

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la recherche Scientifique Université

Benyoucef Benkhedda-Alger 1

Faculté des Sciences

Département Informatique



Module: Introduction Au Traitement D'Images

MASTER 1 ISII

**Développement d'un système de comptage des
pièces de monnaies**

Réalisé par :

Ounis Akram G3

Belhaddad Chawki G3

Charef Sara G2

Année universitaire 2024/2025

TABLE DES MATIÈRES :

1. Introduction

- Contexte et objectifs du projet

2. Étapes du système de comptage

- Chargement de l'image
- Prétraitement des images
- Segmentation
- Filtrage
- Méthodes de détection et d'analyse
- Interface utilisateur

3. Prétraitement des images

- Amélioration du contraste
- Réduction du bruit
- Amélioration de la dynamique
- Ajustement de la saturation
- Troncature des valeurs

4. Segmentation des pièces

- Seuillage classique
- Méthode d'Otsu

5. Méthodes de détection et d'analyse

- La fonction `binarize_image`
- La fonction `detect_large_circles`
- La fonction `display_results`
- La fonction `dynamic_filter_image`
- La fonction `calculate_accuracy`
- La fonction `dynamic_filter_image`

6. Filtrage

- Filtre médian
- Filtre de moyenne arithmétique
- Filtre gaussien
- Filtre Laplacien
- Filtre de Sobel

7. Visualisation des résultats

8. Visualisation de l'interface

9. Conclusion

10. Bibliographie

1-INTRODUCTION:

Le domaine de la vision par ordinateur, qui combine mathématiques, informatique et intelligence artificielle, a connu une évolution fulgurante au cours des dernières décennies. Grâce aux avancées dans les techniques de traitement d'images et d'apprentissage profond, il est désormais possible de résoudre des problèmes complexes avec une grande précision et rapidité. Parmi ces technologies innovantes, la détection d'objets occupe une place centrale. Cette technique permet non seulement d'identifier les objets présents dans une image ou une vidéo, mais également de localiser leur position, de suivre leurs mouvements et de les classer en catégories spécifiques.

Ces capacités font de la détection d'objets une technologie clé dans de nombreuses applications, telles que les systèmes de tri automatique, l'inspection de qualité dans l'industrie manufacturière, la reconnaissance d'objets dans des environnements complexes ou encore l'assistance à la navigation autonome. Inspirés par ces applications variées, nous proposons d'appliquer ces avancées technologiques à un problème concret et pratique : le comptage et la classification des pièces de monnaie à partir d'images numériques.

Ce projet a pour objectif de développer un système automatisé, précis et efficace, capable de détecter des pièces de monnaie dans une image, de les segmenter pour les isoler du fond et de les classer en fonction de leur taille ou de leur valeur nominale. En combinant des techniques de vision par ordinateur et d'apprentissage automatique, ce projet vise à relever les défis liés au traitement d'images dans un contexte pratique.

Pour rendre ce système accessible et convivial, une interface graphique interactive, développée en Tkinter, sera intégrée. Cette interface permettra à l'utilisateur de charger des images, d'afficher les résultats de la segmentation et de visualiser directement les comptages réalisés par le système. Sur le plan technique, l'ensemble de la solution sera développé en Python, offrant ainsi flexibilité et puissance pour répondre aux exigences du projet.

2-Étapes du système de comptage :

1-Chargement de l'image : Charger l'image à l'aide de la bibliothèque Pillow, puis la convertir en tableau NumPy pour simplifier le traitement des pixels.

2-Prétraitement : L'une des premières étapes de la reconnaissance des formes par ordinateur consiste à extraire des primitives de bas niveau à partir d'images. Ces primitives sont souvent affectées par du bruit, rendant nécessaires des étapes de prétraitement pour améliorer la qualité des données. En comparant les notions de connexité et de convexité dans les espaces continus et discrets, nous proposons un ensemble de techniques de prétraitement visant à améliorer les propriétés de connexité de l'image.

3-Segmentation : est un processus qui consiste à découper une image en régions connexes présentant une homogénéité selon certains critères, afin de séparer et d'extraire les entités d'une image, dans le but de leur appliquer un traitement spécifique. Elle est réalisée selon deux approches duales : **l'approche contour** correspond à une variation d'intensité ou à une discontinuité entre les propriétés de deux ensembles de points. Et **l'approche région** : correspond à un ensemble de pixels connexes ayant des propriétés communes qui les différencient des autres régions. Dans ce projet, nous considérons le problème de la segmentation d'une image contenant des **pièces d'argent**, que l'on souhaite segmenter en deux régions : celles contenant des pièces d'argent et celles représentant le fond.

4-Filtrage : Le filtrage d'image a pour but d'améliorer la qualité d'une image numérique. Chaque filtre vise à atténuer un type de défauts plus précis. Il n'y a pas de filtre universel capable de corriger tous les défauts, Il est donc nécessaire de choisir les bons filtres en fonction des défauts que nous désirons corriger. Il existe deux types de filtrage :

- Filtrage linéaire : ou la transformation d'un pixel est le fruit d'une combinaison linéaire des pixels voisins.
- Filtrage non linéaire : ou les pixels voisins interviennent suivant une loi non linéaire.

5- Méthodes de détection et d'analyse : Cette section présente les techniques utilisées pour détecter et analyser les objets d'intérêt dans l'image. Elle inclut la binarisation, la détection des cercles, et l'évaluation de la précision des résultats obtenus. De plus, cette section explique comment la fonction `dynamic_filter_image` gère l'ensemble du traitement d'image en automatisant des étapes clés telles que le prétraitement, filtrage...

6-Interface utilisateur : Permet à l'utilisateur de télécharger une image, d'appliquer des algorithmes de traitement d'image et de visualiser les résultats de manière simple et intuitive:

- * Affichage côte à côte de l'image originale et traitée.
- * Nombre de pièces détectées et réel, avec un pourcentage d'exactitude.
- * Nombre de pièces dans chaque catégorie (petites, moyennes, grandes).

3-Prétraitement des images :

1- Amélioration du Contraste : C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. Il est défini en fonction des luminances de deux zones d'images.

Le contraste se définit par le rapport :

$$L' = g(l) = K_{min} + \frac{K_{max} - K_{min}}{l_{max} - l_{min}} \times (l - l_{min})$$

[lmin, lmax] : Les valeurs minimales et maximales des niveaux de gris de l'image initiale.

[kmin, kmax] : Les nouvelles limites de l'intervalle des niveaux de gris souhaité.

l : Le niveau de gris actuel à normaliser

2-Réduction du bruit : la qualité d'une photo peut être dégradée par du **bruit numérique**, c'est-à-dire par l'apparition aléatoire de "grains" superflus. Il peut être vu comme une image constituée de pixels dont les intensités ont été déterminées de manière aléatoire. Nous pouvons appliquer des opérations mathématiques usuelles, comme l'addition. Ainsi, on parle de **bruit additif** (ex :le bruit gaussien) lorsque l'image bruitée est la somme de l'image originale et du bruit, L'intensité du pixel de coordonnées (x,y) dans l'image bruitée est alors donnée par la relation: $I'(x,y) = I(x,y) + \eta(x,y)$ où $I'(x,y)$, $I(x,y)$ et $\eta(x,y)$ désignent les intensités du pixel (x,y) respectivement dans l'image bruitée, l'image originale et le bruit.

Plusieurs techniques de **lissage** ont été développées afin d'atténuer le bruit dans une image. Le **lissage par moyenne** désigne la solution la plus intuitive.

3- Amélioration de la dynamique : Cette méthode élémentaire consiste à utiliser au mieux l'échelle de niveaux de gris disponible sur le système d'acquisition d'image. Soient $A[i, j]$ l'image de départ et $A'[i, j]$ l'image après transformation. Soient $[a_0, a_1]$ l'intervalle des intensités présentes dans l'image et $[a_{min}, a_{max}]$ l'intervalle disponible. L'expansion de dynamique correspond à la transformation linéaire T suivante :

$$a'_s = \alpha + \beta a_s$$

telle que :

$$\forall a \in [a_0, a_1] \quad a \xrightarrow{T} a' \in [a_{min}, a_{max}]$$

avec :

$$\alpha = \frac{a_{min} \cdot a_1 - a_{max} \cdot a_0}{a_1 - a_0} ; \beta = \frac{a_{max} - a_{min}}{a_1 - a_0}$$

Cette transformation ne fait pas qu'améliorer l'aspect visuel de l'image. En effet, les données étant codées sur un nombre limité de bits, généralement 8, on atténue ainsi les erreurs d'arrondi pouvant se propager dans les traitements ultérieurs. Il est alors possible d'éviter de recourir à la représentation des nombres en virgule flottante, coûteuse en temps de calcul et en place mémoire.

4- Ajustement de la Saturation : La saturation d'une image est une technique qui contrôle l'intensité ou la vivacité des couleurs dans une image. Elle permet de renforcer ou d'atténuer les teintes, rendant les couleurs plus éclatantes ou plus ternes, selon les besoins. Une saturation élevée accentue les différences entre les couleurs, tandis qu'une saturation faible les rapproche des niveaux de gris. L'une des méthodes courantes pour ajuster la saturation consiste à appliquer un facteur multiplicatif sur la composante de saturation dans

des modèles de couleurs comme HSV ou HSL, permettant ainsi de moduler la vivacité tout en conservant les informations sur la teinte et la luminosité

Elle fonctionne en ajustant l'intensité des couleurs d'une image à l'aide d'un facteur appliqué à la saturation de chaque pixel. Le processus consiste à multiplier les composantes chromatiques par un facteur donné, ce qui amplifie ou atténue leur vivacité tout en préservant la teinte et la luminosité. Un facteur inférieur à 1 réduit la saturation en rendant les couleurs plus ternes, tandis qu'un facteur égal à 0 désature complètement l'image, la convertissant en niveaux de gris. Cette transformation permet de contrôler précisément l'apparence des couleurs dans une image pour répondre aux besoins visuels ou analytiques.

5-Troncature : ou clamping, est une technique utilisée pour limiter les valeurs des pixels à une plage d'intérêt donnée [a,b] Cette opération est essentielle lorsque :

- ✚ **Les valeurs des pixels sont hors plage :** les pixels ayant des intensités très faibles ou très élevées peuvent être ignorés pour éviter des artefacts ou de fausses interprétations.
- ✚ **Une normalisation est requise :** Cette méthode force les intensités à rester dans une plage cohérente pour des analyses ou visualisations spécifiques. La fonction $g(l)$ ci-dessous forcera toutes les valeurs d'intensité de l'image à se situer dans la plage [a,b] :

$$g(l) = \begin{cases} a & \text{si } l < a \\ l & \text{si } a \leq l \leq b \\ b & \text{si } l > b \end{cases}$$

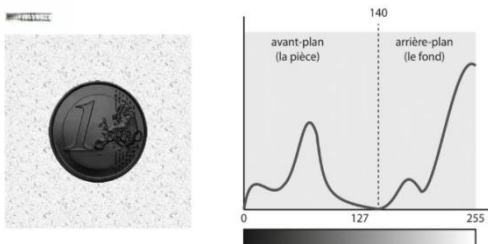
4- Segmentation des pièces :

Le seuillage d'image : est la méthode la plus simple et la plus populaire pour le traitement des images numériques. Ce n'est pas une méthode de segmentation en régions, mais par son approche dite *approche pixel*. À partir d'une image en niveau de gris, le seuillage d'image peut être utilisé pour créer une image comportant uniquement deux valeurs, noir ou blanc (image monochrome).

Le seuillage classique : est une méthode simple de segmentation consiste à utiliser une **valeur de seuillage**, appelée *seuil*, pour partitionner une image en deux parties intitulées avant-plan et arrière-plan. L'exemple de la figure avec une pièce de monnaie sur un fond granuleux clair, si on effectue un seuillage avec un seuil à 140, on sépare l'arrière-plan (fond granuleux) de l'avant-plan (pièce de monnaie).

Dans ce cas précis, l'histogramme représenté est sans ambiguïté. Si $f(x,y)$ représente l'intensité d'un pixel dans l'image, si $g(x,y)$ représente l'intensité de ce même pixel après seuillage et S représente le seuil, on a l'équation de la représentation du seuillage qui s'exprime, dans le cas où le fond est blanc et la pièce est noire par

$$\begin{cases} \text{si } f(x,y) \geq S \text{ alors } g(x,y) = 255 \\ \text{si } f(x,y) < S \text{ alors } g(x,y) = 0 \end{cases}$$



Méthode d'Otsu : essaye de trouver le seuil T qui sépare l'histogramme de façon optimale en deux segments (qui maximise la variance inter-segments ou bien qui minimise une mesure de variance intra-segments). Le calcul de la variance intra-classes est basé sur l'histogramme normalisé $H = [h_0 \dots h_{255}]$ de l'image d'où $\sum h_i = 1$.

La variance inter-classes pour chaque seuil t est donnée par :

$$\sigma_{\text{inter}}^2 = q_1(t) \times q_2(t) \times [\mu_1(t) - \mu_2(t)]^2$$

$$\mu_1(t) = \frac{1}{q_1(t)} \sum_{i=0}^{t-1} h(i) \times i \quad \text{et} \quad \mu_2(t) = \frac{1}{q_2(t)} \sum_{i=t}^{255} h(i) \times i$$

$$q_1(t) = \sum_{i=0}^{t-1} h(i) \quad \text{et} \quad q_2(t) = \sum_{i=t}^{255} h(i)$$

5- Méthodes de détection et d'analyse :

La fonction binarize image : est un processus qui convertit les images originales capturées en couleur ou en niveaux de gris en images numériques binaires composées de petits pixels noirs et blancs.

La binarisation automatique des images est une méthode de détermination du seuil de binarisation avec laquelle le seuil de binarisation est formulé automatiquement sur la base de l'image originale. En plus de la binarisation automatique, les utilisateurs sont autorisés à utiliser la binarisation d'image à double seuil pour mettre en évidence les pixels situés dans une plage spécifique de valeurs de pixels. Dans la binarisation d'image à double seuil, seuls les pixels dont les valeurs sont supérieures ou égales au seuil minimal et inférieures au seuil maximal seront identifiés comme des **pixels clairs**. Les autres pixels seront reconnus comme des **pixels sombres**.

La fonction detect large circles : est une fonction qui identifie les objets circulaires dans une image binaire. Elle se base sur deux critères principaux : la taille (définie par une aire minimale et maximale) et la circularité (un indicateur mathématique mesurant à quel point une région est proche d'un cercle parfait).

$$\text{Circularité} = \frac{4 \cdot \pi \cdot \text{Aire}}{\text{Périmètre}^2}$$

Aire : Nombre de pixels à l'intérieur de la région détectée.

Périmètre : Nombre de pixels en bordure de la région.

Elle utilise une exploration en profondeur (DFS) pour détecter les régions connexes, calcule leurs caractéristiques géométriques (comme l'aire et le périmètre), et évalue leur circularité. Les régions validées sont retournées sous forme de cercles définis par leurs coordonnées centrales et leur rayon estimé (c_y, c_x, r).

c_x et c_y : sont les moyennes des positions des pixels de la région.

r : rayon est estimé comme : $r = \sqrt{\frac{\text{Aire}}{\pi}}$

La fonction display_results: est une méthode utilisée pour visualiser les résultats d'une opération de détection d'objets (cercles) sur une image. Elle prend en entrée une image binaire et une liste de cercles détectés, puis crée une nouvelle image dans laquelle chaque cercle détecté est représenté par un contour coloré en rouge, pour faciliter l'interprétation visuelle.

La fonction dynamic filter image: est responsable de la gestion complète du traitement d'image en automatisant plusieurs étapes clés. Elle commence par le prétraitement de l'image, où la fonction **determiner_besoins_pretraitement** est utilisée pour nettoyer ou segmenter l'image initiale. Ensuite, elle procède à une segmentation avancée, en utilisant la méthode d'Otsu, pour identifier les zones pertinentes de l'image. Une fois l'image segmentée, la fonction analyse ses caractéristiques à l'aide de **determine_filtering_needs** afin de déterminer les filtres les plus adaptés. Ces filtres sont ensuite appliqués dynamiquement grâce à la méthode comme **apply_filtering**.

La fonction calculate_accuracy: est une procédure qui mesure le degré de correspondance entre une valeur détectée et une valeur réelle de référence, exprimé sous forme de pourcentage. Elle permet de quantifier la précision d'un système ou d'un algorithme en comparant les résultats obtenus avec des données attendues, tout en prenant en compte les éventuelles erreurs ou incohérences dans les données fournies.

La fonction detect_coins : a pour objectif de classer des pièces en fonction de leur taille (petites, moyennes ou grandes). Donc elle calcule d'abord la moyenne des rayons des cercles, qui sert de référence pour la classification. Les pièces sont classées de la manière

suivante :
$$\begin{cases} r \approx \text{la moyenne} : \text{moyenne pièce} \\ r < \text{la moyenne} : \text{petite pièce} \\ r > \text{la moyenne} : \text{grande pièce} \end{cases}$$

6-Filtrage:

Filtre médian : est un filtre statistique qui remplace la valeur d'un pixel par la médiane des niveaux d'intensité dans le voisinage de ce pixel. La médiane est calculée sur l'ensemble des pixels dans une zone définie autour du pixel central, appelée voisinage. Ce filtre est particulièrement efficace pour réduire le bruit impulsional (comme le bruit "sel et poivre"), tout en préservant les contours de l'image mieux que les filtres de lissage linéaires de taille similaire. La valeur du pixel au point (x, y) est incluse dans le calcul de la médiane. Ce filtre est particulièrement adapté aux images affectées par du bruit impulsional.

$$I'(u, v) = \text{median} \{ I(u + i, v + j) \mid (i, j) \in R_d \}$$

Filtre de la moyenne arithmétique: est le plus simple des filtres de moyenne. Il consiste à représenter l'ensemble des coordonnées d'une sous-image rectangulaire (voisinage) de taille prédéfinie, centrée au point (x, y). Ce filtre calcule la valeur moyenne des pixels de

l'image corrompue dans la zone définie par ce voisinage. La valeur attribuée au pixel restauré au point (x, y) est alors égale à cette moyenne arithmétique, obtenue à partir des pixels contenus dans la région considérée. Pour un masque de filtre de taille $n \times n$, nous avons :

$$I'(u, v) = \frac{1}{n * n} + \sum_{j=-1}^1 \sum_{i=-1}^1 I(u + i, v + j)$$

Filtre de gaussien : Le filtre gaussien est un filtre isotrope, ce qui signifie que les propriétés de son noyau de convolution sont identiques dans toutes les directions. Ce filtre présente des propriétés mathématiques bien définies. L'effet de ce filtre sur l'image est assez similaire à celui du filtre moyen, mais la moyenne est pondérée dans le sens où les pixels proches du centre ont un effet ou un "poids" plus important que ceux qui sont situés plus loin. Le lissage à l'aide d'un filtre gaussien est justifié lorsqu'aucune information a priori sur les interactions entre les pixels de l'image n'est disponible. On formule alors l'hypothèse d'une loi d'interaction moyenne, autrement dit une distribution gaussienne.

En général, un filtre gaussien avec un sigma inférieur à 1 est utilisé pour réduire le bruit. En revanche, un sigma supérieur à 1 est souvent utilisé pour créer une image floue destinée à fabriquer un "masque flou" personnalisé, une technique fréquemment appliquée aux images satellites. Il est important de noter que plus la valeur de sigma est grande, plus la cloche gaussienne est large et plus l'effet de flou appliqué à l'image est prononcé.

Filtre de laplacien : est un filtre passe-haut dédié à la détection de contours. Sa particularité est de détecter des lignes horizontales, verticales et diagonales. Comme il agit dans le domaine du filtrage linéaire, il possède un masque de convolution qui s'exprime sous deux connexités 4 ou 8. Ce filtre permet de réaliser la détection de contours en une seule passe. D'où sa très grande utilisation dans le domaine de la détection de contours pour sa souplesse d'application pratique.

Les filtres de détection de contours transforment l'image d'entrée en une image noire sauf aux points où un contour est détecté qui est marqué en blanc. En procédant à une inversion de contraste, on obtient des contours noirs sur fonds blancs. La mise en évidence des contours constitue la différenciation de l'image. Un contour sera bien détecté s'il constitue une frontière entre deux régions homogènes. Le contour est un concept essentiellement basé sur la capacité du système visuel humain à distinguer plusieurs zones connexes d'aspect homogène, pour lesquelles la ligne de séparation sera appelée un contour ou une frontière. Il n'y a donc pas de définition précise et on peut dire qu'il y a des contours là où le système visuel humain en voit.

5- Filtre de sobel : ou opérateur de Sobel-Feldman, est une technique utilisée pour détecter les contours en identifiant les changements brusques d'intensité lumineuse dans une image bitmap. Il calcule le gradient d'intensité à chaque pixel pour déterminer à la fois la direction et l'intensité des variations lumineuses, mettant ainsi en évidence les contours. Ce gradient est obtenu en appliquant des matrices de convolution 3×3 :

dérivée horizontale (Gx)

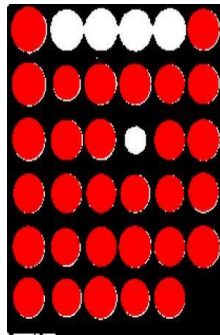
-1	0	1
-2	0	2
-1	0	1

dérivée verticale (Gy)

1	-2	-1
0	0	0
1	2	1

Ces dérivées sont combinées pour calculer la norme du gradient, $(G = \sqrt{Gx^2 + Gy^2})$ qui représente l'intensité du contour, et la direction du gradient $\Theta = \text{atan2}(Gy, Gx)$, qui indique l'orientation du contour. Le filtre de Sobel est largement utilisé pour extraire les bords dans les images, facilitant ainsi leur analyse dans des domaines comme la reconnaissance de formes.

7-Visualisation des résultats :

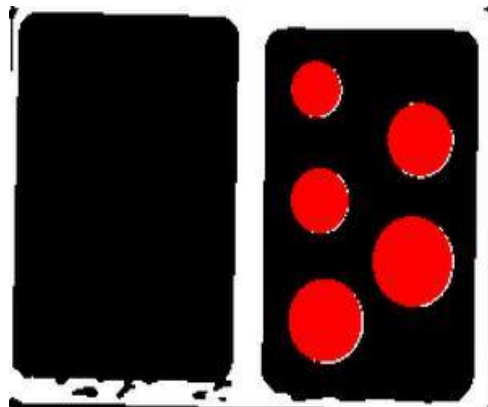


Précision: 85.71%

Détecte: 30 | Réel: 35

Précision: 76.19%

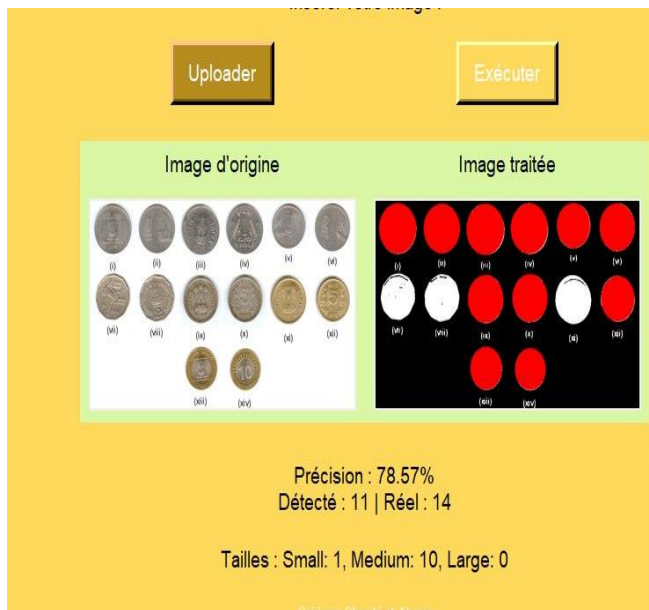
Détecte: 21 | Réel: 16



Précision: 100%

Détecte 6 | Réel: 6

8-Visualisation de l'interface :



9-Conclusion:

En conclusion, le système développé dans le cadre de ce projet a démontré son efficacité en détectant avec succès les pièces de monnaie présentes dans toutes les images testées, tout en assurant une correspondance fidèle entre le nombre de pièces détectées et le nombre réel. Grâce à l'intégration de techniques avancées de vision par ordinateur, telles que la segmentation et la classification.

10-Bibliographie :

- <https://dl.ebooksworld.ir/motoman/Digital.Image.Processing.3rd.Edition.www.EBooksWorld.ir.pdf>
- <https://fac.umc.edu.dz/fstech/cours/Electronique/Master%20ST%C3%A9%A9com/CoursImageProcessing1.pdf>
- <https://fr.scribd.com/document/511219104/Traitement-d-Image>
- Feature Extraction in Computer Vision and Image Processing - Mark S. Nixon A member of the Reed Elsevier plc group First edition 2002 © Mark S. Nixon and Alberto S. Aguado 2002
- https://www.google.dz/books/edition/Traitement_des_images_avec_C_5_et_WPF/E1g9AwAAQBAJ?hl=fr&gbpv=1
- <https://hal.science/hal00706168/file/Analyse%20d'images-Filtrage-et-segmentation-JPC-SF.pdf>
- [Algorithms and ApplicationsSzeliskiBook 20100805 draft.](#)