

Compte Rendu de TP2 — Transformé de Hough

KALALA KABAMBI David

MARTY Julien

Exercice 1)

1)

R [1 ; 100]

Pour un pas de 2, nombre de valeurs possibles de $r = ((100 - 1) / 2) + 1 = 50$.

Pour un pas de 0,5, nombre de valeurs possible de $r = ((100 - 1) / 0,5) + 1 = 199$.

2)

R [1 ; 100], delta de 1, $((100 - 1) / 1) + 1 = 100$ valeurs possibles

C [1 ; 100], delta de 1, $((100 - 1) / 1) + 1 = 100$ valeurs possibles

Rad [5, $100 * \sqrt{2}$] \Leftrightarrow [5, 141], delta de 1, $((141 - 5) / 1) + 1 = 137$ valeurs possibles

On a 100 valeurs possibles pour r, 100 valeurs possibles pour c et 137 valeurs possible pour rad.

Ainsi : $100 * 100 * 137 = 1\,370\,000$

On peut décrire un total de 1 370 000 cercles avec ces valeurs.

3)

Calcul = (indice - 1) * pas + val_min

- acc(1, 1, 1)

$r = (1 - 1) * 1 + 1 = 1$

$c = (1 - 1) * 1 + 1 = 1$

$rad = (1 - 1) * 1 + 5 = 5$

acc(1, 1, 1) décrit le cercle de centre (1, 1) et de rayon 5.

- acc(10, 7, 30)

$r = (10 - 1) * 1 + 1 = 10$

$c = (6 - 1) * 1 + 1 = 7$

$rad = (30 - 1) * 1 + 5 = 34$

acc(10, 7, 30) décrit le cercle de centre (10, 7) et de rayon 34.

4)

- Cercle de centre (40, 40) et de rayon 13 :

$r = ((40 - 1) / 1) + 1 = 40$

$$c = ((40 - 1) / 1) + 1 = 40$$

$$rad = ((13 - 5) / 1) + 1 = 9$$

La case de l'accumulateur associé est (40, 40, 9)

Exercice 3)

1)

La complexité de l'algorithme est de l'ordre N^4 car pour remplir l'accumulateur, on parcourt une fois chaque pixel, on a donc une complexité d'ordre N^2 . Puis pour obtenir le minimum local, on parcourt de nouveau une fois chaque pixel (moins les quelques pixels en bordures) ce qui nous donne encore une fois une complexité d'ordre N^2 . On a alors au total une complexité d'ordre N^4 .

S'il faut un temps T pour effectuer la détection sur une image de 100 pixels, pour une image 6 fois plus grande, il faudra 6^4 fois plus de temps, soit $1296 \cdot T$.

Problèmes rencontrés :

1. Paramétrage de la méthode de Hough

Nous n'avons pas réussi à trouver de paramètres optimaux pour toutes les images. En fonction de l'image sur laquelle nous effectuons la détection de cercle, il faut énormément ajuster les paramètres de l'algorithme pour obtenir une détection optimale. D'après notre analyse ceci est dû à :

- Caractéristiques de l'image : luminosité, contraste.
- Dimensions, échelles et nombre de cercles.

2. Encadrement du rayon pour la mise à jour des cercles de la méthode pyramidale.

Lors de l'utilisation de la méthode pyramidale, nous mettons à jour les centres des cercles et leurs rayons. Lors de la mise à l'échelle à la résolution plus grande, il arrive que les cercles détectés soient significativement plus petits ou plus grands que les cercles présents sur l'image (avant la mise à jour). Cette imprécision est due à plusieurs facteurs.

- Des détections sur des images très petites.
- Des mises à l'échelle imprécises à cause de la conversion des nombres flottants. Pour certaines mises à l'échelle nous avons observé des écarts de rayon allant jusqu'à 6 pixels, ce qui est particulièrement visible sur des images de petites tailles.

Pour compenser ces écarts, il est nécessaire de définir un encadrement des rayons suffisamment large afin de détecter les cercles dont le rayon s'écarte de l'estimation initiale. Cependant sur les images, les traits qui définissent les cercles sont parfois très épais. Il faut donc choisir un encadrement suffisamment grand pour inclure complètement la largeur du trait du périmètre du cercle et ainsi déterminer en moyenne le rayon passant au milieu de cette large bande.

3. Instabilités des cercles dans la méthode pyramidale.

Comme vu précédemment, lors de l'utilisation de la méthode pyramidale pour ajuster les coordonnées et les rayons des cercles, il est nécessaire d'utiliser un encadrement des rayons suffisamment large. Cependant, cette approche peut entraîner des changements significatifs dans les cercles détectés d'une itération à l'autre. En effet, un encadrement plus large augmente le nombre de cases utilisées dans l'accumulateur et ainsi augmente les possibilités de mise à jour. Par conséquent, lors de la recherche de maximum locaux, la taille du kernel, pour filtrer les résultats, est trop petite pour distinguer deux cercles concentriques (les centres des cercles ne sont pas nécessairement identiques). L'intervalle de mise à jour du rayon des cercles peut aller jusqu'à 20 pixels. Un kernel plus grand pourrait permettre de mieux traiter les cercles concentriques mais aurait un impact négatif sur la détection des autres cercles. D'autre part, des images comme « moon » possèdent des cercles concentriques qu'on souhaite détecter, Il alors est difficile de faire la différence entre de nouveaux cercles concentrique mal placé et l'amélioration pertinente des coordonnées des cercles précédemment détecté.

On peut alors se demander pourquoi des cercles « mal placé » ont une valeur plus forte dans l'accumulateur que d'autres cercles auparavant bien placés ? Les images de contour sont différentes à chaque itération puisque leur échelle change. Cette différence peut expliquer les différences de valeur dans l'accumulateur ainsi que leur hiérarchie. Il peut y avoir des petits changements de valeurs qui modifie néanmoins la hiérarchie des cercles.

En décrivant ce problème dans le rapport, nous avons eu l'idée suivante pour le contrer :

- Dans la méthode de mise à jour des coordonnées et rayon des cercles, au lieu de sélectionner les meilleurs cercles parmi tous les cercles calculés, on choisit le meilleur cercle pour chaque cercle mis à jour.

De cette manière, les cercles correctement positionnés n'ont plus tendance à disparaître. Cependant, lors de la mise à l'échelle, nous avons toujours des cercles dont le rayon ne s'ajuste pas correctement à cause du problème énoncé précédemment.

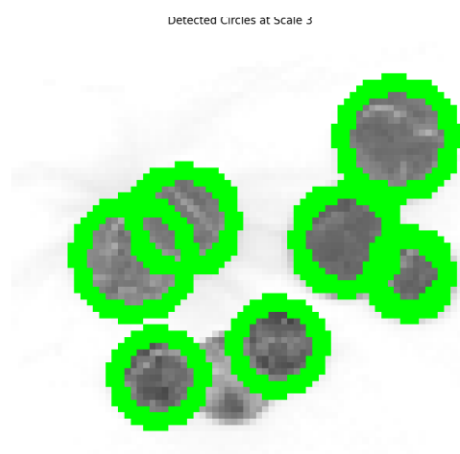


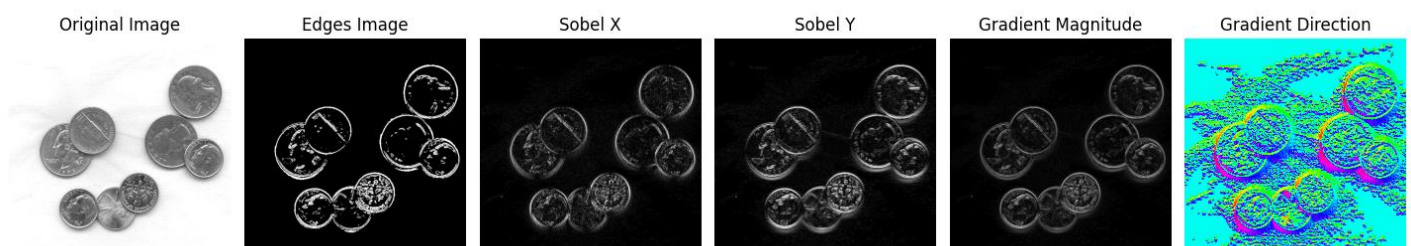
Figure 1 Image réduite 3 fois avec un facteur 2



Figure 2 Image avec la taille initiale

En voyant ces deux images, on peut noter deux choses :

- Il y a une pièce qu'on n'arrive pas à détecter.
- Une pièce est remplie d'une foule de détails considéré comme des contours et fausse les résultats lors de la mise à jour des cercles.



On distingue clairement que l'intérieur de la possède un grand nombre de contours qui fausses nos résultats. Puisque les contours détectés non désirables ne sont pas du bruit mais bel et bien des contours, le filtra gaussien que nous appliquons est inutile.



Figure 3 Meilleur détection sur l'image la plus complexe

Après avoir beaucoup modifié les paramètres de l'algorithme nous avons réussi à obtenir la détection de cercle ci-dessus. Les paramètres que nous avons modifiés et qui le plus significativement améliorer notre détection sont :

- Le rayon minimum des cercles détectés : 7 pixels
- L'intervalle de mise à jour des rayons des cercles qui est de +/- 17 pixels par rapport au rayon initiale du cercle.

Conclusion

Ce projet nous a permis d'explorer les défis et subtilités liés à la détection de cercles dans des images à l'aide de la méthode de Hough. Plusieurs limitations et pistes d'amélioration ont été mises en évidence :

- Paramétrage sensible : L'ajustement des paramètres de la méthode de Hough est essentiel pour obtenir des résultats satisfaisants. Les variations dans les caractéristiques des images (luminosité, contraste, échelle des cercles) nécessitent une calibration manuelle importante, ce qui limite la généralisation de la méthode.
- Problèmes liés à la méthode pyramidale :
 - Les erreurs dans la mise à jour des rayons et des centres des cercles, dues à des imprécisions d'échelle et à la largeur des traits dans les images, ont parfois compromis la précision des résultats.
 - Les instabilités observées dans la détection des cercles, notamment pour des images contenant des cercles concentriques ou des contours complexes, montrent la nécessité d'améliorer le filtrage et l'analyse des résultats dans l'accumulateur.
- Gestion des contours indésirables : Certaines images ou parties de l'image contiennent des détails internes qui génèrent des contours non pertinents pour la détection de cercles. Ces contours perturbent l'accumulateur et montrent les limites des méthodes classiques comme le filtre gaussien.