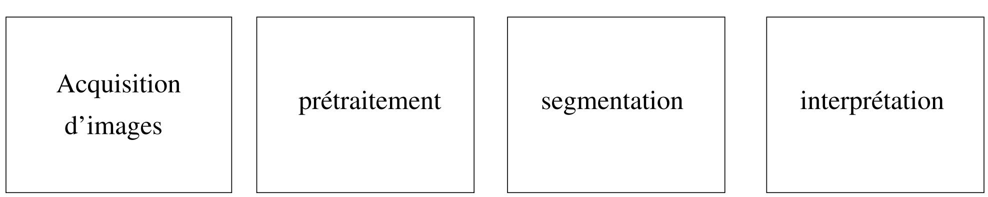
Analyse d'images

La vision nous permet de percevoir et d'interpreter le monde qui nous entoure. La vision artificielle a pour but de reproduire certaines fonctionnalités de la vision humaine au travers de l'**a**nalyse d'images. C'est un problème difficile en raison du fait que l'information disponible : des images 2D fournies par des capteurs (CCD, ...), correspond à une projection du monde 3D. La projection 3D-2D entraîne une perte d'informations importante, de plus l'information disponible n'est pas parfaite (numérisation des capteurs, déformation des objectifs, bruitages).

L'analyse d'images regroupe plusieurs disciplines que l'on classe en deux catégories :

- Les processus de bas-niveaux, qui nécessitent très peu d'informations sur le contenu des images. Il s'agit ici des processus de filtrage, d'amélioration et de restauration d'images, processus que nous regroupons sous le terme *traitement d'images*, ainsi que d'extraction d'indices.
- Les processus de haut-niveaux, qui fonctionnent en aval de ceux de bas-niveaux, et qui peuvent nécessiter des informations sur le contenu des images. Il s'agit de la reconstruction tri-dimensionnelle, la reconnaissance de formes, les processus cognitifs de façon générale.



L'analyse d'images est une chaîne de traitement de l'information.

Le traitement d'images (ou pré-traitement) regroupe l'ensemble des processus visant à améliorer les caractéristiques d'une image.

- Le lissage local : il s'agit de supprimer le bruit, ou les petites variations, présent dans une image. L'intensité d'un pixel est transformé en fonction des intensités sur un petit voisinage du pixel.
- L'amélioration d'images consiste à modifier les caractéristiques visuelles de l'image (contraste, ...) pour faciliter son interprétation par l'œil humain.
- La restauration d'images a pour but de supprimer les dégradations subies par une image à l'aide de connaissance *a priori* sur ces dégradations.

1 Lissage local

Le lissage local (ou filtrage) consiste à supprimer le bruit présent dans une image en étudiant, pour chaque pixel, les valeurs d'intensité sur son voisinage.

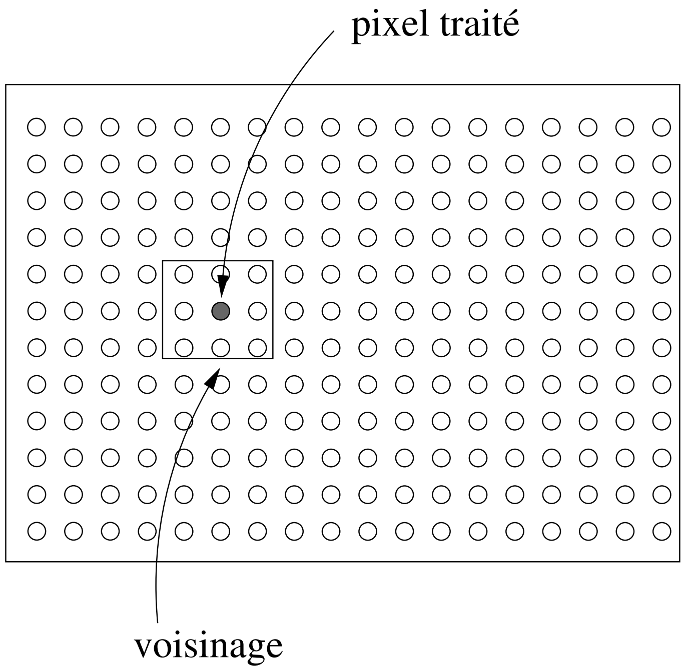
1.1 Moyennage

Une première classe d'approche est basée la redondance d'informations. La nouvelle valeur d'un pixel est calculée par moyennage des valeurs sur un voisinage. Cette opération linéaire peut être vue comme la convolution discrète de l'image par un masque.

$$I'(i,j) = \sum_{(m,n)\in\mathcal{V}} h(m,n)\; I(i-m, j-n),$$

$$\sum_{(m,n)\in\mathcal{V}} h(m,n) = 1,$$

ou I est l'intensité de l'image d'origine, I' est l'intensité de l'image filtrée, $\mathcal{V}$ est le voisinage utilisé et h est le masque de convolution.



Le moyennage sur un voisinage 3x3 :

$$h = 1/9 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

☞ Le moyennage est un filtre passe-bas.

☞ Rend l'image floue, en particulier les contours.

☞ Élimine les dégradations locales de faibles dimensions. Valide lorsque les objets présents dans l'image sont de dimensions supérieures aux dégradations.

Une amélioration du filtre moyenne consiste à jouer sur les valeurs des coefficients du masque :

$$h_1 = 1/10 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

C'est le cas des filtres binomiaux pour lesquels les valeurs des coefficients sont générés par le triangle de Pascal :

$$h_2 = 1/16 \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

- ☞ Approximation discrète d'un filtre Gaussien adaptée au filtrage des bruits Gaussiens.

1.2 Filtres médians

Les filtres de moyennage ont tendance à rendre l'image floue et donc à perdre de l'informations sur les contours caractérisés par des fortes variations d'intensité. Pour diminuer cet effet, on ne moyenne plus sur le voisinage mais on prend la valeur médiane sur ce voisinage. C'est le filtre médian.

Exemple pour un voisinage 3x3 :

$$I = \begin{bmatrix} 2 & 12 & 12 \\ 2 & 12 & 60 \\ 2 & 2 & 12 \end{bmatrix}$$

La valeur médiane est ici 12.

- ☞ Filtre non-linéaire.
- ☞ Élimine le bruit impulsionnel.
- ☞ Préserve l'information de contour et peut être appliqué itérativement.
- ☞ Élimine les contours très fins. Un voisinage adapté permet de limiter cet effet.

1.3 Autres filtres

Toujours dans l'objectif d'améliorer les caractéristiques du filtre et en particulier la préservation des contours dans l'image, on peut utiliser des filtres plus évolués. Il en existe un certain nombre. Nous donnons ici l'exemple d'un filtre faisant intervenir des variations d'intensité.

Soit $d(i,j,k,l)$ la variations d'intensité entre le pixel (i,j) et (k,l) :

$$d(i,j,k,l) = \begin{cases} |I(i,j) - I(k,l)| \text{ si } I(i,j) \neq I(k,l), \\ 1/2 \text{ sinon,} \end{cases}$$

alors les coefficients du masque de convolution sont déterminés par :

$$h(m,n) = \frac{1/d(i,j,i+m,j+n)}{\sum_{(m,n)\in\mathcal{V}} 1/d(i,j,i+m,j+n)}$$

- ☞ La distance est plus importante sur un contour que dans une région homogène.
- ☞ Les points fortement bruités ont peu d'influence (h(m,n) petit).
- ☞ Si le pixel est sur un contour, les pixels voisins de même région auront un poids important alors que les pixels voisins de région différente auront peu de poids. Le contour est de cette manière préservé.

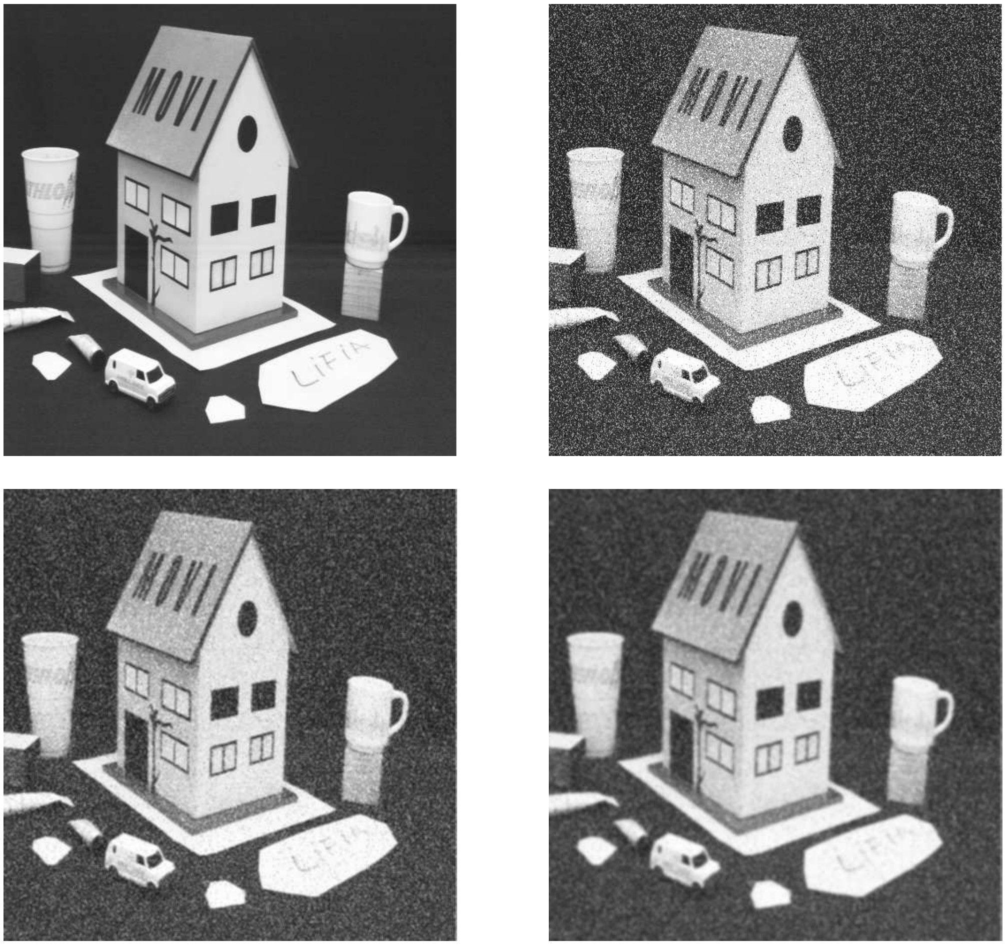


Figure 1: Filtrages par moyennage : 1 et 3 itétations.

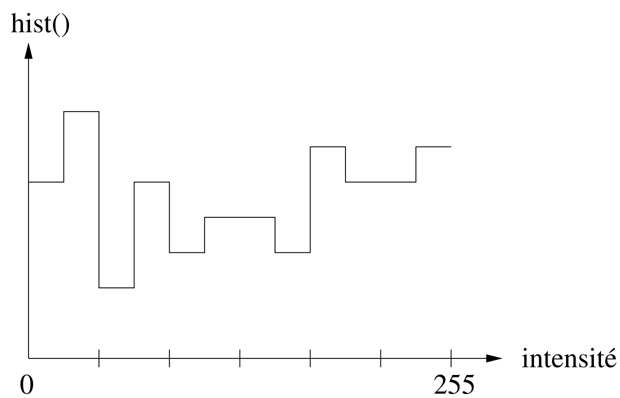
Figure 2: Filtrages médians : 1 et 3 itérations.

2 Amélioration d'images

L'amélioration d'images consiste à modifier les caractéristiques visuelles de l'image de manière à en faciliter son interpretation par l'œil humain. Il peut s'agir de rehausser les contrastes, d'accentuer certaines intensités pour mettre en valeur une région, ... Les histogrammes sont fréquemment utilisés pour effectuer ce type d'opérations

2.1 Les histogrammes

L'histogramme d'une image $hist(i)$ est la fonction qui associe à une valeur d'intensité i le nombre de pixels dans l'image ayant cette valeur.



- ☞ Pour une image couleur, il y a un histogramme par composante.
- ☞ L'histogramme peut être normalisé pour donner une estimation de la densité de probabilité des pixels :

$$p(i) = hist(i)/\sum_j hist(j),$$

$$\sum_i p(i) = 1.$$

- ☞ Un histogramme peut avoir un pic (unimodale), deux pics (bimodale) ou plusieurs pics (multimodale).

Algorithme

```
int row, col, rowmax, colmax;
int count[GREYMAX];
for(row = 0; row < rowmax; row++)
for(col = 0; col < colmax; col++)
   hist[image[row][col]]++;
```

2.2 Modifications d'histogrammes

Pour modifier les caractéristiques de l'image (accentuer les contrastes en géneral), une approche générale consiste à appliquer une fonction qui associe à chaque valeur d'intensité dans l'image une nouvelle valeur. Cette fonction va modifier l'histogramme de l'image.

Soit $i, i < MaxInt$ les valeurs d'intensité de l'image traitée , on considère alors les transformations du type :

$$i' = T(i),$$

qui donne une nouvelle valeur d'intensité i' pour chaque valeur i de l'image. On suppose que la fonction $T()$ est telle que :

- $T(i)$ est monotone (souvent croissante) sur l'intervalle. Cette condition assurant que l'ordre des intensités est préservé après transformation.
- $0 \leq T(i) \leq MaxInt$ pour $0 \leq i \leq MaxInt$ qui garantit que la nouvelle image est cohérente avec les niveaux d'intensité autorisés.
- La transformation inverse satisfait les deux conditions précédentes.

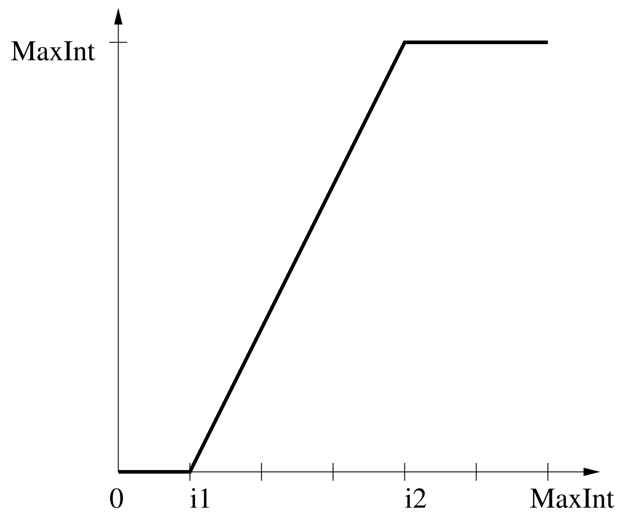
Quelques exemples représentatifs de modifications d'histogrammes.

2.3 Transformations linéaires

Les transformations linéaires d'histogrammes sont nombreuses et variées. Elles permettent d'accentuer une zone d'intensité ou de modifier la répartition des valeurs d'intensité. Un exemple de transformation linéaire est le recadrage.

Le recadrage consiste à modifier l'intervalle des valeurs d'intensité (la dynamique) de façon à obtenir pour l'image améliorée un intervalle de valeurs maximal. Si $[i1, i2]$ est l'intervalle de l'image traitée, alors la transformation équivalente s'écrit simplement :

$$T(i) = MaxInt\ (i - i1)/(i2 - i1).$$



2.4 Transformations non-linéaires

Une modification d'histogramme très répandue pour augmenter le contraste de manière automatique est la linéarisation d'histogramme. Le principe est de transformer l'image de manière à obtenir un histogramme plat, soit une distribution uniforme des intensités. Cela revient à maximiser l'entropie de l'image et donc à obtenir théoriquement une image présentant une information maximale.

La fonction de distribution idéale après transformation est :

$$p'(i') = 1/Maxint.$$

Supposons tout d'abord que l'on soit dans le cas continu et que la transformation T soit strictement croissante. Alors :

$$p'(i') = p(i)\,\frac{di}{di'},$$

avec $i' = T(i)$. D'où :

$$di' = MaxInt\; p(i)\; di,$$

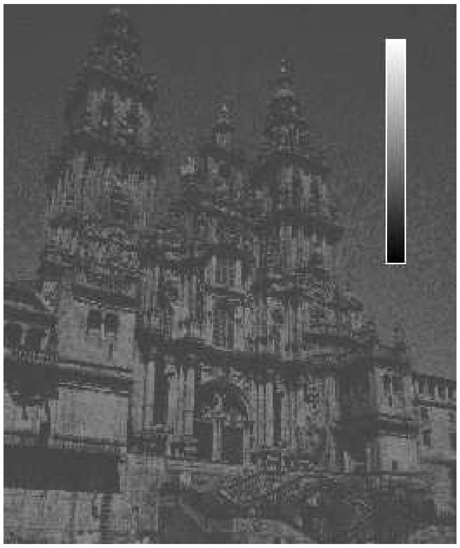
soit :

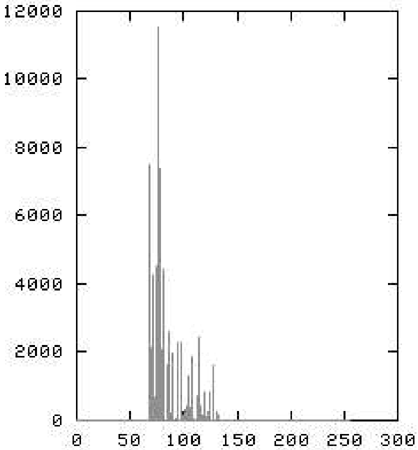
$$i' = T(i) = MaxInt\; \int_0^i p(w)\, dw.$$

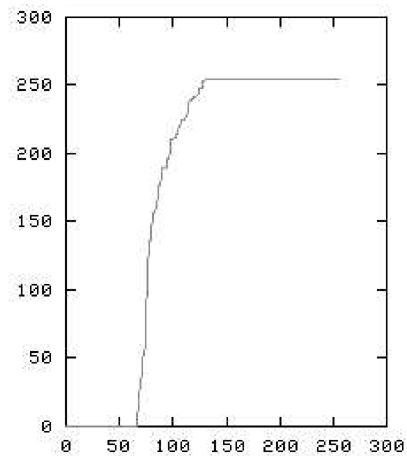
Dans le cas discret, la transformation s'écrit :

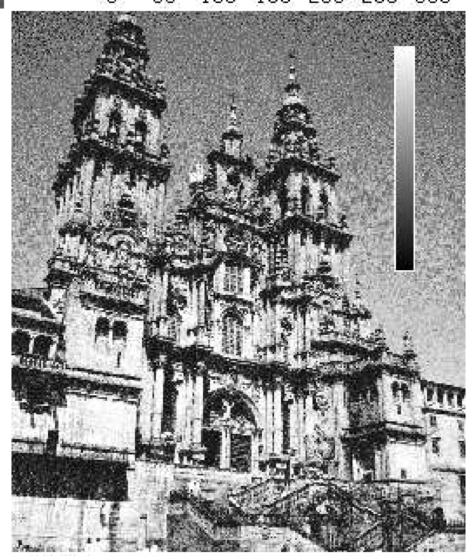
$$T(i) = MaxInt\; \sum_{j=0}^{i} hist(j) / \sum_{j=0}^{MaxInt} hist(j),$$

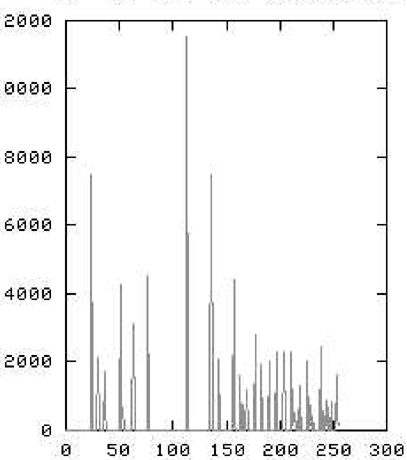
- ☞ A noter que dans le cas discret, et du fait que l'histogramme soit une approximation d'une fonction de densité de probabilité, l'histogramme résultant est très rarement parfaitement plat.
- ☞ Pour une image majoritairement claire la linéarisation va augmenter la dynamique de la partie sombre de l'histogramme au détriment de la partie claire.
- ☞ Permet de faire des comparaisons d'images sur une même base.
- ☞ L'opération peut s'effectuer par régions : linéarisation adaptative.

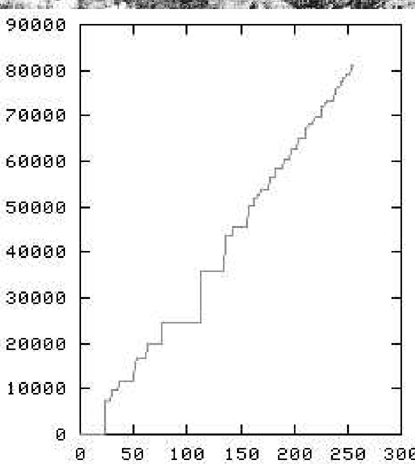



12000
10000
8000
6000
4000
2000
0
0 50 100 150 200 250 300


300
250
200
150
100
50
0
0 50 100 150 200 250 300




12000
10000
8000
6000
4000
2000
0
0 50 100 150 200 250 300


90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 50 100 150 200 250 300