

Filtrage non-linéaire

- Filtres morphologiques
- Filtres homomorphiques
- Filtres de Volterra
- Filtres d'ordre



Filtres morphologiques

- Issus de la morphologie mathématique il s'agit d'une généralisation des opérateurs binaires classiques de morphologie mathématique.

- Définition « didactique »

Définir un voisinage : soit VBp, le voisinage du pixel p ayant pour fonction structurante B (définition de la forme du voisinage). B centré en p définit un ensemble de pixels autour de p VBp.

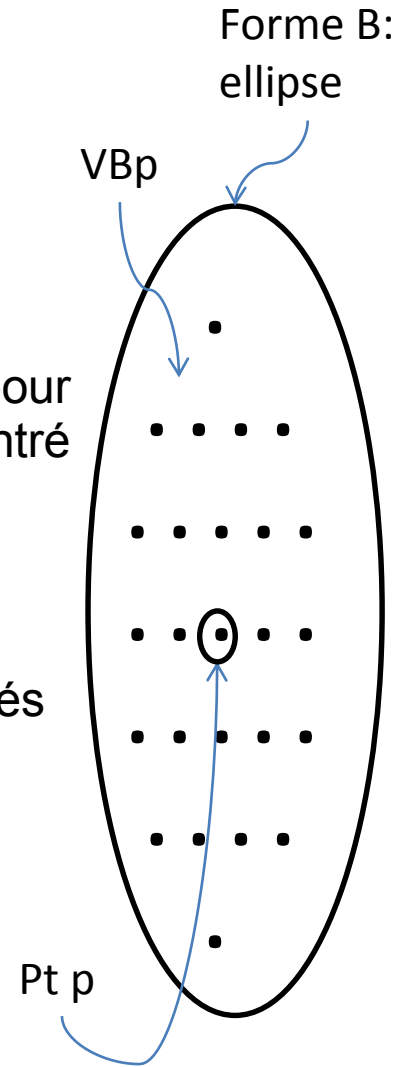
Les deux opérateurs de base sont :

- ✓ L'érosion : Tout pixel p de I prend la valeur minimale des intensités présentes dans l'ensemble des pixels de VBp.

$$E(I) = I \ominus B = \min \{ I(p-p'), p \in I, p-p' \in VBp \}$$

- ✓ La dilatation : Tout pixel p de I prend la valeur maximale des intensités présentes dans l'ensemble des pixels de VBp.

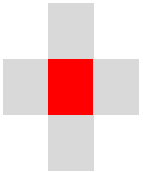
$$D(I) = I \oplus B = \max \{ I(p-p'), p \in I, p-p' \in VBp \}$$



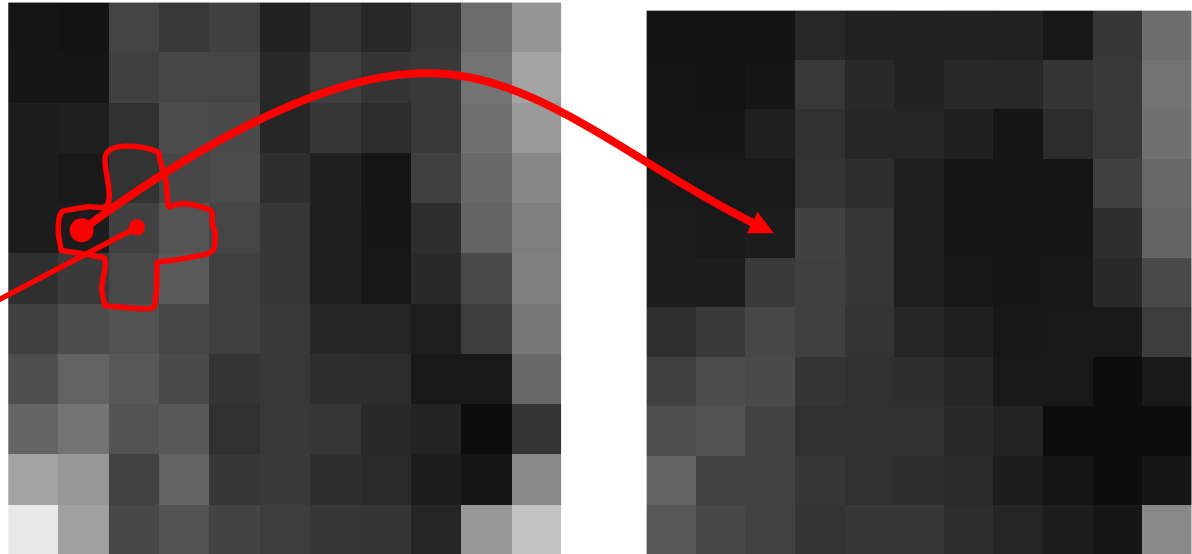
Filtres morphologiques

- Érosion : élément structurant B: Croix 5 pixels

B



P:
Pixel traité



Filtres morphologiques

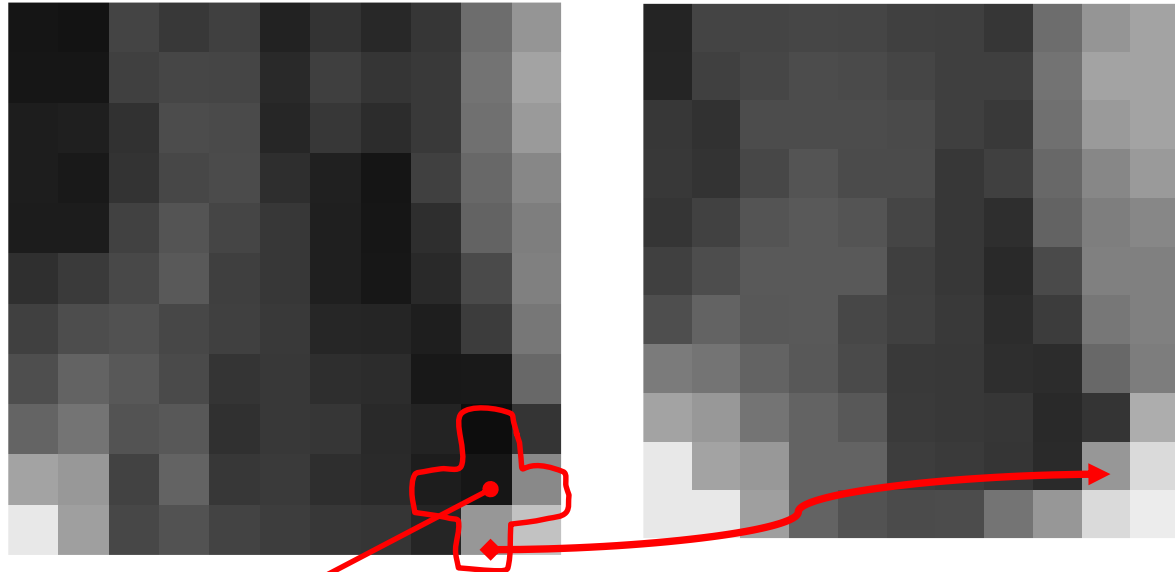
- Érosion : Sur cet exemple, donnez un élément structurant possible pour obtenir ce résultat



Filtres morphologiques

- Dilatation : élément structurant B: Croix 5 pixels

B

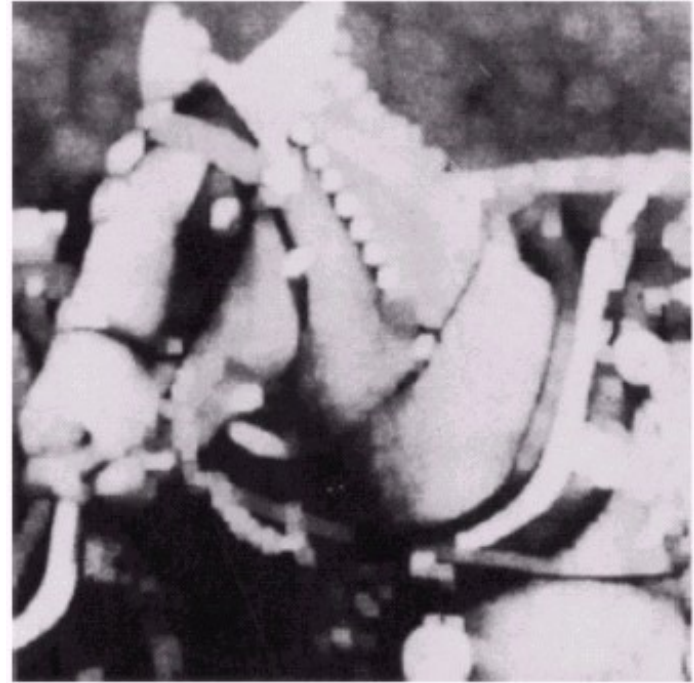


P:
Pixel traité



Filtres morphologiques

- Dilatation : quel élément structurant pourrait donner ce résultat?

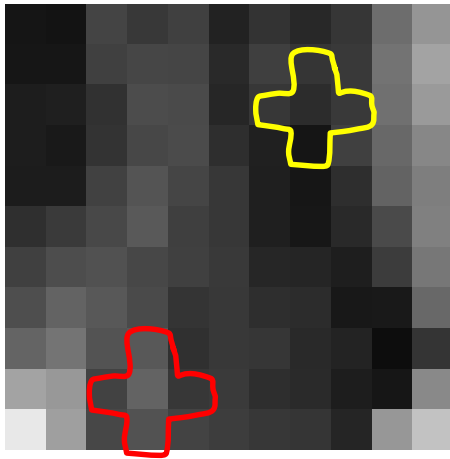
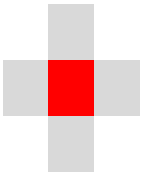


Filtres morphologiques

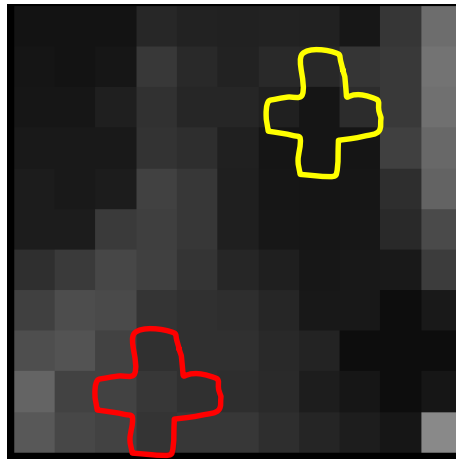
– ouverture: élément structurant B: Croix 5 pixels

$$I_o = (I \circ B) = (I \ominus B) \oplus B = D(E(I))$$

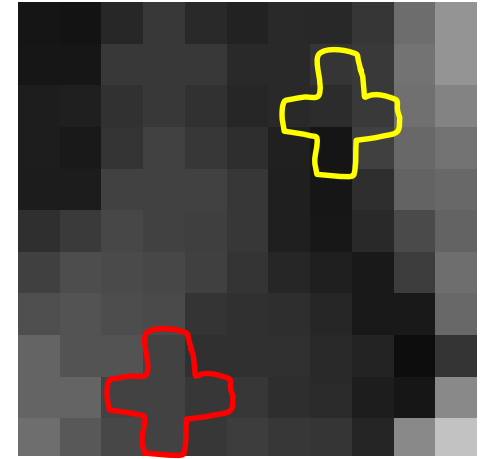
B



I



E(I)



D(E(I))

Propriété de l'érosion mais en revenant vers l'image d'origine



Filtres morphologiques

– Ouverture

$$I_o = (I \circ B) = (I \ominus B) \oplus B = D(E(I))$$

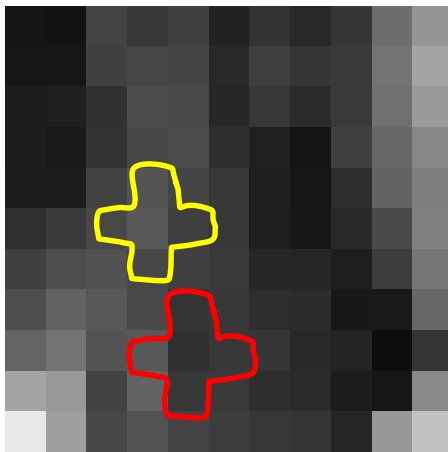


Filtres morphologiques

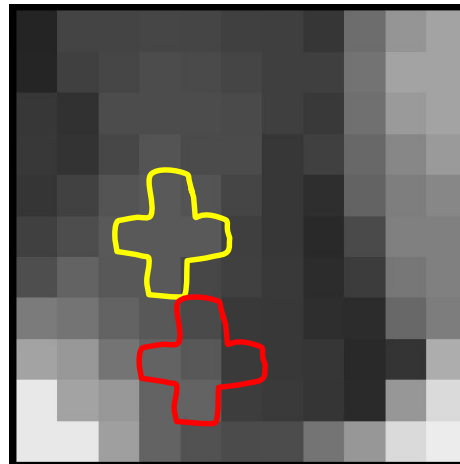
- Fermeture: élément structurant B: Croix 5 pixels

$$I_f = I \bullet B = (I \oplus B) \ominus B = E(D(I))$$

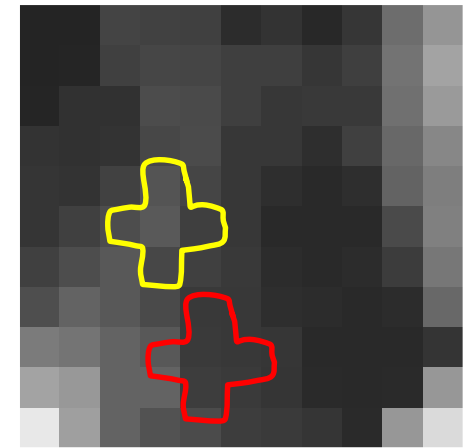
B



I



D(I)



E(D(I))

Propriété de la dilatation mais en revenant vers l'image d'origine



Filtres morphologiques

– Fermeture

$$If = I \bullet B = (I \oplus B) \ominus B = E(D(I))$$



Filtres morphologiques

– Chapeau haut-de-forme (Top-hat)

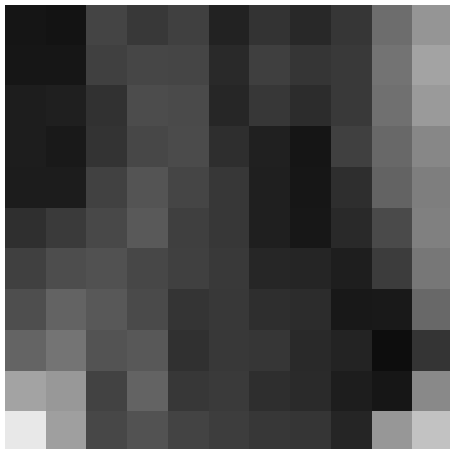
$$I' = I - (I \circ B) \text{ (détection des pics clairs)}$$

- Par analogie on définit aussi :

$$I'' = (I \bullet B) - I \text{ (détection des pics sombres ou bottom-hat)}$$

$$I''' = \sup\{(I \bullet B) - I ; I - (I \circ B)\} \text{ (détection de tous les pics)}$$

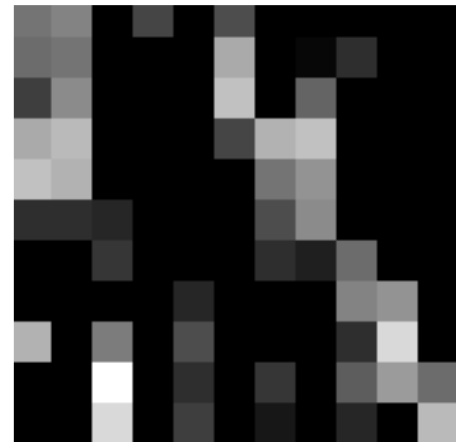
B



I



Top-hat



Bottom-hat



Filtres morphologiques

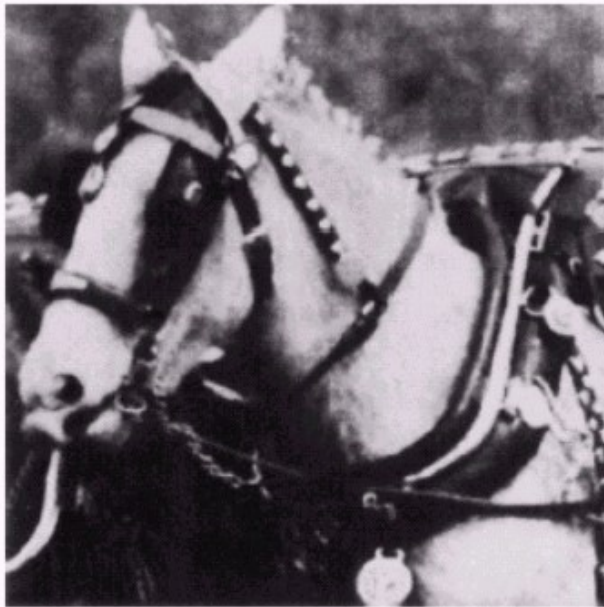
– Chapeau haut-de-forme (Top-hat)

$$I' = I - (I \circ B) \text{ (détection des pics clairs)}$$

- Par analogie on définit aussi :

$$I'' = (I \bullet B) - I \text{ (détection des pics sombres ou bottom-hat)}$$

$$I''' = \sup\{(I \bullet B) - I ; I - (I \circ B)\} \text{ (détection de tous les pics)}$$



Filtres morphologiques

- Gradient et Laplacien morphologiques
 - Gradient : Dilaté – Erodé
 - $I_g = (I \oplus B) - (I \ominus B) = D(I) - E(I)$
 - Laplacien : Détection des pentes ascendantes et descendantes de la fonction gradient
 - $I_{lap} = ((I \oplus B) - I) - (I - (I \ominus B)) / 2 = (D(I) + E(I) - 2I) / 2$



Filtres morphologiques - Illustrations

Image originale



Image fermeture avec disk1



Image ouverture avec disk1



Image gradient avec disk1



Filtre morphologique

- Exemple : Application de la transformation « chapeau haut de forme » pour un réhaussement de détail (ou réhaussement de contraste local).

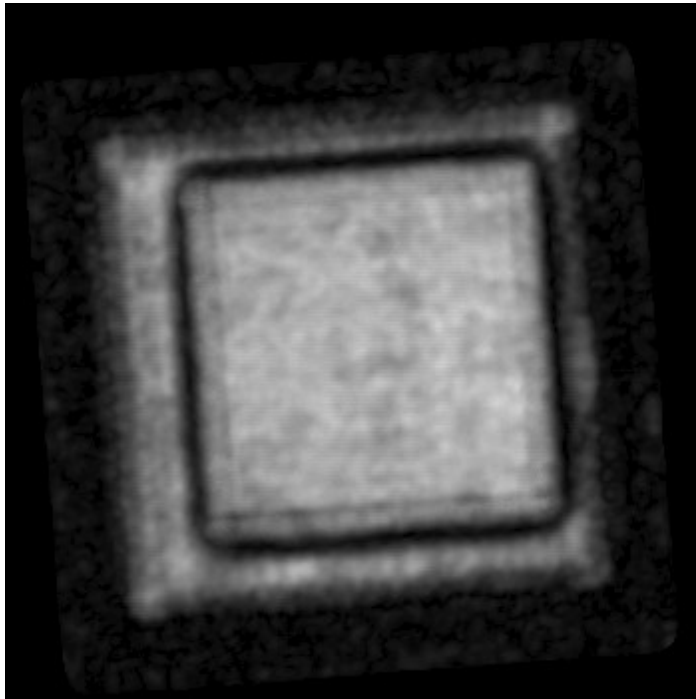
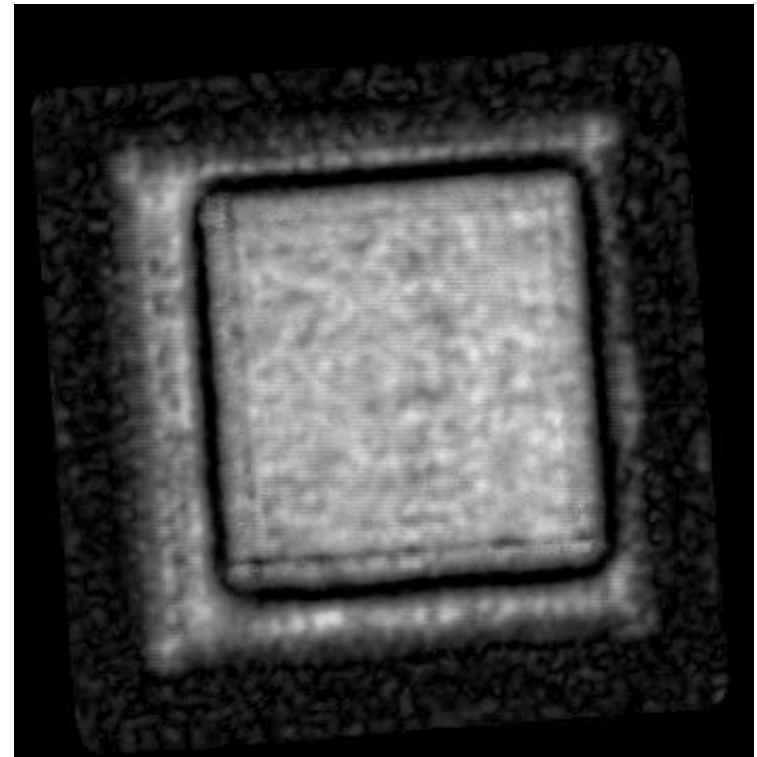


Image originale

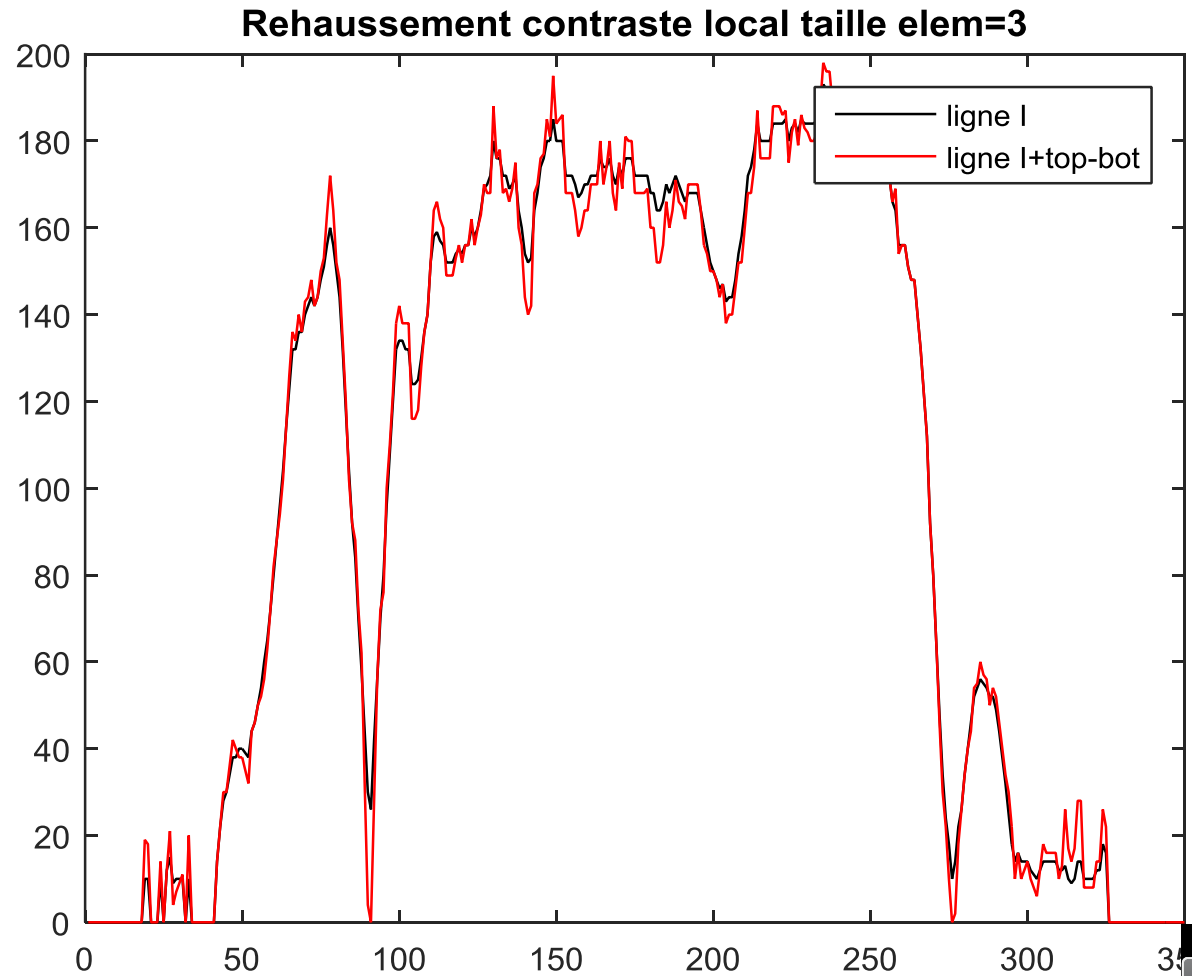
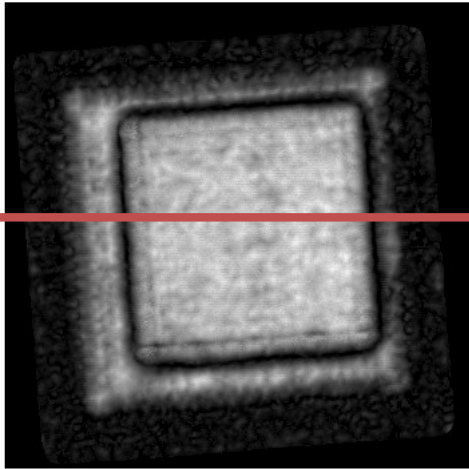


$(\text{Image} + \text{Top_Hat}) - \text{Bottom_Hat}$



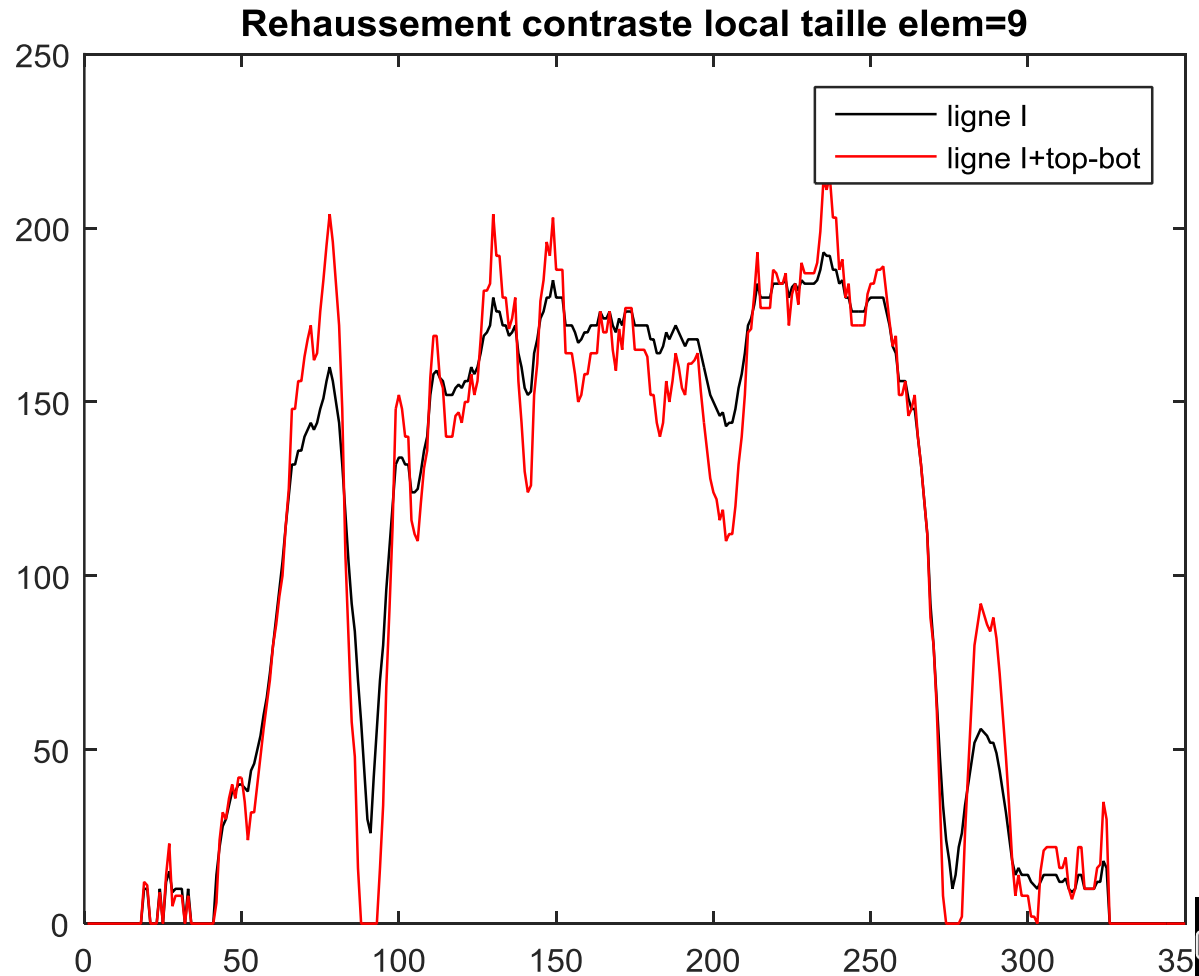
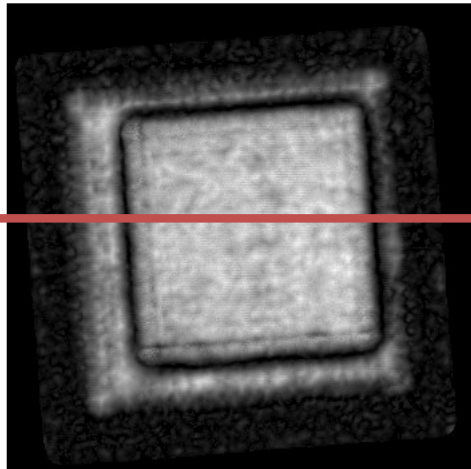
Détails : élément structurant taille 3

(original+top-hat)-image-bottom-hat



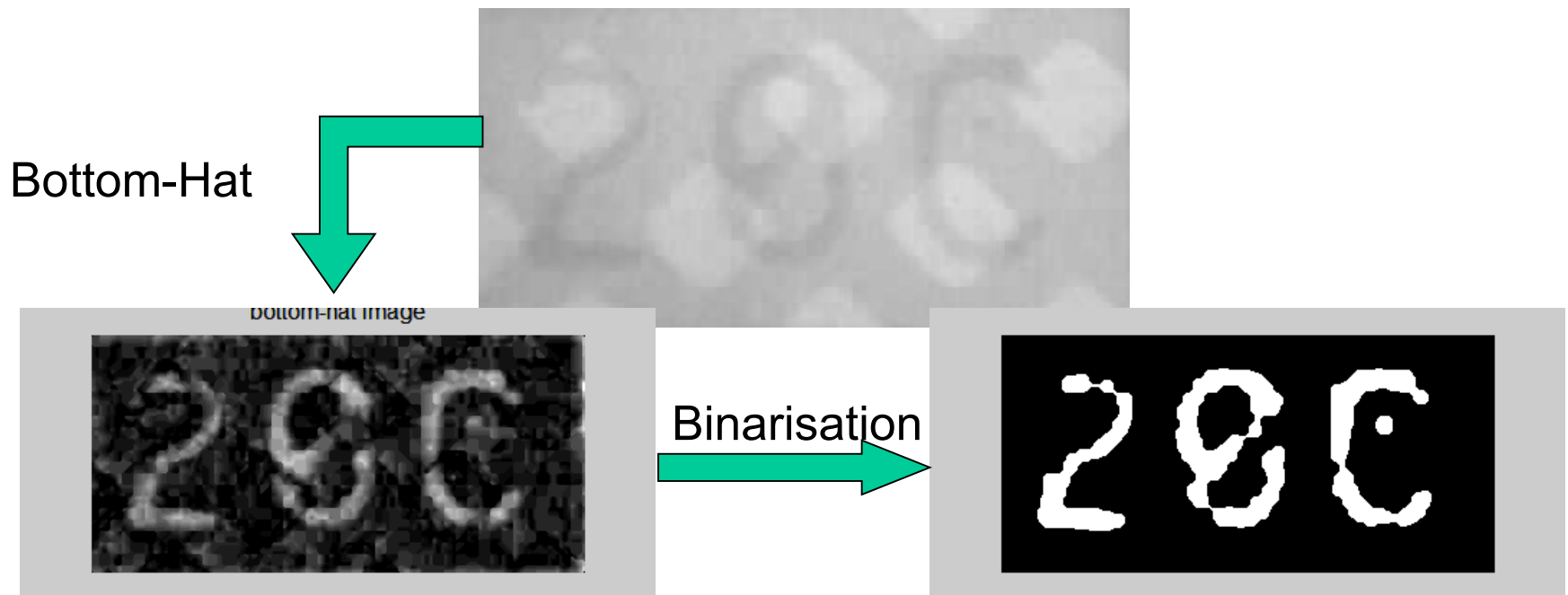
Détails : élément structurant taille 9

(original+top-hat)-image-bottom-hat



Filtres morphologiques

- Exemple : Extraction du code dans l'image
 - Observation : le code est localement le niveau de gris le plus faible
 - Détecter les « vallées » (image bottom-hat) est donc une solution
 - Binariser l'image bottom-hat nous donne des résultats satisfaisants



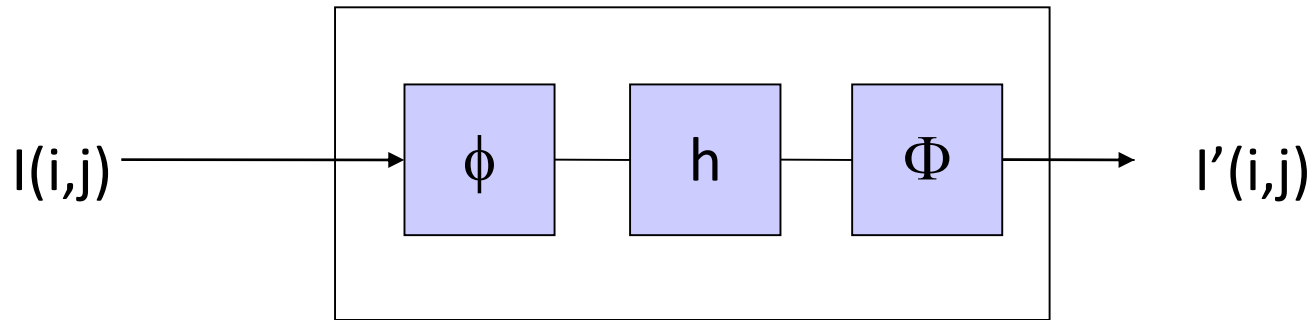
Filtres morphologiques

- Filtres alternés séquentiels
 - Itération de succession d'ouvertures et fermetures de taille croissante. Soit B l'élément structurant.
 - $FAS(I) = Fr^B(O^{rB}(\dots F^{2B}(O^{2B}(F^B(O^B(I))))\dots))$
 - Permet d'éliminer progressivement les pics clairs puis sombres de taille croissante.
- A la base d'algorithmes de segmentation



Filtres homomorphiques

- Principe : combinaison de filtres linéaires et non-linéaires



ϕ et Φ sont non-linéaires réciproques

h : réponse impulsionnelle d'un filtre linéaire (PB en général)

Ces opérateurs sont particulièrement utilisés pour des bruits convolutifs mais aussi multiplicatifs

Exemple : soit $I(i,j)=I(i,j).B(i,j)$ (bruit multiplicatif) alors en prenant $\phi = \ln$ et $\Phi=\exp$,
Le filtre h agit sur une image qui sera vue comme une image avec bruit additif .



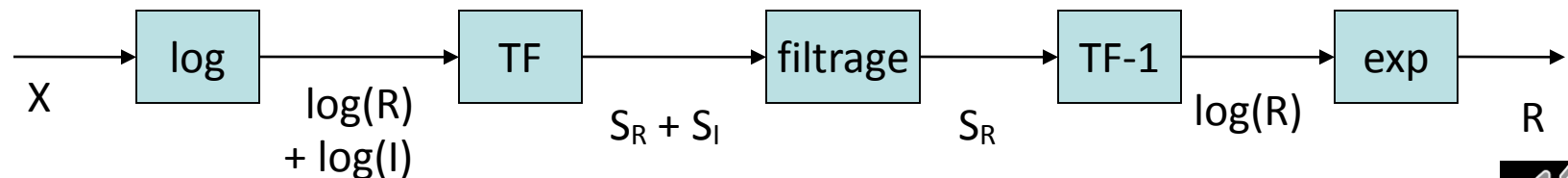
Filtres homomorphiques

Composition d'une img X : résulte en chaque pixel p du produit

- de la réflectance $R(p)$ caractéristique des propriétés physiques et optiques du point p. L'img de réflectance définit le contenu de la scène
- de l'illumination de la scène $I(p)$. Si l'illumination n'est pas constante sur toute l'img, l'img est de mauvaise qualité (zones surexposées, sous-exposées...)

$$X(p) = R(p) \cdot I(p)$$

Idée : récupérer R et s'affranchir de I en transformant cette multiplication en une somme puis traitement par filtrage fréquentiel



Hyp : illumination = composante BF



Filtres homomorphiques

Exemples



Exemple filtre homomorphique sur image couleur



Exemple filtre homomorphique sur image couleur



Filtres d'ordre

1	2	3
4	...	
	...	L

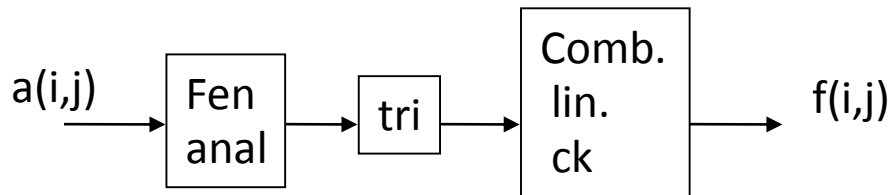
Principe : Soit une fenêtre d'analyse comportant L pixels. On appelle j ème statistique d'ordre, notée $a(j)$, la valeur de rang j dans a , où a est l'image d'entrée et $a(i,j)$ l'intensité du pixel (i,j) .

$$a(1) \leq a(2) \leq \dots \leq a(L)$$

Pour un filtre d'ordre la sortie y est définie comme une fonction des statistiques d'ordre. Les données sont prises en compte selon leur rang dans la fenêtre d'analyse.

L-filtres

$$f(i, j) = \sum_{k=1}^L c_k a(k)$$



Objectifs : trouver les meilleurs c_k pour réduire les bruits considérés.

Hyp : on ne modifie pas une zone homogène:

$$\sum_{k=1}^L c_k = 1$$

Les valeurs de coefs optimaux vont dépendre de la ddp du bruit



- Si bruit gaussien :

$$c_k = \frac{1}{L} \quad \forall k \quad \rightarrow \text{identique au filtre moyenne}$$

- Si bruit uniforme :

$$c_k = \begin{cases} 1/2 & \text{si } k = 1 \text{ ou } k = L \\ 0 & \text{sinon} \end{cases} \quad \rightarrow \text{identique au filtre milieu}$$

-Filtre médian

Le L-filtre le plus utilisé et le plus courant est le filtre médian

$$c_k = \begin{cases} 1 & \text{si } k = \frac{L+1}{2} \\ 0 & \text{sinon} \end{cases} \quad \text{On note : } y = \text{med}(a)$$

Filtre idéal pour les bruits impulsionnels



Exemple

120	120	120	120	120	120	120	120
120	120	120	120	120	120	120	120
120	120	120	120	120	120	120	120
120	120	120	120	5	120	120	120
120	120	120	120	120	120	120	120
120	120	255	120	120	120	120	120
120	120	120	120	120	120	120	120
120	120	120	120	120	120	120	120

Tri

5	120	120	120	120	120	120	120	120	120
---	-----	-----	-----	-----	-----	-----	-----	-----	-----

a(1)

a(9)

5 est remplacé par 120, dans le cas du filtre moyenne, on aurait $(120 \cdot 8 + 5) / 9 = 107$

Dans cette image il y a visiblement 2 points parasites, le filtre médian permet donc de les supprimer.





Image perturbée par
un défaut de largeur 1 ou 2 pixels



Quel filtre permet de retrouver l'image
sans défaut?



- Filtre médian pondéré (Weighted Median Filter)

- Définition :

$$f(I) = \text{med}(W \diamond I)$$

- où \diamond désigne la répétition, W les poids associés au masque et I l'image

- Exemple :

$$W = \begin{pmatrix} 2 & 1 & 2 \\ 1 & 3 & 1 \\ 2 & 1 & 2 \end{pmatrix} \quad I = \begin{pmatrix} 56 & 37 & 42 \\ 19 & 16 & 12 \\ 41 & 83 & 65 \end{pmatrix}$$

$$f(I) = \text{med}([56, 56, 37, 42, 42, 19, 16, 16, 16, 12, 41, 41, 83, 65, 65] = \\ [12, 16, 16, 16, 19, 37, 41, 41, 42, 42, 56, 56, 65, 65, 83]) = 41$$

- Ce principe peut s'appliquer à tous les filtres d'ordre, comme par exemple le filtre milieu pondéré

Ici : $f(I) = (12 + 83) / 2 = 47.5$



- Variante des filtres médians
 - Dans l'ensemble E des valeurs triées, on sélectionne le morceau du vecteur E de longueur L tel que ce sous-ensemble soit le plus compact possible :
- Dynamique la plus faible
- Communément : $L = n/2$ ou $(n+1)/2$ avec n longueur de E
 - La valeur médiane est ensuite calculée sur ce sous-ensemble compact



- Applications
 - Filtrage de bruit (type passe-bas)
 - Zoom



Application : Zoom

- Par 2 dans chaque direction

$$\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{bmatrix} \Rightarrow$$

- Algorithme par les médians

$$I_{ij}^{00} = a_{ij}$$

$$I_{ij}^{11} = \text{MEDIAN}[a_{ij}, a_{i+1j}, a_{ij+1}, a_{i+1j+1}]$$

$$I_{ij}^{01} = \text{MEDIAN}[a_{ij}, a_{ij+1}, 0.5 \diamond I_{i-1j}^{11}, 0.5 \diamond I_{i+1j}^{11}]$$

$$I_{ij}^{10} = \text{MEDIAN}[a_{ij}, a_{i+1j}, 0.5 \diamond I_{ij-1}^{11}, 0.5 \diamond I_{ij+1}^{11}]$$

$$\begin{bmatrix} I_{1,1}^{00} & I_{1,1}^{01} & I_{1,2}^{00} & I_{1,2}^{01} & I_{1,3}^{00} & I_{1,3}^{01} \\ I_{1,1}^{10} & I_{1,1}^{11} & I_{1,2}^{10} & I_{1,2}^{11} & I_{1,3}^{10} & I_{1,3}^{11} \\ I_{2,1}^{00} & I_{2,1}^{01} & I_{2,2}^{00} & I_{2,2}^{01} & I_{2,3}^{00} & I_{2,3}^{01} \\ I_{2,1}^{10} & I_{2,1}^{11} & I_{2,2}^{10} & I_{2,1}^{11} & I_{2,3}^{10} & I_{2,3}^{11} \\ I_{3,1}^{00} & I_{3,1}^{01} & I_{3,2}^{00} & I_{3,2}^{01} & I_{3,3}^{00} & I_{3,3}^{01} \\ I_{3,1}^{10} & I_{3,1}^{11} & I_{3,2}^{10} & I_{3,2}^{11} & I_{3,3}^{10} & I_{3,3}^{11} \end{bmatrix}$$

