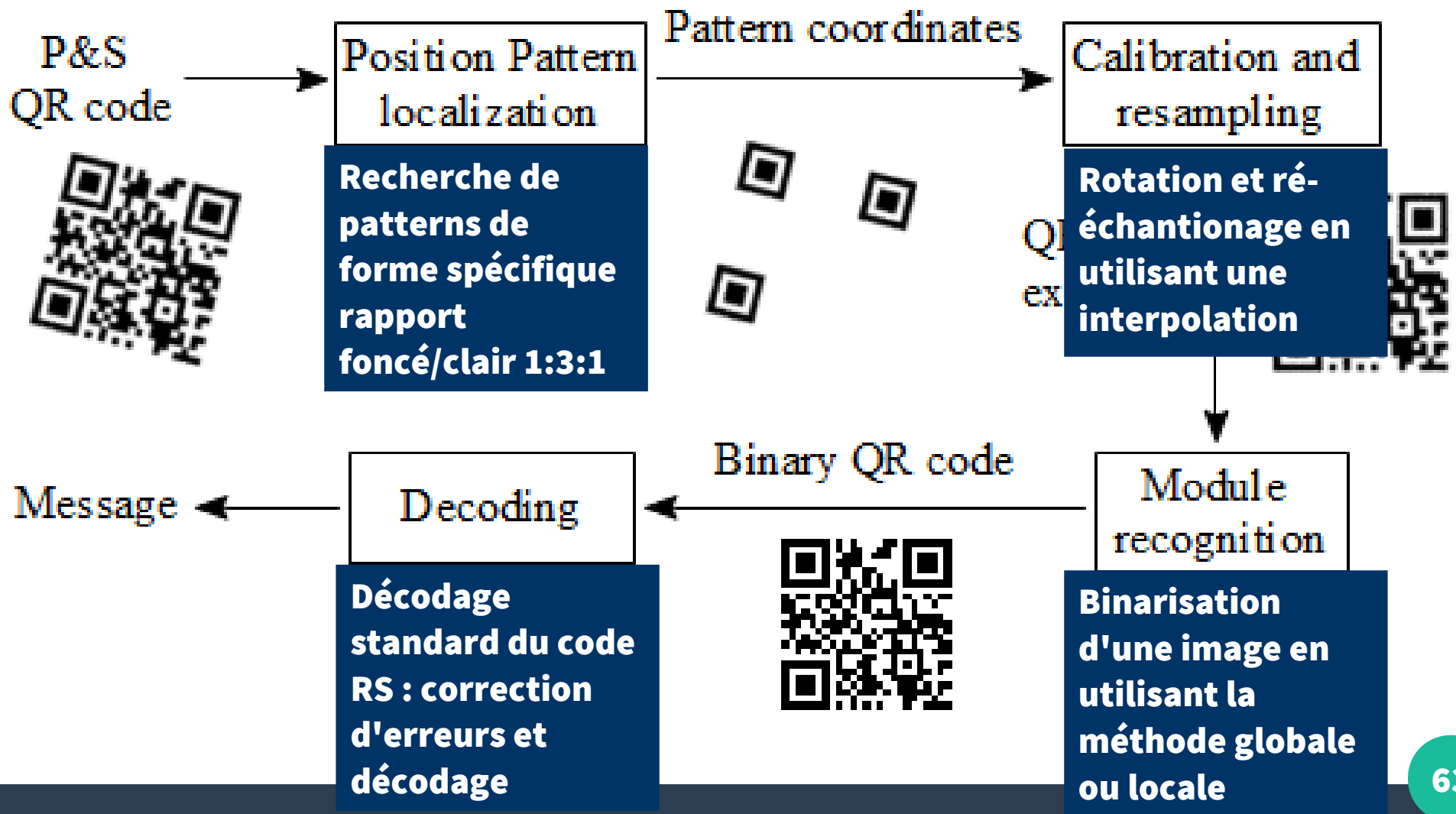
Lecture de QR code



Lecture de QR code



Lecture de QR code



Problèmes de lecture

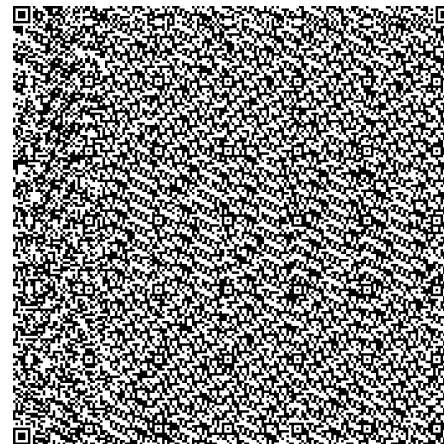
- conditions d'éclairage non homogènes



Binarisation



- Petite taille des modules



Version 40
Taille des modules
3x3 pixels
Taille du code
2.2x2.2 cm²

Problème de lecture

- Image flou
- Impact du système d'acquisition
(camera, scanner)
 - Géométrie
 - Décalage des modules

