Master 2 Informatique | CIM

Programmation créative - 2018-2019

TD 2

Processing TD2

Rappels

Quelques raccourcis intéressants :

Ctrl + **R** : Exécuter le programme
Ctrl + **Maj** + **T** : "Tweak" : permet de modifier la valeur des variables et voir le résultat en direct
Ctrl + **Espace** : complétion intelligente
Ctrl + **T** : indentation automatique du code

Documentation : `https://processing.org/reference`

1 Analyse d'images : opérations locales

Exercice 1 : Filtres et débruitage

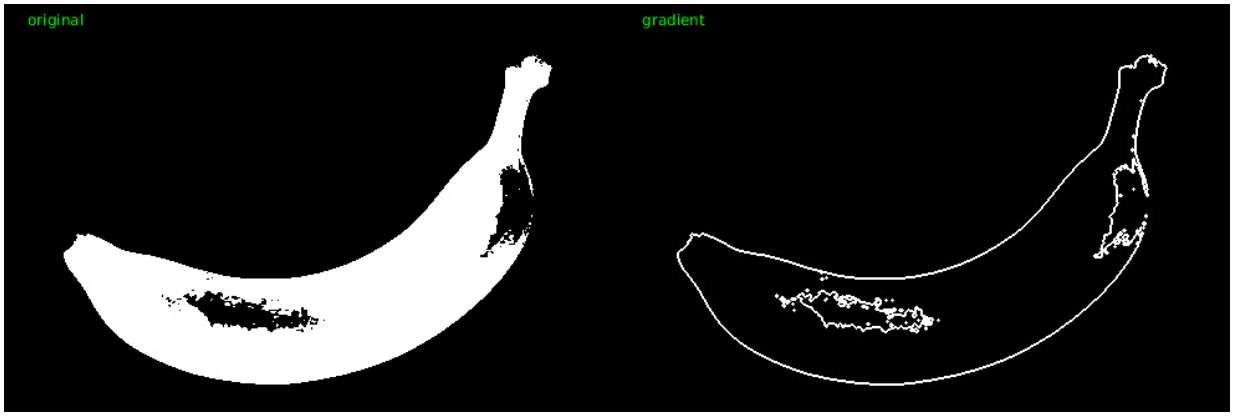
Nous nous intéressons au débruitage des images. Les filtres permettent d'effectuer des traitements locaux sur l'image, particulièrement adaptés à ce type de problème.

Lisez attentivement le fichier **filtres.pde**. Tous les filtres vus en cours ont été implémentés (median, moyenneur et gaussien). L'image est entachée de bruit poivre et (surtout) sel.

1. Exécutez le programme. L'image générée par défaut est le résultat de l'application d'un filtre moyenneur de taille 3x3.
2. Dans la fonction `draw()`, modifiez le type de filtre. Essayez successivement les filtres `median` et `gaussian`. De même, essayez différentes tailles pour chaque filtre.
3. Quel filtre semble le plus adapté pour corriger un bruit de type poivre et sel ?
4. Chargez l'image `lenagauss.jpg`, puis appliquez et comparez les différents filtres sur cette image. Cette image contient du bruit gaussien (intensité aléatoire suivant une loi de distribution gaussienne).
5. Quel filtre semble le plus adapté pour corriger un bruit de type gaussien ?

Exercice 2 : Extraction de contours

L'objectif de cet exercice est de manipuler les opérateurs de morphologie mathématique (érosion et dilatation) afin de calculer le gradient (contours) d'une image binaire (cf. image de gauche).



Lisez attentivement le fichier **contours.pde**. A l'exécution, l'image originale est à droite, et son image érodée est à gauche.

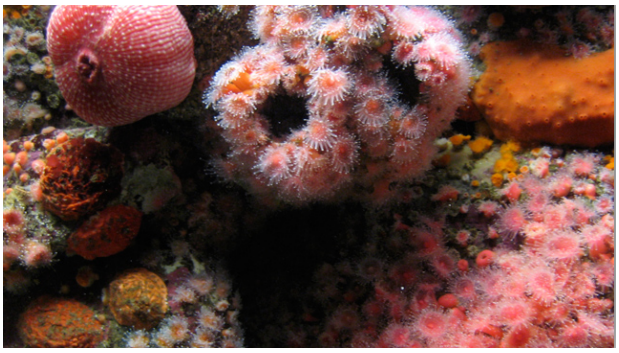
La fonction `filter()` de Processing permet de calculer l'érosion et la dilatation. La taille de la région du filtre est de 3x3 par défaut.

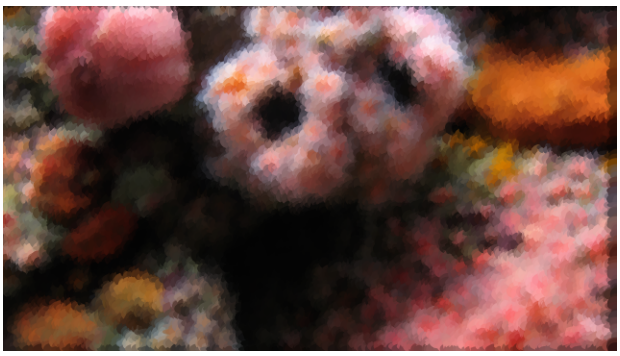
Questions :

1. D'après la documentation, est-il possible de modifier la taille du filtre (5x5, 7x7) ? Comment effectuer une érosion avec une région de taille plus importante ?
2. Sur le même principe que le calcul de l'image érodée, créez une nouvelle image `PImage dilated`, dans laquelle vous stockerez le résultat de la dilatation. Affichez ensuite ce résultat à la place de l'image érodée.
3. Créez une nouvelle image `PImage gradient`, dans laquelle vous calculerez l'image **gradient** de l'image originale. On rappelle que l'image gradient est obtenue par une différence pixel à pixel entre l'image dilatée et l'image érodée.
4. Quel est le problème avec les contours obtenus ?

Exercice 3 : Pointillisme

L'objectif de cet exercice est de créer un effet pointilliste sur une image quelconque. L'idée est d'ajouter des cercles dans l'image dont la couleur correspond à la **couleur moyenne des pixels dans ce cercle** (cf. figure de droite).





Nous rappelons ici le pseudo-algorithme pour parvenir à un tel effet :

1. $t =$ taille de la région
2. Pour chaque pixel p de l'image :
   (a) $m_c =$ moyenne des intensités dans la région de taille t
   (b) Créer un cercle de centre p et de rayon t dont la couleur de remplissage est m_c

Lisez attentivement et exécutez le fichier **pointillism.pde**. Une fenêtre noire apparaît. L'image est déjà chargée dans le programme, mais n'est pas affichée.

Modifiez uniquement la fonction `couleurMoyenne(x,y,img)` qui renvoie la couleur moyenne dans la région de taille `rayonRegion` centrée sur le pixel (x, y) :

1. La moyenne des couleurs doit se faire indépendamment pour chaque canal (R, G, B). Déclarez trois variables qui contiendront la moyenne pour chaque canal.
2. Parcourez l'ensemble des pixels dans la région de taille `rayonRegion`, et faites la somme des intensités dans cette région, indépendamment pour chaque canal à l'aide des fonctions `red()`, `green()`, et `blue()`.
3. Calculez la moyenne en divisant cette somme par le nombre de pixels dans la région.
4. Retournez la couleur contenant la moyenne pour les trois canaux.

L'image avec l'effet devrait s'afficher. Faites varier les valeurs de rayon à l'aide de la fonctionnalité **Tweak** de Processing.