Renaud Sirdey et Fabrice Houssier

Égalisation d'histogramme

Éditions des Nik's News
www.niksnews.com/editions/

1998

1 Méthode

L'égalisation d'histogramme est une technique qui modifie la répartition des niveaux de gris dans une image afin d'en améliorer le contraste. Dans ce qui suit on ne considérera que des images carrées 256×256 digitalisées sur 256 niveaux de gris.

1.1 Changement de loi

Considérons une variable aléatoire X continue de loi quelconque

$$X \rightsquigarrow \mathcal{L}(\cdots) \tag{1}$$

définie sur un intervalle $[x_{min}, x_{max}]$.

On note

- $f_X(x)$ la fonction de densité de X ;
- $F_X(x) = P(X < x) = \int_{x_{min}}^{x} f(x)dx$ la fonction de répartition de X.

On cherche à trouver une fonction φ strictement croissante telle que

$$Y = \varphi(X) \rightsquigarrow \mathcal{U}_{[y_{min}, y_{max}]} \tag{2}$$

soit

- $f_Y(y) = \frac{1}{y_{max} - y_{min}} \mathbf{1}_{[y_{min}, y_{max}]}$;
- $F_Y(y) = \frac{1}{y_{max} - y_{min}} y \mathbf{1}_{[y_{min}, y_{max}]}$.

On a

$$F_Y(y) = P(Y < y) = P(\varphi(X) < y) = P(X < \varphi^{-1}(y)) = F_X(\varphi^{-1}(y)) \tag{3}$$

Par conséquent, pour que (2) soit vérifiée, il est nécessaire que

$$\varphi^{-1}(y) = \frac{1}{y_{max} - y_{min}} F_X^{-1}(y) \tag{4}$$

pour tout $y \in [0, y_{max} - y_{min}]$,
afin que

$$\begin{aligned} F_Y(y) &= F_X(\varphi^{-1}(y)) \\ &= F_X(\frac{1}{y_{max} - y_{min}} F_X^{-1}(y)) \\ &= \frac{1}{y_{max} - y_{min}} F_X(F_X^{-1}(y)) \ (*) \\ &= \frac{1}{y_{max} - y_{min}} y \end{aligned} \tag{5}$$

(*) : la fraction $\frac{1}{y_{max}-y_{min}}$ est sortie en utilisant les propriétés de linéarité de l'opérateur d'intégration.

Finalement

$$\varphi(x) = (y_{max} - y_{min})F_X(x) + y_{min} \tag{6}$$

pour tout $x \in [x_{min}, x_{max}]$.

La variable aléatoire $Y = \varphi(X)$ suit alors une loi uniforme sur $[y_{min}, y_{max}]$.

1.2 Egalisation d'histogramme

Chaque pixel de l'image à égaliser est considéré comme la réalisation d'une variable aléatoire $X = I_o(\vec{u})$ [1] de loi inconnue définie sur $[0, 255]$. L'ensemble des pixels de l'image représente alors un échantillon iid de la variable aléatoire X.

On estime $f_X(x)$ et $F_X(x)$ en utilisant respectivement l'histogramme[2]

$$h(x) = \frac{n_x}{n} \tag{7}$$

et l'histogramme cumulé

$$h_c(x) = \frac{\sum_{k=x_{min}}^{x} n_k}{n} \tag{8}$$

La conversion des valeurs en chaque pixel de l'image originale se fait alors de la façon suivante

$$i_e(\vec{u}) = (n_{gris} - 1)h_c(i_o(\vec{u})) \tag{9}$$

dans la mesure où l'on souhaite que $I_e(\vec{u})$ suive une loi uniforme sur $[0, n_{gris}-1]$.

Afin d'effectuer tous les calculs avec des valeurs entières, l'histogramme ne doit pas être normalisé (pour des raisons de perte de précision), la normalisation se fera sur l'histogramme cumulé en divisant ce dernier non pas par son maximum (le nombre de pixels de l'image) mais par $\frac{n}{n_{gris}-1}$ de tel sorte que son maximum soit égale à $n_{gris} - 1$.

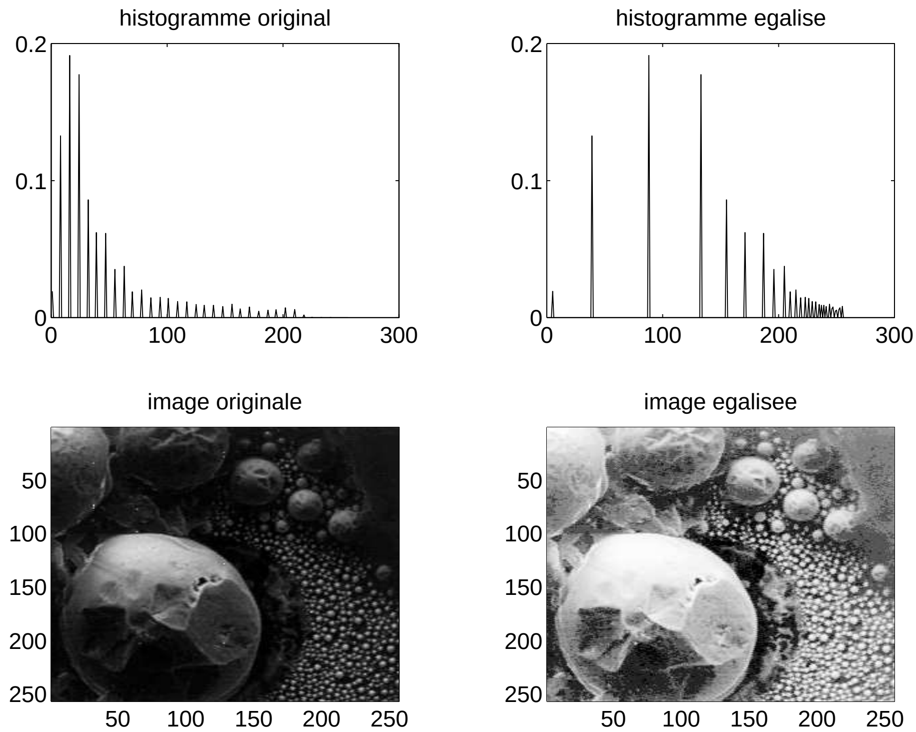
2 Résultats

La figure 1 montre le résultat de l'égalisation de l'histogramme sur une image $256 \times 256 \times 256$. La figure 2 représente les histogrammes cumulés avant et

1. $\vec{u} = (x, y)$.
2. n_x est le nombre de pixels de niveau de gris égal à x et n est le nombre de pixels total.

après égalisation (pour plusieurs niveaux de gris). On remarque que les histogrammes cumulés des images égalisées approchent très nettement la fonction de répartition théorique d'une variable aléatoire uniforme[3] et ce d'autant mieux que le nombre de niveaux de gris est faible.

FIG. **1** – Résultat de l'égalisation d'histogramme.

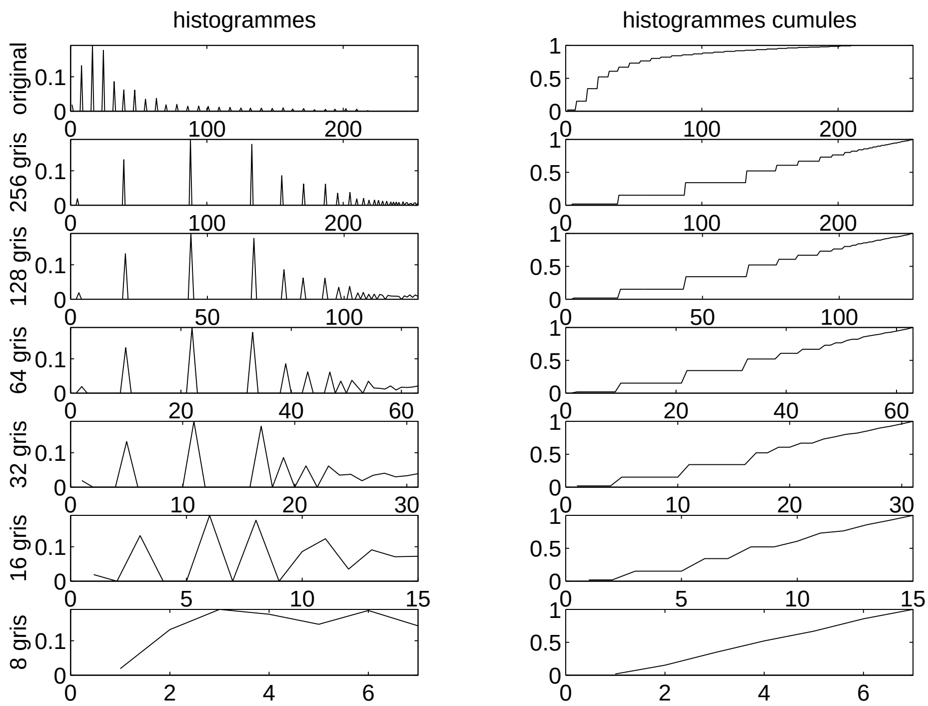


3 Pourquoi ?

Dans ce qui suit, on s'efforce de prouver que plus le nombre de niveaux de gris de l'image de sortie est petit, plus on a des chances de trouver la « bonne transformation » (c'est-à-dire la transformation qui garantit que les pixels de l'image résultante suivront une loi uniforme). Une bonne façon de procéder semble consister à étudier la variance de la transformation.

3. $y = \frac{1}{p-1}x$ où $p - 1$ est le nombre de niveaux de gris souhaité.

FIG. **2** – Histogrammes et histogrammes cumulés avant et après égalisation (pour plusieurs niveaux de gris).



On a
$$I_o(\vec{u}) = X \rightsquigarrow \mathcal{L}(\cdots) \tag{10}$$
variable aléatoire discrète à valeurs dans $[0, 255]$ de loi inconnue, et

- $f_X(x) = P(X = x)$: fonction de densité,
- $F_X(x) = P(X \leq x)$: fonction de répartition.

On s'intéresse à la variable aléatoire $H_c(x)$ qui représente l'histogramme cumulé non normalisé au point x (c'est-à-dire le nombre de réalisations qui ont été $\leq x$). La variable aléatoire Z « $X \leq x$ ou non » est un schéma de Bernouilli de paramètre $p = F_X(x)$. Or, l'histogramme cumulé au point x ($H_c(x)$) consiste à compter le nombre de fois que ce schéma aura pris la valeur 1 sur l'échantillon représenté par les pixels de l'image initiale. La variable aléatoire $H_c(x)$ est donc binomiale de paramètres n (nombre de pixels de l'image[4]) et $p = F_X(x)$.

Son espérance est
$$E[H_c(x)] = nF_X(x) \tag{11}$$
et sa variance est
$$Var[H_c(x)] = nF_X(x)(1 - F_X(x)) \tag{12}$$
Finalement, on s'intéresse à la variable aléatoire $T(x)$ qui représente la transformation que l'on va appliquer aux pixels ayant la valeur x dans la première image.

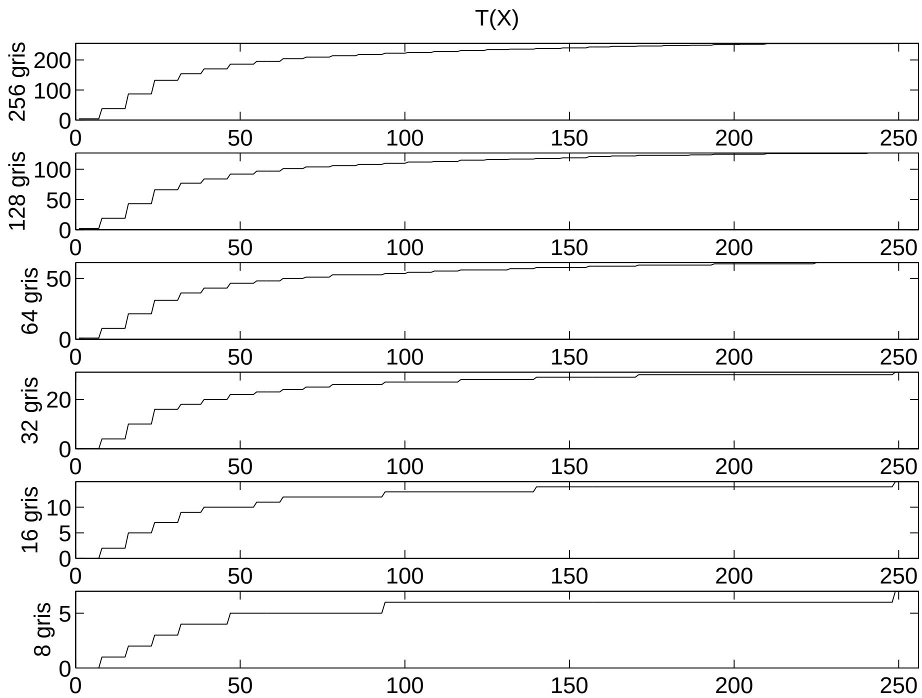
On a
$$T(x) = \alpha H_c(x) + y_{min}, \text{ avec } \alpha = \frac{y_{max} - y_{min}}{n} \tag{13}$$
et
$$E[T(x)] = \alpha E[H_c(x)] + y_{min} = (y_{max} - y_{min})F_X(x) + y_{min} \tag{14}$$
ce qui est bien la transformation recherchée.

De plus
$$Var[T(x)] = \alpha^2 Var[H_c(x)] = \frac{(y_{max} - y_{min})^2}{n} F_X(x)(1 - F_X(x)) \tag{15}$$
Si on pose $\beta = y_{max} - y_{min}$ et que l'on considère la dérivée de $Var[T(x)](\beta)$ par rapport à β
$$\frac{\delta Var[T(x)]}{\delta\beta}(\beta) = \underbrace{\frac{2\beta}{n}}_{\geq 0} \underbrace{F_X(x)}_{\geq 0} \underbrace{(1 - F_X(x))}_{\geq 0} \tag{16}$$

[4] Qui représente la taille de l'échantillon iid observé.

on s'aperçcoit qu'il s'agit d'une fonction croissante de β qui n'est ni plus ni moins que le nombre de niveaux de gris de l'image finale moins 1. Autrement dit, la variance de la transformation décroit avec le nombre de niveaux de gris souhaités. Par conséquent, plus le nombre de niveaux final est faible plus on a des chances de trouver la « bonne transformation » qui permettra de transformer la variable aléatoire X en variable uniforme.

FIG. **3** – Transformations pour plusieurs niveaux de gris.



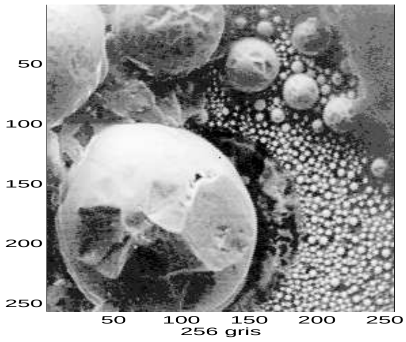
Conclusion

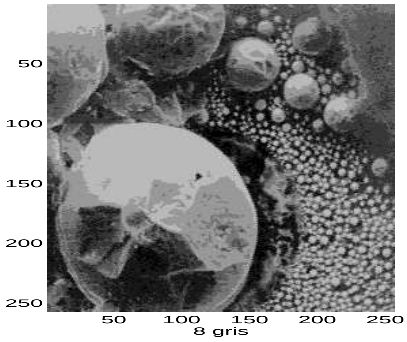
Finalement, la méthode d'égalisation d'histogramme revient à faire un compromis entre

- la quantité d'information perdue (lorsque le nombre de niveaux de gris final est inférieur au nombre initial, voir figure 4) ;

– la qualité de la transformation utilisée (dont la variance est une fonction croissante du nombre de niveaux de gris).

Fig. 4 – Illustration de la perte d'information induite par une égalisation sur trop peu de niveaux de gris.





4 Mise en oeuvre

On suppose que l'image est mémorisée dans un tableau de type

```
unsigned char d[256*256];
```

et que l'histogramme et l'histogramme cumulé le seront respectivement dans

```
unsigned long h[256],r[256];
```

4.1 Calcul de l'histogramme

```
memset(h,0,sizeof(unsigned long)<<8);
for(k=0;k<256*256;k++) h[d[k]]++;
```

4.2 Calcul de l'histogramme cumulé

```
r[0]=h[0];
for(k=1;k<256;k++) r[k]=r[k-1]+h[k];
```

4.3 Normalisation de l'histogramme cumulé

```
for(k=0;k<256;k++)
{ r[k]/=_H_NORM_; r[k]+=_X_MIN_; }
```

Où

```
#define _WIDTH_  256
#define _HEIGHT_ 256
#define _X_MIN_  0
#define _X_MAX_  255
#define _N_GREY_ _X_MAX_-_X_MIN_+1
#define _H_NORM_ (_WIDTH_*_HEIGHT_)/(_N_GREY_-1)
```

4.4 Egalisation

```
for(k=0;k<256*256;k++) d[k]=(unsigned char)r[d[k]];
```