

# Traitement d'images

## TP 3

Garcia Maxime, Palisse Benjamin, Petitbon Pierre

L'objectif de ce TP est de comparer deux méthodes de détection de contour d'une image, l'une par gradient l'autre par laplacien. Nous testerons les deux programmes sur des versions plus ou moins bruitées d'une image dont les contours intérieurs et extérieurs sont représentés sur les figures suivantes :

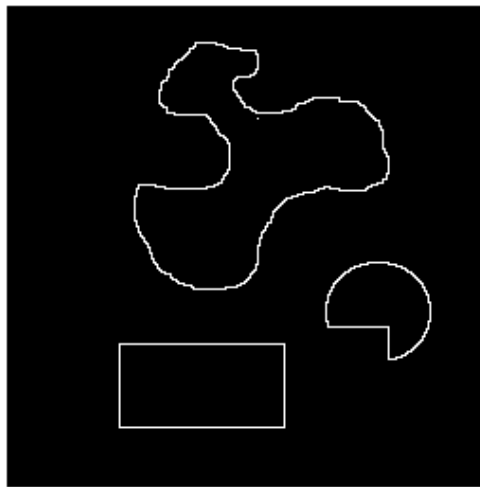


FIGURE 1 – contour externe

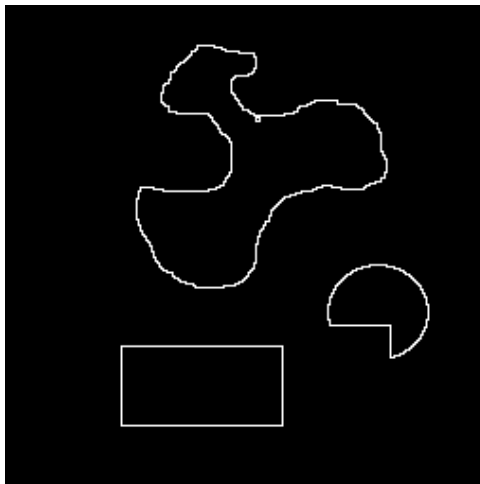


FIGURE 2 – contour interne

Nous testerons aussi nos programmes sur les images *tangram.pdf* et *lacornou.pdf*, cette dernière étant bien plus compliqué.

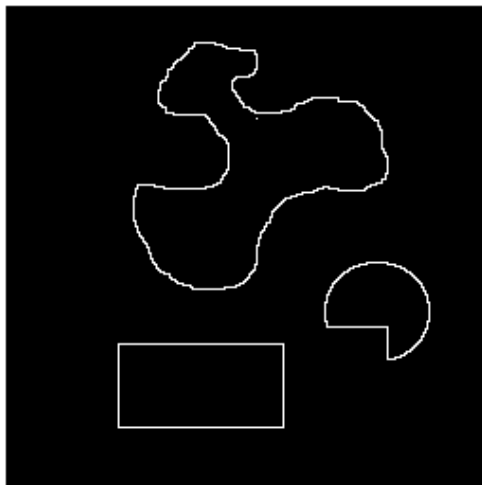


FIGURE 3 – tangram.pdf

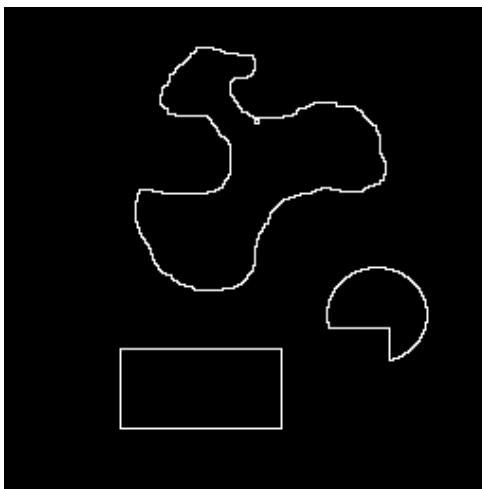


FIGURE 4 – lacornou.pdf

Pour compiler nos fichiers sources avec le makefile, il suffit de taper : `-make gradient`  
`-make gradNaif` `-make laplacien` `-make laplacienDoG` `-make laplacienCouronne`

Pour exécuter nos programmes sur les images proposées, il suffit de placer les images dans un répertoire parent appeler `img` puis de lancer les commandes `-make execGradPrewitt` `-make execGradSobel` `-make execGradNaif` `-make execLaplacienDoG` `-make execLaplacien1` `-make execLaplacien2` `-make execLaplacien3` `-make execLaplacienCouronne`

# 1 Détection par gradient

## 1.1 gradientNaif

Cette détection de contour est obtenu en calculant le gradient de notre image, puis en considérant que les contours sont les points où le gradient est le plus important sans utiliser la suppression des non-maxima. En appliquant cette méthode sur l'image bruitée suivante (image simulant un bruit multiplicatif de type speckle), on obtient ceci :

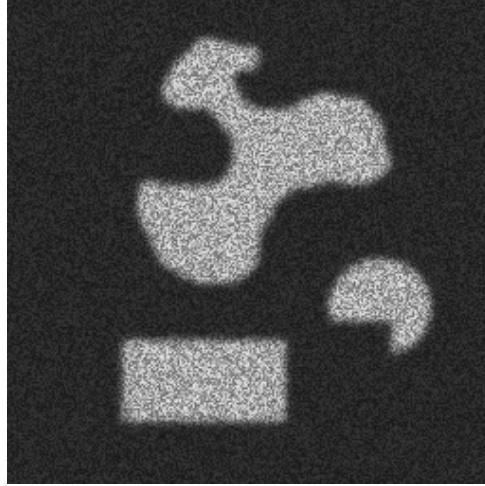


FIGURE 5 – image ayant un bruit multiplicatif de type speckle



FIGURE 6 – image ayant un bruit multiplicatif de type speckle

On constate qu'en identifiant les points ayant un gradient supérieur à 250 comme point de contour, le programme identifie bien la forme principale des différents objets de l'image mais il interprète l'intérieur des objets aussi comme des contours. En augmentant la valeur du seuil à 680, on constate que le seuil est trop grand car même les points qui

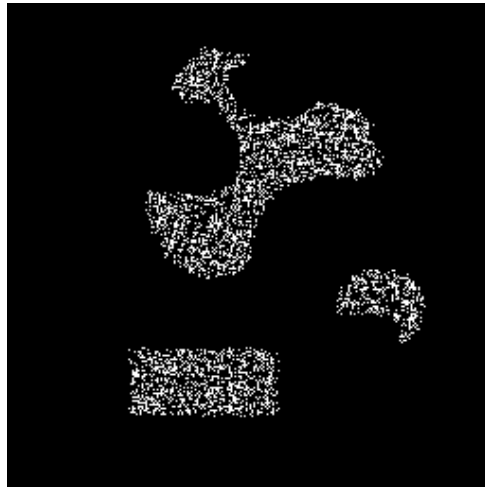


FIGURE 7 – image ayant un bruit multiplicatif de type speckle

sont réellement des contours tendent à disparaître. En augmentant encore le seuil, c'est la totalité de la forme qui disparaît. Cette méthode n'est donc pas bonne car le programme n'est pas capable de distinguer le contour de l'intérieur de l'objet.

## 1.2 gradients Prewitt et Sobel

Cette fois-ci, on supprime les non maxima et on fait un seuillage par hystérésis en utilisant les gradient de Prewitt ou de Sobel. On constate que les contours de l'objet sont reconnus, mais que même l'intérieur de l'objet est cependant considéré comme un contour.

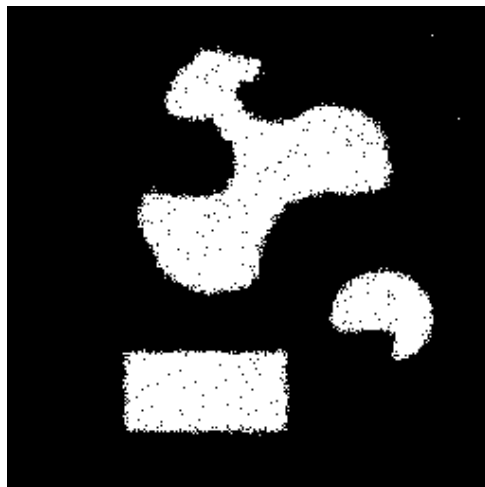


FIGURE 8 – seuilMax = 500 et seuilMin = 20

Pour l'image tangram, les formes sont reconnus et en parties colorié, mais on remarque cependant quelque chose de positif : l'intérieur du triangle. A l'endroit où il y a un léger renforcement dans l'image initiale, l'image obtenu change de couleur. Cela signifie que

notre programme reconnaît un contour délimitant une autre zone à l'intérieur d'un contour malgré le fait qu'il affiche l'intérieur des surfaces comme leurs contours.

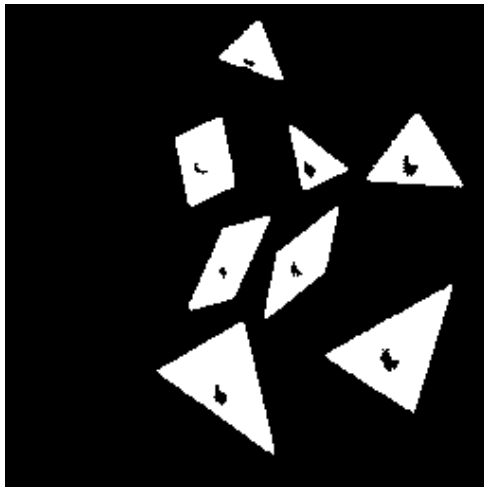


FIGURE 9 –  $\text{seuilMax} = 400$  et  $\text{seuilMin} = 20$

Pour une image plus complexe comme la cornou, avec 500 et 20 pour seuil, les contours sont bien soulignés. On pourrait utiliser cette méthode pour réhausser les contours dans une image.



FIGURE 10 –  $\text{seuilMax} = 500$  et  $\text{seuilMin} = 20$

## 2 Détection par Laplacien

### 2.1 Lissage

Nous avons convolué l'image initiale avec les 3 matrices de laplacien proposées dans l'énoncé. Les images obtenus ne permettent pas de reconnaître les formes initiales.

## 2.2 difference of Gaussian

Dans cette partie, nous calculons la transformée de Fourier de notre image, nous la multiplions avec la transformée de Fourier de la différence de deux filtres Gaussiens, puis nous cherchons les passages par zéro du laplacien obtenue.

On utilise la détection de contour sur l'image *formes1bb25.pdf*. Si l'on prend  $\sigma_1 = 10$  et  $\sigma_2 = 6.25$ (le rapport des 2 faisant bien 1.6), on obtient

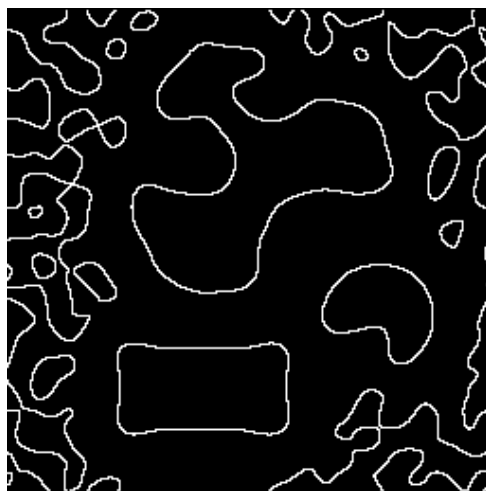


FIGURE 11 – laplacien avec 10 et 6.25 pour valeur de filtre

Les formes principales des contours ressortent, cependant le programme rajoute de nombreux contours "parasites" autour des formes à cause du bruit. Les contours reconnus par le programme sont plus arrondis que ne le sont les originaux réellement.

Avec comme paramètre des filtres lors de la différence de gaussienne  $\sigma_1 = 30$  et  $\sigma_2 = 18.75$ , on obtient les contours suivants :

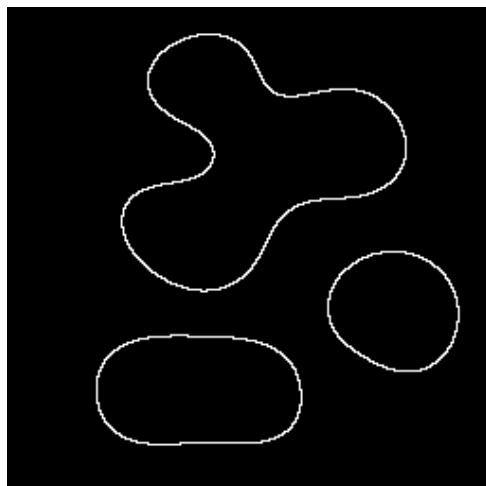


FIGURE 12 – laplacien avec 30 et 18.75 pour valeur de filtre

Les contours sont très apparents mais plus arrondis que l'image originale, cela est notamment très visible sur le rectangle en bas. Les traits un peu "tremblants" de la figure au milieu en haut dans l'image initiale apparaissent ici très courbés. La figure de droite est devenu un rond. Par rapport au cas précédent, les contours sont plus arrondis, et donc moins fidèle à la réalité, mais les contours superflus des bords ont disparu.

Pour l'image tangram, avec  $\sigma_1 = 5$  et  $\sigma_2 = 3.125$  :

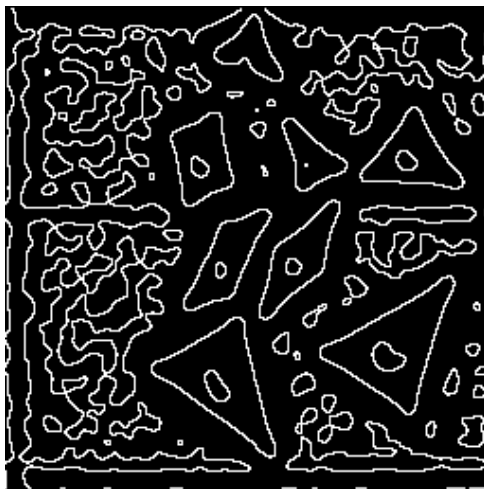


FIGURE 13 – laplacien avec 5 et 3.125 pour valeur de filtre

Les triangles initiaux sont reconnus par le programme, ainsi que le petit effet de relief qu'il y a en chacun d'eux. Cependant, de nombreux contours d'objets qui n'étaient pas dans l'image initiale apparaissent, comme dans les cas précédents. Le côté gauche à l'origine vide est alors remplie par des objets.

Si l'on augmente les valeurs des filtres,  $\sigma_1 = 12$  et  $\sigma_2 = 7.5$ , on obtient :

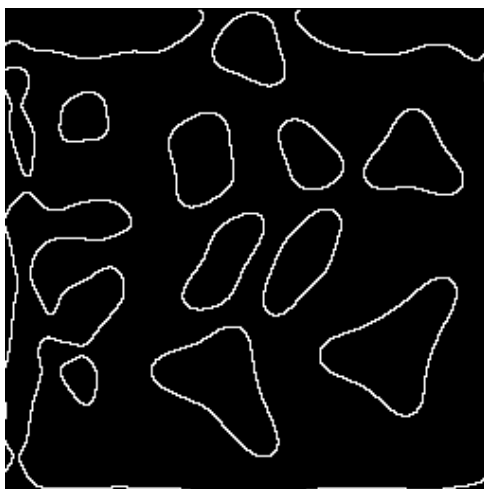


FIGURE 14 – laplacien avec 12 et 7.5 pour valeur de filtre

Les triangles de la figure initiale apparaissent aussi dans le résultat produit par notre algorithme. Cependant, ils prennent une forme plus ronde sur les bords, et l'effet de relief disparaît. Les contours apparaissant sur les bords du aux bruits sont moins nombreux.

Pour l'image lacornou, en prenant  $\sigma_1 = 8$  et  $\sigma_2 = 5$

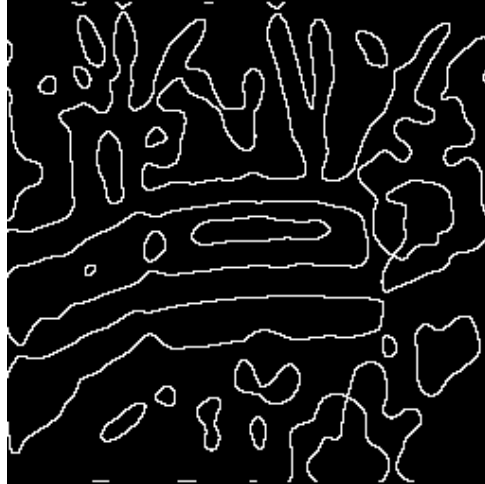


FIGURE 15 – laplacien avec 8 et 5 pour valeur de filtre

On constate que pour une image avec ce niveau de détail, notre programme ne fournit pas des contours satisfaisant.

## 2.3 Couronne

Notre couronne est composée de deux carrés concentriques. Nous calculons la transformée de Fourier de notre image, puis nous mettons à 0 les valeurs ne se situant entre les deux couronnes.

En appliquant notre filtre a l'image *bb5* avec pour valeur  $R2 = 20$  et  $R1 = 10$ , nous obtenons l'image

Pour l'image tangram, nous reconnaissons bien les triangles sur la partie avec leurs renforcements en leurs centres.

Sur l'ensemble des figures testés, le rapport  $R2 = 2R1$  semble fonctionner le mieux.



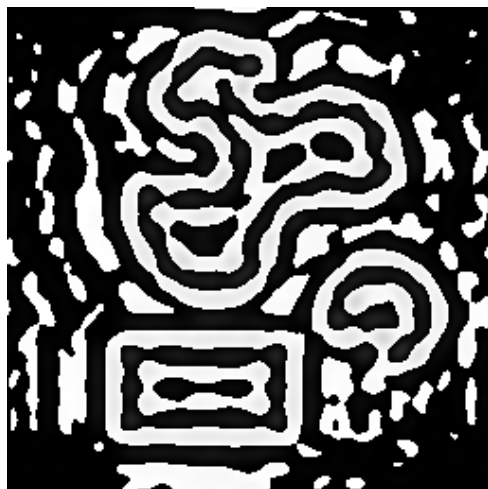


FIGURE 16 – laplacien avec R2 de 20 et R1 de 10

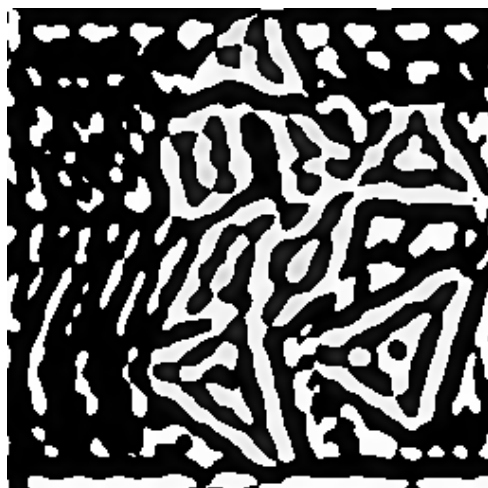


FIGURE 17 – laplacien avec R2 de 20 et R1 de 10