TIFO

TP2: Filtrage d'images et détection de contours

09/04/19

L'objectif de cette séance de 3 heures est d'appliquer et de comprendre diverses méthodes de filtrages d'image et de détection de contours, dans le domaine spatial ou fréquentuel. Le TP doit être réalisé en Python, et testé sur les images disponibles dans le répertoire *images*.

1 Ajustement d'images par correction gamma

Soit I une image, $I : \mathbb{R}^n \mapsto \mathbb{R}$. Appliquer une correction gamma revient à calculer en tout voxel (x, y):

$$I_f(x, y) = I(x, y)^\gamma \tag{1}$$

avec $\gamma \in \mathbb{R}$.

1.1 Appliquer une correction gamma à l'image , avec $\gamma \in [0,1; 0,5; 0,9]$. Commenter.

1.1.1 Même chose avec $\gamma \in [2; 3; 4]$. Commenter.

2 Filtrage d'images

2.1 Filtrage dans le domaine spatial

On rappelle la formule de la distribution gaussienne 2D centrée en 0 d'écart-type σ:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{\frac{-1}{2\pi\sigma^2}(x^2+y^2)} \tag{2}$$

2.1.1 Calculer la fonction *gaussian_kernel* qui permet de créer un noyau gaussien d'écart-type σ, de moyenne nulle, et de taille NxN.

2.1.2 Calculer la fonction *mean_kernel* qui permet de créer un noyau moyenneur de taille NxN.

2.1.3 Calculer la fonction *median_kernel* qui permet d'appliquer un filtre médian de taille NxN à une image I.

2.1.4 Appliquer un filtrage gaussien, moyenneur et gaussien à l'image boat512.gif.

2.1.5 Tester différentes valeurs de N et σ.

2.1.6 Ajouter un bruit blanc gaussien $\sigma = 20$ à l'image filtrée utiliser la fonction np.random.normal. Comparer les filtres gaussien ($\sigma \in [1; 2]$), médian ($N \in [3; 5; 7]$) et moyenneur ($N \in [3; 5; 7]$).

2.1.7 Ajouter un bruit impulsionnel à l'image, en remplaçant aléatoirement l'intensité de n% des pixels par la valeur minimale d'intensité, et l'intensité de n% des pixels par la valeur maximale d'intensité, avec $n \in [4, 20]$. Comparer les filtres gaussien ($\sigma \in [1; 2]$), médian ($N \in [3; 5; 7]$) et moyenneur ($N \in [3; 5; 7]$).

2.2 Filtrage Gaussien dans le domaine fréquentiel

Soit I une image de dimension MxN. La transformée de Fourier d'un noyau gaussien de variance σ et de moyenne 0 s'écrit:

$$G(u, v) = e^{-\frac{1}{2}\pi^2\sigma^2\left(\left(\frac{u}{M}\right)^2+\left(\frac{v}{N}\right)^2\right)} \tag{3}$$

avec u,v fréquences.

2.2.1 Quelle relation relie la transformée de Fourier d'une convolution d'images à la transformée de Fourier de chacune de ces images ? A l'aide de ce résultat, calculer la transformée de Fourier de l'image convoluée par un noyau gaussien de variance σ et de moyenne 0.

2.2.2 Afficher la transformée de Fourier calculée, et la comparer à la transformée de Fourier de l'image (les fréquences doivent être centrées en 0). En déduire la nature du filtre gaussien (passe-haut, passe-bas ou passe-bande).

2.3 Correction de flou dans le domaine fréquentiel

2.3.1 Appliquer un filtre gaussien $\sigma = 3.5$ à l'image. Calculer la transformée de Fourier de l'image convoluée par un filtre gaussien $\sigma = 3.5$. Déconvoluer l'image filtrée par ce même noyau. Afficher l'image réelle obtenue. Commenter.

2.3.2 Même chose avec $\sigma = 10$.

Testons le filtrage de Wiener, dont la transformée de Fourier s'écrit:

$$W = \frac{\overline{F(u,v)G(u,v)}}{\lambda + \left|F(u,v)G(u,v))\right|^2} \tag{4}$$

Avec F transformée de Fourier de l'image I, G transformée de Fourier du noyau gaussien d'écart-type σ.

2.3.3 Appliquer le filtre de Wiener à l'image 2.2.3 avec $\lambda = 0{,}01$. Commenter.

3 Détection de contours

3.1 Approche gradient

On rappelle que le gradient d'une image I est un vecteur $\vec{G} = (G_x, G_y)$ avec $G_x(x,y) = \frac{\partial I(x,y)}{\partial x}$, $G_y(x,y) = \frac{\partial I(x,y)}{\partial y}$. Ces dérivées peuvent être approximées par différences finies:

$$G_x(x,y) = I(x+1,y) - I(x-1,y) \tag{5}$$

$$G_y(x,y) = I(x,y+1) - I(x,y-1) \tag{6}$$

3.1.1 Calculer et afficher G_x, G_y et $G = \sqrt{G_x + G_y}$. Commenter.

Le gradient horizontal G_x et vertical G_y d'une image I peut être approximé à partir de la convolution de cette image I par les masques suivants:

$$Opérateur\ de\ Sobel : G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$Opérateur\ de\ Prewitt : G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

3.1.2 Calculer et afficher les gradients G_x et G_y de l'image. Commenter.

3.1.3 Calculer et afficher la magnitude du gradient $G = \sqrt{G_x + G_y}$. Commenter.

3.1.4 Appliquer un seuillage sur G afin d'obtenir l'image binaire des contours. Commenter.

3.2 Approche Laplacien

On donne les deux masques de convolution suivant:

$$G_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} G_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

3.2.1 Calculer le Laplacien L à l'aide des deux masques ci -dessus, puis obtenir l'image binaire des contours. Commenter.

3.3 Amélioration de contraste

3.3.1 Calculer l'image suivante: $G(x, y) = I(x, x) - \lambda L(x, y)$, L laplacien de I, $\lambda = 0,05$. Commenter. Tester plusieurs valeurs de λ.

3.3.2 Calculer le laplacien L de l'image I puis remplacer I par $I - \lambda L$ avec $\lambda = 0,05$. Itérer 10 fois. Commenter.