

TS 225 – Projet Image

Lecture de code-barres
par lancers aléatoires de rayons

1) Préambule

1.1) Objectifs

Les applications concernant des images ou de la vidéo sont extrêmement nombreuses. De manière générale, elles sont également relativement complexes à mettre en œuvre. Le projet a pour but d'appliquer les enseignements fondamentaux de traitement des images vus au cours du semestre dans un contexte applicatif réel et non trivial.

1.2) Mode d'évaluation

Le projet sera effectué par groupe de 4 élèves (ou 5 élèves selon les effectifs) pendant 13h20 et sera mené sous *Python*.

L'évaluation du travail comporte deux volets : un travail continu et un rapport d'environ 20 pages.

Le rapport doit être rendu dans un format pdf. Il est important de faire figurer les différentes images de la chaîne complète de traitement. Les images accompagnant les commentaires des résultats doivent notamment être choisies et présentées de sorte à mettre en évidence les phénomènes mentionnés (prendre garde au choix de la palette et des échelles de couleurs, aux grandissements, aux graphiques en surimpression, aux légendes, etc.). Les équations doivent être générées avec un éditeur d'équations, en utilisant une police de caractères homogène avec le reste du document, et être numérotées afin d'être référencées dans le texte. Les commentaires doivent être pertinents et tout résultat doit être justifié. Les programmes seront mis en annexe en double colonne en utilisant une police de caractères relativement petite tout en restant suffisamment lisible. Les rapports devront également comporter un bilan de l'organisation et du déroulement du projet qui comprendra, en particulier, la liste des tâches entreprises avec une évaluation du temps passé par chacun des membres du groupe lors de chaque séance et, le cas échéant, entre les séances.

La qualité du code produit et une bonne organisation seront des points pris en compte pour la note de travail continu.

Une interface *Matlab* peut également être proposée et sera comptabilisée en tant que bonus.

1.3) Contact

Le projet est encadré par M. Donias (Bureau S222, mél : marc.donias@enseirb-matmeca.fr).

2) Lecture de code-barres

2.1) Contexte

La plupart des marchandises (denrées alimentaires, biens de consommation, lettres et colis, etc.) et, à présent, un certain nombre de documents (facture téléphonique, avis d'imposition, etc.) sont dotés d'un identificateur numérique et graphique sous la forme d'un code à barres. Celui-ci permet, par exemple, de garantir un suivi lors des phases de production, de transport ou encore de commercialisation.

L'objectif du projet est de réaliser la lecture de codes-barres à partir d'images numériques. Dans un premier temps, le code-barres considéré sera photographié de sorte à être « parallèle » aux bords de l'image et sera décodé à l'aide d'une démarche algorithmique nécessitant la supervision d'un opérateur. Une méthode permettant de traiter le cas d'un code-barres orienté de manière quelconque sera développée dans un second temps. Eventuellement, la levée complète de la supervision sera envisagée au moyen de la détermination automatique de l'emplacement probable du code-barres.

2.2) Code EAN 13

Un des codes-barres le plus répandu est le code EAN (European Article Numbering) dont les spécifications sont gérées par le GS1. La version EAN 13 (voir Figure 1) qui est inscrite sur la quasi-totalité des produits disponibles dans les centres commerciaux est la plus utilisée en France : elle permet notamment de déterminer le prix à payer lors du passage à la caisse.



Figure 1 – Exemple de code-barres EAN 13.

La norme complète est détaillée sur de nombreux sites Internet. Ce paragraphe a pour objectif d'en rappeler les points essentiels.

Le code EAN 13 se compose de 13 chiffres qui sont codés sous une forme graphique « en barres ». Il est à noter que deux chiffres se particularisent : le premier chiffre n'est pas représenté et se déduit des six suivants, tandis que le dernier chiffre fait également office de clé de contrôle et permet de garantir la validité de l'ensemble.

2.2.1) Organisation spatiale

Le code complet se compose de cinq parties distinctes (voir Figure 2) dont chacune a une largeur multiple d'une unité de base :

- Une région de garde (normale à gauche) de largeur égale à 3 fois l'unité de base.

- Une région de six chiffres dont chacun occupe une largeur égale à 7 fois l'unité de base.
- Une région de garde (centrale) de largeur égale à 5 fois l'unité de base.
- Une région de six chiffres dont chacun occupe une largeur égale à 7 fois l'unité de base.
- Une région de garde (normale à droite) de largeur égale à 3 fois l'unité de base.

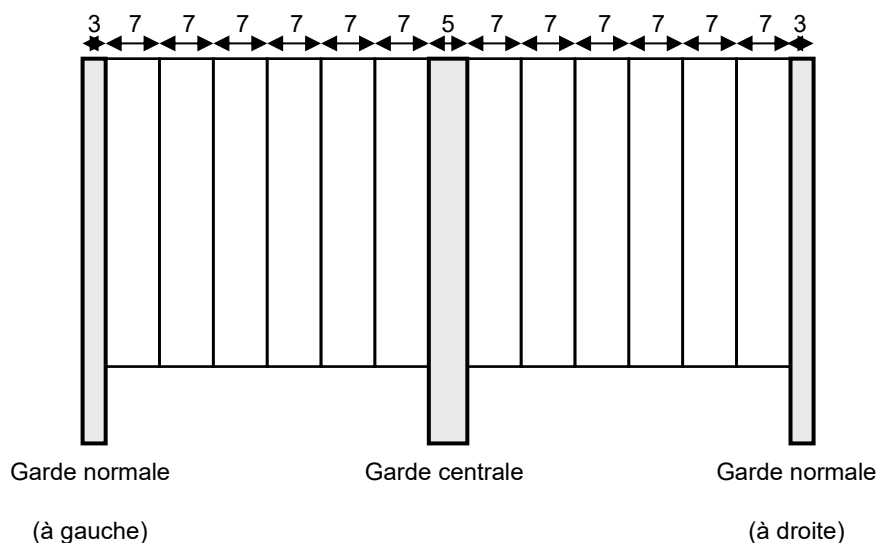


Figure 2 – Régions de garde et de chiffres.

Les régions de garde se composent d'une alternance de 3 ou 5 barres élémentaires noires (N) et blanches (B) de largeur égale à l'unité de base : elles sont représentées par un code « binaire » NBN pour les gardes normales et BNBNB pour la garde centrale.

Au bilan, le code-barres est composé de la juxtaposition de 95 unités de base.

2.2.2) 1.2 Codage d'un chiffre

Chaque chiffre du code-barres, du deuxième au treizième, est également représenté par un code graphique binaire dont la largeur totale est égale à 7 fois celle de l'unité de base : le chiffre (0 à 9) est déterminé en fonction de la présence (N) ou de l'absence (B) de 7 barres noires successives. Les différentes configurations se distinguent en plus par une famille d'appartenance : éléments de type A, B ou C. Elles sont toutes regroupées dans la table suivante.

	Elément A	Elément B	Elément C
0	BBBNNBN	BNBBNNN	NNNBBNB
1	BBNNBBN	BNNBBNN	NNBBNNB
2	BBNBBNN	BBNNBNN	NNBNBBB
3	BNNNNBN	BNBBBBN	NBBBBNB
4	BNBBBBN	BBNNBNB	NBNNBBB
5	BNNBBBB	BNNNBBN	NBBNNNB
6	BNBNNNN	BBBBBNB	NBNBBBB
7	BNNBNBN	BBNBBBN	NBBBNBB
8	BNNBNNN	BBBNBBN	NBBNBBB
9	BBBNBNN	BBNBNNN	NNNBNNB

2.2.3) Amorce : codage du premier chiffre

Le premier chiffre se déduit en analysant les familles d'appartenance de la séquence formée par les chiffres de rang 2 à 7 selon la table suivante.

Séquence	1 ^{er} Chiffre	Séquence	1 ^{er} Chiffre
AAAAAA	0	ABBAAB	5
AABABB	1	ABBBAA	6
AABBAB	2	ABABAB	7
AABBBA	3	ABABBA	8
ABAABB	4	ABBABA	9

2.2.4) Clé de contrôle

Le complément à 10 du dernier (13ème) chiffre du code-barres est égal au chiffre des unités du nombre calculé en additionnant la somme des chiffres de rangs impair (1er, 3ème, 5ème, 7ème, 9ème et 11ème) à trois fois la somme des chiffres de rang pair (2ème, 4ème, 6ème, 8ème, 10ème et 12ème).

2.3) Approche par lancers aléatoires de rayons

Dans de nombreuses situations d'achat de marchandises, l'utilisation d'un lecteur laser portable sans fil ou « douchette scanner » par le client est désormais classique : de manière continue, à de multiples reprises, des intensités sont lues le long d'un segment matérialisé par un faisceau lumineux projeté sur le code-barres visé et comparées à la norme EAN13 jusqu'à son identification correcte.

La méthode de lecture de code-barres par traitement d'images numériques qui est présentée ici est très semblable : des segments ou « rayons » issus de tirages ou « lancers » aléatoires simulent les tentatives de visée du client tandis que l'intensité analysée est extraite de l'image optique.

De manière évidente, le nombre de lancers de rayons nécessaires à une lecture correcte et, par conséquent, le temps de calcul de l'approche sont liés à la part en surface que représente le code-barres au sein de l'image traitée. Aussi, préalablement à la lecture, une identification ou « segmentation » des régions probables de présence du code-barres permet de contraindre les tirages aléatoires et de réduire drastiquement leur nombre.

Au bilan, l'approche complète s'articule en deux étapes dont la première n'est pas indispensable et dont la seconde est, par nature, itérative :

- Segmentation en régions d'intérêt (voir 2.3.1)
- Lecture de code-barres
 - Lancer aléatoire d'un rayon (voir 2.3.2)
 - Extraction de la signature le long d'un rayon (voir 2.3.3)
 - Identification des chiffres codés dans la signature (voir 2.3.4)
 - Itération du procédé avec un nouveau tirage en cas d'échec de l'identification

2.3.1) Segmentation en régions d'intérêt

Les codes-barres EAN13 sont composées de lignes verticales parallèles, de largeur variable mais multiple (de 1 à 4) d'une largeur de base, de hauteur plus importante que leur largeur. Cette géométrie particulière induit une configuration spécifique des vecteurs gradients de l'intensité de l'image $\nabla I = (I_x, I_y)$ qui peut être mise en évidence au moyen du tenseur de structure local $T(x, y)$ défini par :

$$T(x, y) = \begin{bmatrix} T_{xx}(x, y) & T_{xy}(x, y) \\ T_{xy}(x, y) & T_{yy}(x, y) \end{bmatrix} \quad (1)$$

avec

$$T_{ab}(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} W(u, v) I_a(x + u, y + v) I_b(x + u, y + v) du dv \quad (2)$$

où a et b désignent indifféremment x ou y et $W(x, y)$ est une fonction bidimensionnelle de pondération du voisinage au point (x, y) .

En pratique, les vecteurs gradients sont calculés au moyen des filtres de Canny (filtres convolutifs correspondants aux dérivées horizontale et verticale d'une gaussienne bidimensionnelle paramétrée par un écart-type σ_G) tandis que la fonction de pondération du voisinage est un passe-bas gaussien (filtre convolutif gaussien bidimensionnel paramétré par un écart-type σ_T).

Les paramètres d'échelle ou d'espace σ_G et σ_T doivent être choisis en fonction des dimensions des codes-barres traités.

Le tenseur de structure local $T(x, y)$ décrit en particulier la dispersion ou la cohérence du champ de gradients au voisinage du point (x, y) . Il s'agit d'une matrice définie semi-positive et diagonalisable. Une mesure de cohérence $D(x, y)$ distinguant les configurations très ordonnées de celles très désordonnées, définie entre 0 et 1, est la suivante :

$$D(x, y) = (\lambda_1(x, y) - \lambda_2(x, y)) / (\lambda_1(x, y) + \lambda_2(x, y)) \quad (3)$$

où λ_1 et λ_2 sont les valeurs propres de T , positives ou nulles, rangées dans l'ordre décroissant.

Cette mesure de cohérence peut être exprimée directement à partir des composantes du tenseur de structure local $T(x, y)$ selon la formulation :

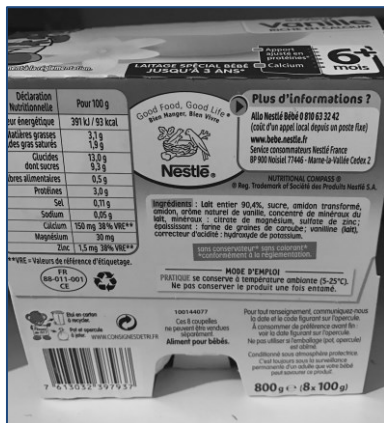
$$D(x, y) = \sqrt{(T_{xx}(x, y) - T_{yy}(x, y))^2 + 4(T_{xy}(x, y))^2} / (T_{xx}(x, y) + T_{yy}(x, y)) \quad (4)$$

Il est à noter que la mesure de cohérence est davantage discriminante en calculant le tenseur de structure local $T(x, y)$ à partir de vecteurs gradients $\nabla_n I$ préalablement normalisés et définis par

$$\nabla_n I = \nabla I / \|\nabla I\| \quad (5)$$

Les régions d'intérêt de l'image traitée c'est-à-dire les lieux probables de présence d'un code-barres peuvent alors être mis en évidence en binarisant la mesure de dispersion $D(x, y)$ au moyen d'un seuil à déterminer. D'une manière générale, la segmentation obtenue (voir Figure 3) fait apparaître :

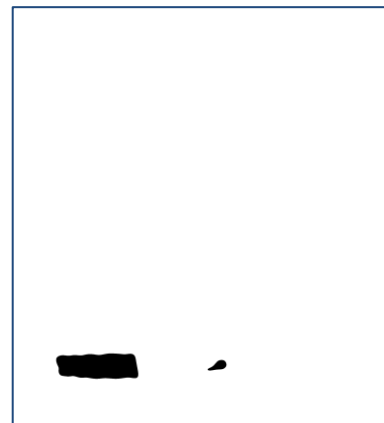
- une empreinte de codes-barres réduite et quasi-rectangulaire selon les paramètres d'échelle choisis,
- des détections parasites par rapport à l'objectif recherché.



(a)



(b)



(c)

Figure 3 – a) image traitée b) mesure de cohérence c) régions probables de présence de code-barres.

2.3.2) Lancer aléatoire d'un rayon

Le lancer aléatoire d'un rayon a pour objectif la détermination d'une ligne qui traverse le code-barres de part en part dans le sens de la « longueur » et qui permet sa lecture. De manière évidente, une ligne adéquate doit traverser le code-barres dans son intégralité, au plus près, sans être d'une longueur trop importante par rapport aux dimensions du code-barres qui rendrait difficile, en particulier, la détermination de ses limites (voir 2.3.3).

Les rayons doivent être inclus dans l'image traitée. Plusieurs méthodes de tirage aléatoire de leurs caractéristiques sont possibles :

- Extrémités « gauche » et « droite »,
- Centre, angle et longueur,
- Extrémité « gauche », angle et longueur,
- ...

Quelle que soit la méthode choisie, chaque rayon est au final défini par deux points de coordonnées respectives (x_1, y_1) et (x_2, y_2) .

Dans l'hypothèse où une identification des régions probables de présence de code-barres a été opérée préalablement, les tirages aléatoires peuvent être contraints de manière efficace en s'appuyant sur les propriétés des régions segmentées (extension, centre de gravité, axes d'inertie, etc.).

Une discrétisation ou échantillonnage de chaque rayon choisi aléatoirement est nécessaire pour permettre la lecture. Les code-barres de la norme EAN13 étant composés de la juxtaposition de 95 unités de base, chaque rayon discrétisé appelé signature doit, par conséquent, être décrit par un nombre minimal de 95 échantillons. La signature s'obtient, tout d'abord, en échantillonnant régulièrement le segment allant du point (x_1, y_1) au point (x_2, y_2) et, ensuite, en interpolant l'image traitée (au plus proche voisin en arrondissant les coordonnées, interpolation bilinéaire en tenant compte des pixels adjacents, interpolation cubique, etc.) en chaque point considéré. Supérieur ou égal à 95, le nombre de points d'échantillonnage L adéquat doit également être en rapport avec le nombre de pixels traversés par le segment.

2.3.3) Extraction de la signature le long d'un rayon

La signature du code-barres obtenue par lancer aléatoire de rayon ne peut pas être confrontée immédiatement à la norme EAN13 afin d'identifier les chiffres recherchés : d'une part, les valeurs des échantillons sont réelles ou entières et ne sont pas binaires et, d'autre part, la signature contient des échantillons superflus qui ne concernent pas le code-barres proprement dit.

Aussi, l'extraction de la signature du code-barres le long d'un rayon comporte plusieurs étapes :

- Extraction d'une première signature par échantillonnage de l'intensité de l'image le long du rayon initial,
- Estimation d'un seuil de binarisation de la signature,
- Détermination des limites « gauche » et « droite » de la partie utile de la signature binarisée,
- Estimation des extrémités d'un rayon utile correspondant aux limites « gauche » et « droite »,
- Extraction d'une seconde signature par échantillonnage de l'intensité de l'image le long du rayon utile,
- Binarisation de la seconde signature.

La deuxième étape et la troisième étape ont pour objectif d'éliminer d'éventuelles parties superflues de la signature, respectivement situées à l'extrême gauche et à l'extrême droite : il s'agit ici d'identifier, de manière très précise, le début et la fin du code-barres qui sont indiqués par les deux régions de garde normale. La méthode proposée consiste à binariser (segmentation en deux classes) la signature : les positions recherchées (barres noires) correspondent alors respectivement au premier et au dernier échantillon de la première classe. Le seuil permettant la binarisation est calculé avec l'algorithme de Otsu qui s'appuie sur le calcul préalable d'un histogramme h sur N classes (usuellement et dans notre cas, $N=256$). Le seuil d'Otsu est directement lié à l'indice de classe k qui maximise le critère suivant :

$$crit(k) = w(k)[\mu(N-1) - \mu(k)]^2 + (1 - w(k))\mu(k)^2 \quad (6)$$

$$\text{avec } w(k) = \frac{\sum_{i=0}^k h(i)}{\sum_{i=0}^{N-1} h(i)} \text{ et } \mu(k) = \frac{\sum_{i=0}^k i \cdot h(i)}{\sum_{i=0}^{N-1} i \cdot h(i)}.$$

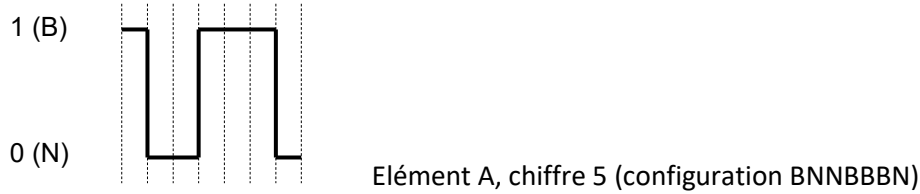
De manière évidente, l'identification des chiffres codés dans la signature est plus aisée si le nombre d'échantillons qui la composent est un multiple de 95. La quatrième étape et la cinquième étape ont pour objectif de produire une nouvelle signature vérifiant cette propriété. Il s'agit, tout d'abord, le long du rayon initial, d'estimer la position de points limites virtuels $(\tilde{x}_1, \tilde{y}_1)$ et $(\tilde{x}_2, \tilde{y}_2)$ correspondants au premier et au dernier échantillon de la signature utile. Ensuite, une nouvelle signature est extraite le long du rayon utile défini par $(\tilde{x}_1, \tilde{y}_1)$ et $(\tilde{x}_2, \tilde{y}_2)$. Le nombre de points d'échantillonnage \tilde{L} , tout comme dans le cas de la première signature, doit être en rapport avec le nombre de pixels traversés par le segment utile et doit, ici, vérifier $\tilde{L} = 95u$ où u est un coefficient entier représentant un nombre d'échantillons par unité de base. La signature s_{CB} permettant l'identification des chiffres est obtenue en binarisant la nouvelle signature à l'aide du seuil d'Otsu déjà estimé.

2.3.4) Identification des chiffres codés dans la signature

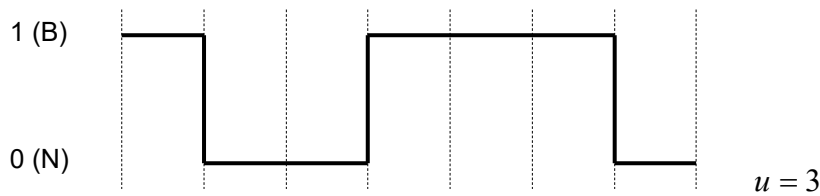
La signature complète s_{CB} du code-barres de longueur \tilde{L} est segmentée en douze parties ou signatures partielles représentant chacune un chiffre. La découpe est opérée en fonction du choix préalable de l'unité de base u et de l'organisation spatiale décrite précédemment (voir 2.2.1).

Chaque signature partielle ainsi obtenue, notée s_p , est confrontée à l'ensemble des signatures théoriques permises (3 familles de 10 configurations) afin d'identifier le chiffre candidat le plus probable. A cette fin, pour chaque configuration possible, trois opérations sont réalisées :

- Construction d'une signature théorique s_{th} de longueur 7.



- Dilatation de la signature théorique s_{th} en fonction de l'unité de base u choisie.



- Mesure de ressemblance de la signature théorique avec la signature partielle observée par comptage du nombre de différences selon la formulation

$$c(s_{th}, s_p) = \|s_{th} - s_p\| \quad (7)$$

où $\| \cdot \|$ désigne la norme.

Conformément à la norme EAN 13, le premier chiffre du code-barres est déterminé en fonction des six suivants tandis que le dernier chiffre peut donner lieu au calcul de la clé de contrôle pour valider le code dans son intégralité.

3) Travail à réaliser

Il s'agit d'implémenter la technique proposée sous *Python*, de justifier de son fonctionnement correct et de vérifier sa robustesse à travers, par exemple, les étapes des sections suivantes.

3.1) Jeux de tests

Construire une base de données de codes-barres à partir de plusieurs sources : bases de données déjà existantes, appareils photographiques numériques autonomes ou embarqués au sein de téléphones portables, numériseurs, caméras numériques, etc. Veiller à ce que les codes-barres retenus aient des caractéristiques diverses et représentatives de conditions réelles.

3.2) Extraction de la signature le long d'un rayon

A des fins de mise au point, la démarche peut s'appuyer sur une initialisation manuelle : à la souris, l'opérateur trace un segment ou rayon traversant, au plus près, de part en part, le code-barres dans sa « longueur ».

Calculer la longueur du rayon et extraire une première signature.

Calculer le critère d'Otsu et observer sa forme. Estimer un seuil de binarisation.

Binariser la signature, identifier les échantillons limites de la partie utile, et déterminer les coordonnées des points définissant le rayon utile.

Calculer la valeur adéquate de l'unité de base u , extraire et binariser une nouvelle signature le long du rayon utile

3.3) Identification des chiffres

Décomposer la signature en ses différentes parties.

Identifier les chiffres du code-barres et afficher les mesures de ressemblance associées.

Déterminer le premier chiffre puis valider l'ensemble du code à l'aide de la clé de contrôle

3.4) Approche automatique

Implémenter des tirages aléatoires de rayons jusqu'à obtenir une lecture correcte.

Afficher le nombre de tirages aléatoires nécessaires et vérifier que les mesures de ressemblance sont suffisantes.

3.5) Approche optimale

Calculer la mesure proposée de cohérence du champ de vecteurs gradients normalisés.

Identifier un seuil permettant de déterminer les régions d'intérêt correspondant aux lieux probables de code-barres.

Contraindre les tirages aléatoires de rayons par les caractéristiques des régions d'intérêt.

3.6) Performances

Traiter l'intégralité d'une base de données.

Analyser les résultats obtenus et déterminer les conditions requises pour l'utilisation de la méthode proposée : qualité de l'image, grandissement, éclairage ambiant, etc.

Bonus : implémenter une interface graphique mettant en œuvre l'ensemble des traitements.