

Rendu TP 2

Traitement d'images

BOUTIN André
DE ROECK Dimitri

Novembre 2023

Exercice 1

1. On peut appliquer : $\frac{\text{Borne supérieure} - \text{Borne inférieure}}{2+1} = 50$ valeurs possibles pour r . Pour un pas de 0,5, nous aurons $\frac{100-1}{0.5+1} = 199$ valeurs possibles.
2. En suivant le même raisonnement :
 - r : 100 valeurs possibles
 - c : 100 valeurs possibles
 - rad : $100 * \sqrt{2} - 5/1 + 1 \approx 137$ valeurs possiblesNous aurons donc $100 \times 100 \times 137 = 1370000$ cercles possibles.
3. En se basant sur les intervalles de la question précédente, le cercle associé au $acc(1, 1, 1)$ est le cercle de centre $(1, 1)$ car la première valeur pour r et c est 1 et de rayon 5 pixels car la première valeur de l'intervalle rad est 5. Quant au cercle associé au $acc(10, 7, 30)$:
 - 10^e valeur discrète pour $r \rightarrow 10$
 - 7^e valeur discrète pour $c \rightarrow 7$
 - 30^e valeur discrète pour rad sachant que l'intervalle commence à 5 $\rightarrow 34$Cela nous donne un cercle de centre $(10, 7)$ et de rayon 34 pixels.
4. En inversant le procédé, on tombe donc sur la valeur la valeur suivante : $acc(40, 40, 9)$.

Exercice 2

Lors de la réalisation de cet exercice, nous avons rencontré plusieurs problèmes :

- Un des problèmes que nous avons rencontré et qui avait été mentionné en cours était celui de la difficulté à trouver les petits cercles à cause du vote qui favorisait les grands cercles. Afin de pallier ce manque d'égalité pour les petits cercles de notre algorithme, il nous a fallu normaliser les votes pour pas que la quantité de pixels d'un cercle soit un facteur déterminant dans le vote. Ainsi, pour chaque vote, celui ci était divisé par le rayon du cercle. Cela s'est avéré efficace puisque nous avons pu trouver les cercles plus petits des images (notamment `coins.png`).
- - Nous avons eu des difficultés à trouver une solution générique pour chaque image. En particulier concernant l'image `coins2.jpg` qui contient beaucoup de bruits et sur laquelle nous n'avons pas trop réussi à identifier les cercles de l'image en faisant varier les paramètres. On a réussi à identifier 3 cercles sur cette image avec : `blur_kernel = (7, 7)` et `t = 0,6`.



Outre cette image, nous avons réussi à trouver les cercles sur les autres images en faisant varier les paramètres dont notamment un nettoyage plus grand autour des top valeurs de l'accumulateur afin de pallier le manque de précision de notre algorithme. Bien qu'il soit imprécis, cela nous a permis d'identifier à quelques pixels près tous les cercles des autres images autre que `coins2.jpg`. (paramètres : seuil t de 0.3, N en fonction du nombre de cercle à détecter).

Nous avons cependant essayé de tester un algorithme inspiré d'une vidéo utilisant les coordonnées polaires cependant cela n'a pas donné énormément de résultat supplémentaire. Enfin, en jouant notamment avec le filtre gaussien pour essayer d'atténuer le bruit de l'image, nous avons réussi à détecter 3 à 4 cercles, cependant, cela affectait par conséquent notre détection sur les autres images.

De plus, la manière dont on itère et nettoie les 26 valeurs autour d'une centre du cercle détecté n'est pas du tout optimisé.

Il est probable que mettre un seuil plus bas 0.2-3 pour ' t ' (: notre seuil à partir duquel on choisi un contour) sur `coin2` avec un kernel plus large pour mieux lisser l'image nous permettra de gagner en précis pour la détection des cercles puisque les contours seront mieux détecter mais notre temps d'exécution sera extrêmement long.

Exercice 3

La complexité de l'algorithme est de l'ordre de $O(n^4)$ car nous effectuons un parcours des contours et de tous les pixels de l'image (double boucle) afin de pouvoir faire les votes. Nous itérons ensuite une dernière fois sur l'accumulateur afin de trouver les maximas locaux.