Université de Marne-La-Vallée
Institut Gaspard MONGE

Master d'Informatique **Module de Traitement d'Images**

ITI

CORRIGĒ

EXAMEN

Année 2011-2012

On répondra directement sur les feuilles d'examen en indiquant en pied de page ses NOM et Prénom. L'usage de documents n'est pas autorisé.

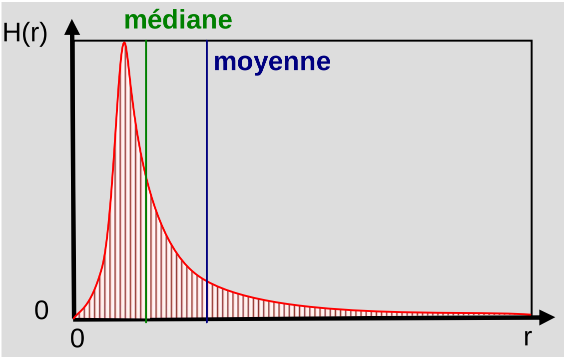
1. Moyenne et médiane

a. Quelle est la différence entre la moyenne et la médiane d'une distribution ?

1 • Moyenne – est le barycentre d'une distribution (somme des valeurs divisée par le nombre d'échantillons).

• Médiane – est la valeur pour laquelle exactement 50% des individus ont une valeur inférieure et 50% des individus restants ont une valeur supérieure.

1 b. Illustrer sur un histogramme ci-contre un cas où les valeurs de la moyenne et de la médiane sont très différentes. On reportera au mieux ces deux valeurs.



2. Calibration min-max

Soit une image en entrée [r(i,j), i=0..(M-1), j=0..(N-1)] de 16 bits non signés qu'on désire traiter pour produire en sortie une image en 8 bits non signés r'(i,j), étirée (« *stretchée* ») linéairement entre les valeurs minimum et maximum des comptes numériques r(i,j) de l'image en entrée.

a. Quelle est l'intervalle des valeurs permises dans l'image en entrée ?

1 $\forall$ i=0..(M-1), j=0..(N-1), r(i,j) $\in$ $[0, 2^{16}-1]$ = **[0,65535]**

b. Sachant que l'image en entrée présente une valeur d'arrière-plan (*background*) égale à 0, écrire en pseudo-code un programme permettant de retrouver les valeurs min et max des comptes numériques r(i,j) en entrée.

1
```
min ← 65535
max ← 0
Pour i ← 0, i < M, i ← i+1 Faire
      Pour j ← 0, j < N, j ← j+1 Faire
            Si (r(i,j) > 0) Alors
                  Si (r(i,j) < min) Alors min ← r(i,j) FinSi
                  Si (r(i,j) > max) Alors max ← r(i,j) FinSi
            FinSi
      FinFaire
FinFaire
```

c. Par quelle formule mathématique (et non pseudo-code informatique) est calculée la valeur r'(i,j) du stretching linéaire entre min et max à partir de la valeur en entrée r(i,j). On n'oubliera pas de traiter le cas du background.

1
$$r'(i,j) = f\left(r(i,j)\right) = \begin{cases} 0 & si \quad r(i,j) = 0 \quad (background) \\ 1 & si \quad r(i,j) < \min \\ 1 + Arrondi\left(254 \times \dfrac{r(i,j) - \min}{\max - \min}\right) & si \quad \min \le r(i,j) \le \max \\ 255 & si \quad r(i,j) > \max \end{cases}$$

NOM : ………………………………….. Prénom : …………………………….

d. Pour calculer l'image de sortie, on peut s'épargner le calcul pixel par pixel du stretching linéaire détaillé dans la question c précédente. Une telle optimisation est opérée en calculant puis appliquant une table de look-up (LUT). Ecrire le programme dans un pseudo-code permettant de réaliser ces deux opérations.

2

```
Etape 1 – Initialiser la LUT
LUT[0] ← 0
Pour r ← 1, r < 65536, r ← r+1 Faire
      Si (r < min) Alors
            LUT[r] ← 1
      Sinon
            Si (r > max) Alors
                  LUT[r] ← 65535
            Sinon
                  LUT[r] ← Arrondi(255 * (r(i,j) – min) / (max – min))
            FinSi
      FinSi
FinFaire

Etape 2 – Calculer l'image de sortie en appliquant la LUT
Pour i ← 0, i < M, i ← i+1 Faire
      Pour j ← 0, j < N, j ← j+1 Faire
            r'(i,j) ← LUT[r(i,j)]
      FinFaire
FinFaire
```

e. Soient m la moyenne de l'image en entrée, calculer la valeur de la moyenne m' de l'image de sortie en fonction de m et des paramètres de la formule mathématique trouvée à la question 2.c. Pour simplifier les calculs, on supposera qu'il n'y a pas de pixels de background. On justifiera toutes les étapes du calcul.

1,5

$$
\begin{aligned}
\overline{m'} &= \frac{1}{MN} \times \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} r'(i,j) \\
&= \frac{1}{MN} \times \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(255 \times \frac{r(i,j) - \min}{\max - \min} \right) \\
&= \frac{255}{\max - \min} \times \left(\frac{1}{MN} \times \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} r(i,j) \right) - \frac{1}{MN} \times \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(255 \times \frac{\min}{\max - \min} \right) \\
&= \frac{255}{\max - \min} \times \overline{m} - \frac{1}{MN} \times MN \times \left(255 \times \frac{\min}{\max - \min} \right) \\
\overline{m'} &= 255 \times \frac{\overline{m} - \min}{\max - \min}
\end{aligned}
$$

f. De la même manière, calculer la valeur de l'écart-type σ' en fonction de l'écart-type σ en entrée.

1,5

$$
\begin{aligned}
\sigma'^2 &= \frac{1}{MN} \times \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} r'(i,j)^2 - \overline{m'}^2 \\
&= \frac{1}{MN} \times \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(255 \times \frac{r(i,j) - \min}{\max - \min} \right)^2 - \left(255 \times \frac{\overline{m} - \min}{\max - \min} \right)^2 \\
&= \frac{1}{MN} \times \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(\left(\frac{255}{\max - \min} \right)^2 \times r(i,j)^2 - 2 \times \left(\frac{255}{\max - \min} \right) \times r(i,j) \times \min + \left(\frac{255}{\max - \min} \right)^2 \times \min^2 \right) - \\
&\quad \left(\left(\frac{255}{\max - \min} \right)^2 \times \overline{m}^2 - 2 \times \left(\frac{255}{\max - \min} \right) \times \overline{m} \times \min + \left(\frac{255}{\max - \min} \right)^2 \times \min^2 \right) \\
&= \left(\frac{255}{\max - \min} \right)^2 \times \frac{1}{MN} \times \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} r(i,j)^2 - \left(\frac{255}{\max - \min} \right)^2 \times \overline{m}^2 \\
&= \left(\frac{255}{\max - \min} \right)^2 \times \sigma^2 \\
\sigma &= \left(\frac{255}{\max - \min} \right) \times \sigma
\end{aligned}
$$

3. Matrices de convolution

a. Somme de filtres

2,5

On désire rehausser les contours verticaux et horizontaux d'une image en « ajoutant à l'image originale la moitié du gradient Nord-Sud et le tiers du gradient Est-Ouest ».

Quels sont les coefficients, le gain et l'offset de la matrice M de convolution 3x3 qui réalise un tel effet ?

Justifier sa réponse en détaillant les opérations sur les matrices.

$$M = Id + \frac{1}{2} \times G_{NS} + \frac{1}{3} \times G_{EO}$$

$$M = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \frac{1}{2} \times \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} + \frac{1}{3} \times \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

$$M = \frac{1}{6} \times \left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} -3 & -3 & -3 \\ 0 & 0 & 0 \\ 3 & 3 & 3 \end{bmatrix} + \begin{bmatrix} 2 & 0 & -2 \\ 2 & 0 & -2 \\ 2 & 0 & -2 \end{bmatrix} \right)$$

$$M = \frac{1}{6} \times \begin{bmatrix} -1 & -3 & -5 \\ 2 & 6 & -2 \\ 5 & 3 & 1 \end{bmatrix}$$

b. Convolutions successives

2,5

Une image 8-bits non signés est convolée par le filtre de « Gauss 3x3 » puis l'image résultante est convolée à son tour par le filtre « Sobel NW-SE ». Quel est le filtre de convolution opérant en une seule passe qui équivaudrait à ces deux convolutions successives ?

On montrera les étapes du calcul.

$$Filtre = \frac{\alpha}{\frac{4\sqrt{2}}{3} \times 3} \times \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & +1 \\ 0 & +1 & +2 \end{bmatrix} \otimes \left[\frac{1}{9} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} + 0 \right] + 128$$

$$Filtre = \frac{\alpha}{\frac{4\sqrt{2}}{3} \times 3 \times 9} \times \begin{bmatrix}
\begin{bmatrix} -2\times1 & & \\ & & \\ & & \end{bmatrix} & \begin{bmatrix} -2\times2 & -1\times1 & \\ & & \\ & & \end{bmatrix} & \begin{bmatrix} -2\times1 & -1\times2 & 0\times1 \\ & & \\ & & \end{bmatrix} & \begin{bmatrix} & -1\times1 & 0\times2 \\ & & \\ & & \end{bmatrix} & \begin{bmatrix} & & 0\times1 \\ & & \\ & & \end{bmatrix} \\
\begin{bmatrix} -2\times2 & & \\ -1\times1 & & \\ & & \end{bmatrix} & \begin{bmatrix} -2\times4 & -1\times2 & \\ -1\times2 & 0\times1 & \\ & & \end{bmatrix} & \begin{bmatrix} -2\times2 & -1\times4 & 0\times2 \\ -1\times1 & 0\times2 & 1\times1 \\ & & \end{bmatrix} & \begin{bmatrix} & -1\times2 & 0\times4 \\ & 0\times1 & 1\times2 \\ & & \end{bmatrix} & \begin{bmatrix} & & 0\times2 \\ & & 1\times1 \\ & & \end{bmatrix} \\
\begin{bmatrix} -2\times1 & & \\ -1\times2 & & \\ 0\times1 & & \end{bmatrix} & \begin{bmatrix} -2\times2 & -1\times1 & \\ -1\times4 & 0\times2 & \\ 0\times2 & 1\times1 & \end{bmatrix} & \begin{bmatrix} -2\times1 & -1\times2 & 0\times1 \\ -1\times2 & 0\times4 & 1\times2 \\ 0\times1 & 1\times2 & 2\times1 \end{bmatrix} & \begin{bmatrix} & -1\times1 & 0\times2 \\ & 0\times2 & 1\times4 \\ & 1\times1 & 2\times2 \end{bmatrix} & \begin{bmatrix} & & 0\times1 \\ & & 1\times2 \\ & & 2\times1 \end{bmatrix} \\
\begin{bmatrix} & & \\ -1\times1 & & \\ 0\times2 & & \end{bmatrix} & \begin{bmatrix} & & \\ -1\times2 & 0\times1 & \\ 0\times4 & 1\times2 & \end{bmatrix} & \begin{bmatrix} & & \\ -1\times1 & 0\times2 & 1\times1 \\ 0\times2 & 1\times4 & 2\times2 \end{bmatrix} & \begin{bmatrix} & & \\ & 0\times1 & 1\times2 \\ & 1\times2 & 2\times4 \end{bmatrix} & \begin{bmatrix} & & \\ & & 1\times1 \\ & & 2\times2 \end{bmatrix} \\
\begin{bmatrix} & & \\ & & \\ 0\times1 & & \end{bmatrix} & \begin{bmatrix} & & \\ & & \\ 0\times2 & 1\times1 & \end{bmatrix} & \begin{bmatrix} & & \\ & & \\ 0\times1 & 1\times2 & 2\times1 \end{bmatrix} & \begin{bmatrix} & & \\ & & \\ & 1\times1 & 2\times2 \end{bmatrix} & \begin{bmatrix} & & \\ & & \\ & & 2\times1 \end{bmatrix}
\end{bmatrix} + 128$$

$$Filtre = \frac{\alpha}{\frac{4\sqrt{2}}{3} \times 3 \times 9} \times \begin{bmatrix} -2 & -5 & -4 & -1 & 0 \\ -5 & -12 & -8 & 0 & 1 \\ -4 & -8 & 0 & 8 & 4 \\ -1 & 0 & 8 & 12 & 5 \\ 0 & 1 & 4 & 5 & 2 \end{bmatrix} + 128$$

4. Filtrage adaptatif

1 a. Quelle est la définition d'un filtre adaptatif r'(i,j)=F(r(i,j)) ?

Un filtre adaptatif calcule pixel par pixel la valeur r'(i,j) de l'image de sortie à partir de la valeur r(i,j) du pixel correspondant dans l'image en entrée et de statistiques locales telles la moyenne m(i,j) et l'écart-type σ(i,j) estimées autour de ce pixel.

Le filtre de Lee ci-dessous a été mis au point pour atténuer le chatoiement (« *speckle* ») dans les images radar.

$$r'(i,j) = \overline{m}(i,j) + \frac{\sigma^2(i,j)}{\sigma^2(i,j) + \sigma_n^2 \times \overline{m}(i,j)} \times \left[r(i,j) - \overline{m}(i,j)\right]$$

1 b. Indiquer le sens des différentes variables

- r(i,j) est la radiométrie du pixel en entrée.
- r'(i,j) est la radiométrie du pixel en sortie.
- m(i,j) est la moyenne locale estimée autour du pixel (i,j).
- σ(i,j) est l'écart-type local estimé autour du pixel (i,j).
- σ_n est l'écart-type centré réduit du bruit de speckle donné à-prori.

3 c. Programmer en pseudo-code le Filtre de Lee pour une fenêtre statistique de taille CxC (C impair) et en ne traitant pas les bords de l'image. L'image est supposée sans background. On ne demande pas d'optimisations.

```
Pour i ← C/2, i < M-C/2, i ← i+1 Faire
      Pour j ← C/2, j < N-C/2, j ← j+1 Faire
            sum1 ← 0
            sum2 ← 0
            Pour k ← -C/2, k ≤ C/2, k ← k+1 Faire
                  Pour l ← -C/2, l ≤ C/2, l ← l+1 Faire
                        sum1 ← sum1 + r[i+k,j+l]
                        sum2 ← sum2 + r[i+k,j+l] * r[i+k,j+l]
                  FinFaire
            FinFaire
            m(i,j) ← sum1 / (C * C)
            σ(i,j) ← SQRT(sum2 / (C * C) – m(i,j) * m(i,j))
            r'(i,j) ← m(i,j) + σ(i,j) * σ(i,j) /
                      (σ(i,j) * σ(i,j) + σn * σn * m(i,j)) *
                      (r(i,j) – m(i,j))
      FinFaire
FinFaire
```