



DIRO  
IFT 6150

## TRAVAIL PRATIQUE N° 2

### DÉTECTION DES CONTOURS D'UNE IMAGE FILTRE DE CANNY

*Max Mignotte*

DIRO, Département d'Informatique et de Recherche Opérationnelle.

[http : //www.iro.umontreal.ca/~mignotte/ift6150](http://www.iro.umontreal.ca/~mignotte/ift6150)

*e-mail : mignotte@iro.umontreal.ca*

#### FILTRE DE CANNY

Le filtre de Canny est d'une bonne efficacité en traitement d'image pour détecter et segmenter les contours d'une image. Cet algorithme est constitué de 3 étapes distinctes. Ces étapes sont les suivantes.

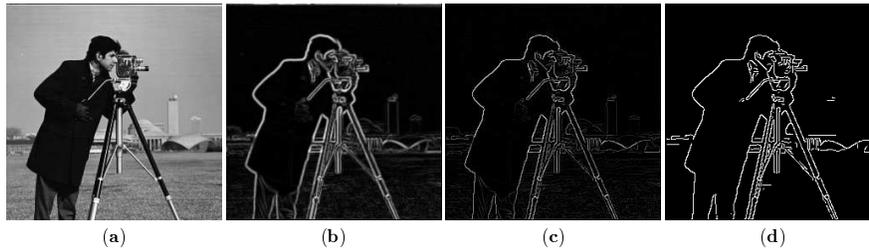


FIG. 1 – (a) Image originale `photograph.pgm`. (b) Rehaussement de Canny (avec un filtre Gaussien d'écart type  $\sigma = 1$ ). (c) Résultat de la suppression des non maximums (étape 2). (d) Résultat du seuillage par hystérésis avec  $\tau_l = 33$  et  $\tau_h = 66$  (étape 3).

#### 1. REHAUSSEMENT DE CANNY

- Appliquer un filtre gaussien sur l'image.
- Calculer pour chaque pixel  $(i, j)$  de l'image, la norme du gradient  $(e_s(i, j))$  et l'angle de la normale au gradient (i.e., l'angle de la tangente à la direction du contour),  $(e_o(i, j))$ ,

$$e_s(i, j) = \sqrt{J_x^2(i, j) + J_y^2(i, j)},$$
$$e_o(i, j) = \arctan \frac{-J_x}{J_y},$$

où  $J_x$  et  $J_y$  représente le gradient en  $x$  et en  $y$ .

- Approximer l'angle de la normale au gradient (i.e., la tangente à la direction du contour),  $(e_o(i, j))$  en chaque point par l'une des 4 directions suivantes,  $0^\circ, 45^\circ, 90^\circ, 135^\circ$ .

## 2. SUPPRESSION DES NON MAXIMUMS

i.e., pour tous les pixels  $(i, j)$ ,

- Si  $e_s(i, j)$  est plus petit qu'au moins un de ses deux voisins le long du gradient, alors supprimer le contour (i.e.,  $e'_s(i, j) = 0$ ), sinon conserver-le (i.e.,  $e'_s(i, j) = e_s(i, j)$ ).

## 3. SEUILLAGE PAR HYSTÉRESIS

i.e., pour tous les pixels  $(i, j)$  pour lesquels  $e'_s(i, j) > \tau_h$ ,

- Considérer comme contour, tous les pixels pour lesquels  $e'_s(i, j) > \tau_l$  et qui sont connectés au point de contour précédent perpendiculairement à son gradient (i.e., à la tangente à la direction du contour).

(1) Faire une détection des contours de l'image **photograph.pgm** par l'algorithme de Canny.

(2) On se propose maintenant de programmer une méthode permettant de rendre l'algorithme de Canny semi-automatique. Plus précisément, la méthode dont il est question permet de n'être dépendant que d'un seul seuil et non plus des deux seuils généralement utilisés dans la méthode de Canny, (i.e.,  $\tau_l$  et  $\tau_h$ ).

Pour cela, on fixe  $\tau_l = 0.5\tau_h$  et  $\tau_h$  est estimé automatiquement comme étant la valeur de module de gradient pour laquelle la fonction de répartition du module du gradient (calculé sur toute l'image) atteint la valeur  $p_h$  ( $p_h < 1$ ). Implémenter cette méthode et tester pour des valeurs de  $p_h$  variant entre 0.70 et 0.95.

## REMISE

### A. DATE DE REMISE ET CONDITIONS

Vous devez rendre physiquement au démonstrateur électroniquement le(s) programme(s) fait en C (voir le barème) avant la date de remise spécifiée dans le fichier *barème* situé sur la page web du cours dans la rubrique *Introduction & Program*. Pour la remise électronique, utilisez le programme *remise* (*man remise* pour plus de détails) pour remettre votre code dans le répertoire TP<Numéro du Tp>. N'oubliez pas d'inscrire vos noms, courrier électronique en commentaire en haut du fichier *.c* remis. Les noms des programmes à remettre devront avoir le format suivant *TpIFT6150- <Numéro du Tp>-<Numéro de la question>.c*. Les programmes devront se compiler et s'exécuter sur Linux tel qu'indiqué dans le fichier barème.