Cours 2 :
Restauration d’images

Images et filtres
ELEC3

Lionel Fillatre
2018-2019

Introduction

- Les images subissent des dégradations et des bruits
- La restauration des images consistent à atténuer ces dégradations
- But : augmenter la qualité visuelle ou faciliter l'analyse des images

Avant

Après

Sommaire

- Principe de la restauration
- Exemple de modélisation
- Filtrage inverse

PRINCIPE DE LA RESTAURATION

Principe

- Restaurer une image consiste à essayer de compenser les dégradations subies par cette image.

- 3 types majeurs de dégradations :
  - bruit additif : $f(x, y) = i(x, y) + b(x, y)$
  - bruit multiplicatif : $f(x, y) = i(x, y) \times b(x, y)$
  - Convolution : $f(x, y) = i(x, y) * b(x, y)$

Type de dégradations

- En pratique, de nombreuses causes possibles de dégradations liées :
  - au milieu de mesure (atmosphère)
  - au système optique (diffraction, défocalisation)
  - au capteur (mouvement, diffusion, bruit)
  - à la scène (sujet en mouvement)
  - à la conversion (quantification, échantillonnage)
  - à la transmission (compression)
  - à la conservation (vieux films)

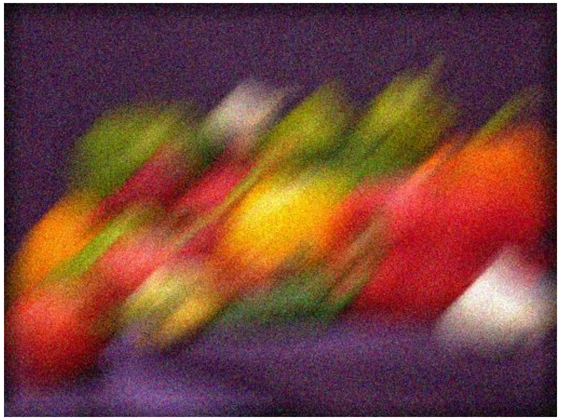
Exemple: flou de bougé

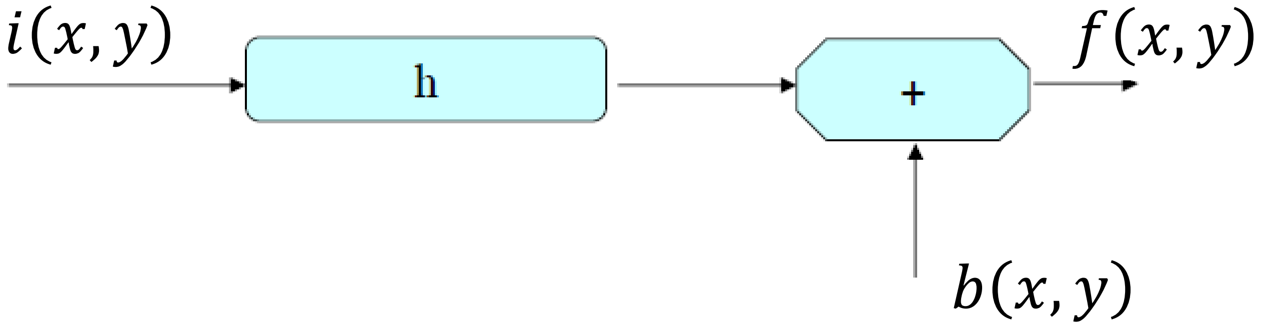
Exemple : bruit d'acquisition

Bruit uniforme (gaussien)

Bruit aléatoire (impulsionnelle)

Modèle de dégradation



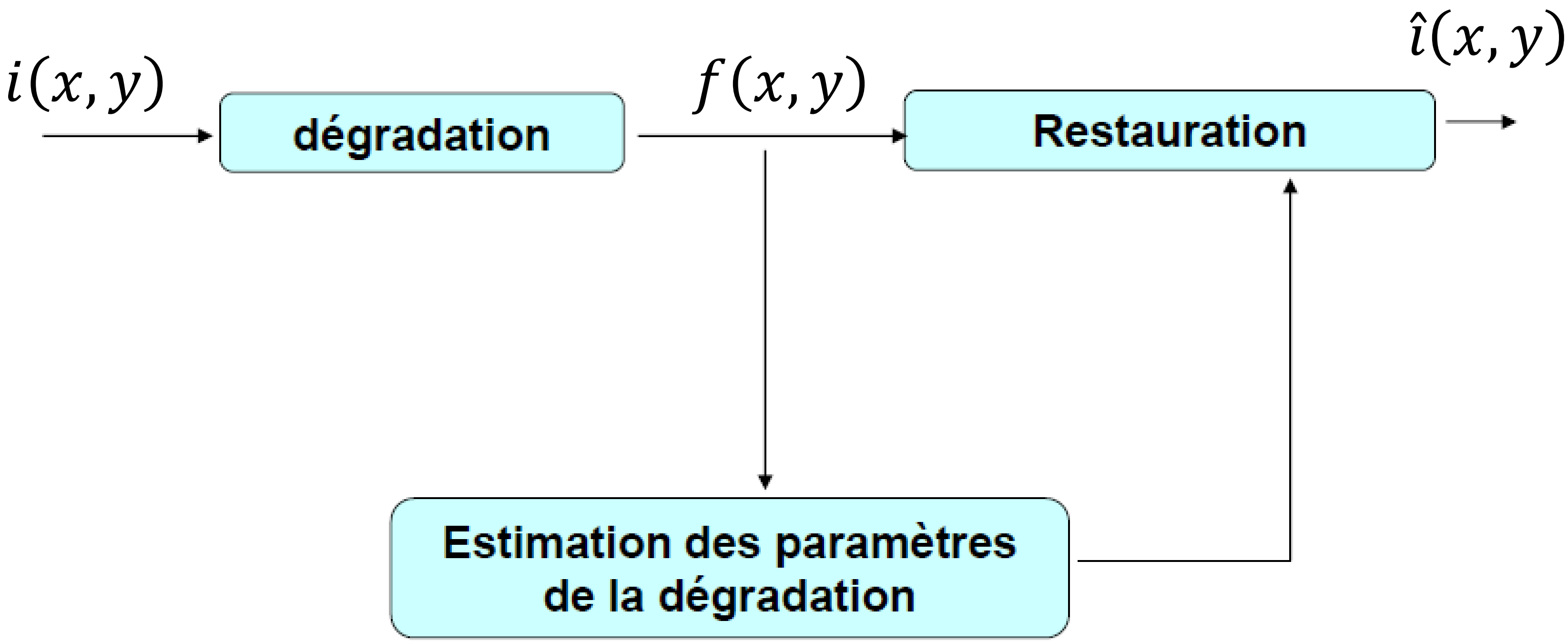


Objectif

- Calculer à partir de $f(x, y)$ une image $\hat{\imath}(x, y)$ aussi proche que possible de l'image originale $i(x, y)$.

- On a besoin donc de connaître :
  - Le filtre h,
  - La nature du bruit.

Principe de la restauration



Transformation de Fourier

- Le filtre de dégradation est symétrique par rapport à l'origine :

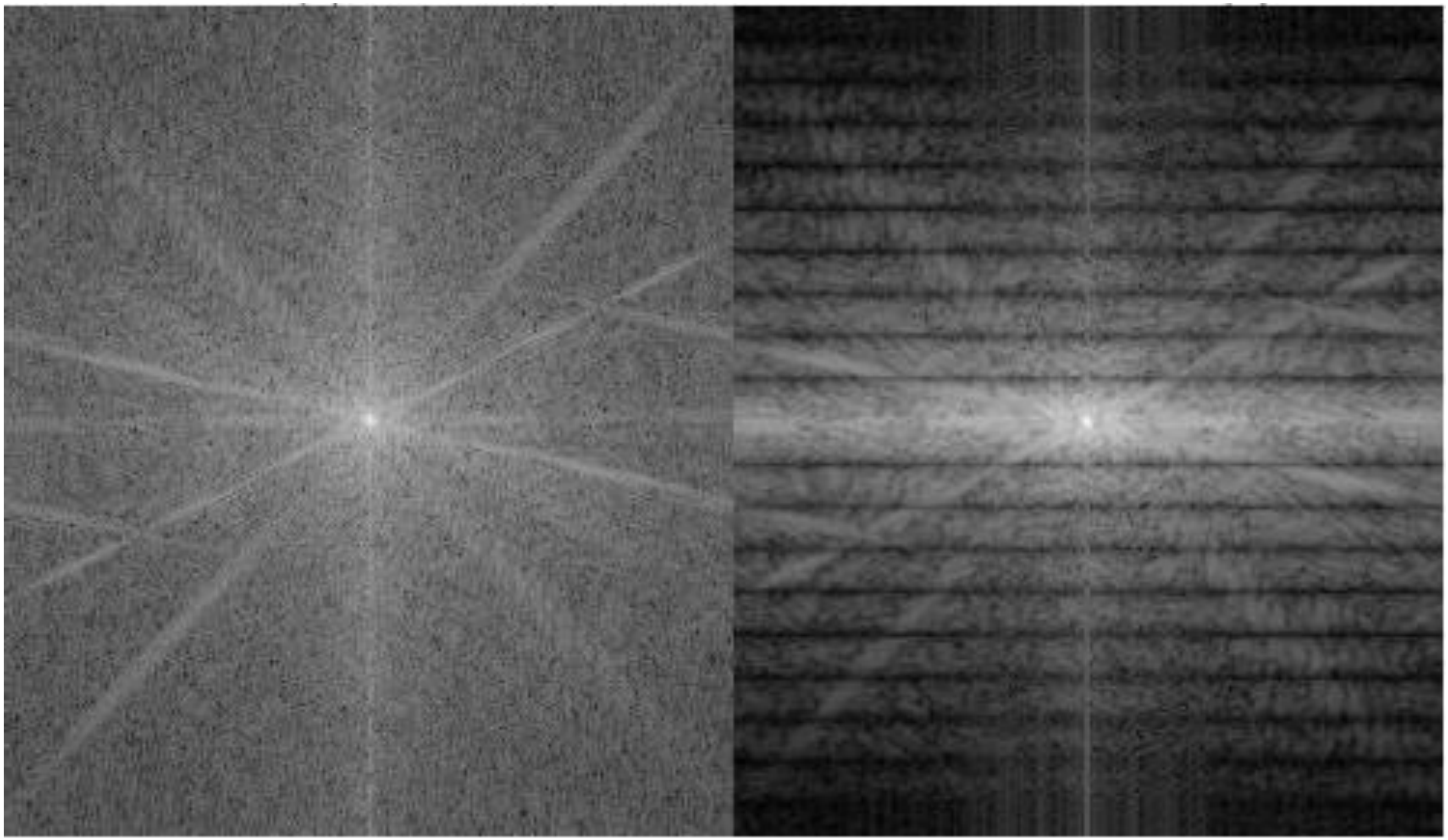
$$f(x,y) = \sum_{x\prime,y\prime} h(x',y')i(x+x',y+y') + b(x,y)$$

$$f(x,y) = h(x,y) * i(x,y) + b(x,y)$$

- En prenant la transformée de Fourier :

$$F(u,v) = H(u,v)I(u,b) \ + B(u,v)$$

Exemple : flou de bougé (domaine spectral)

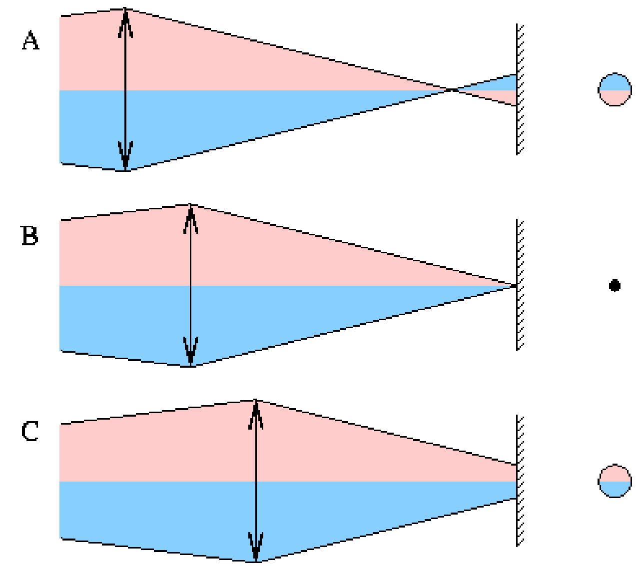


EXEMPLE DE MODÉLISATION

Paramètres de la dégradation : défocalisation

- Défocalisation

  Chaque point de la scène donne sur l'image une tache en forme de disque, cette tache étant d'autant plus grande que la défocalisation est importante.



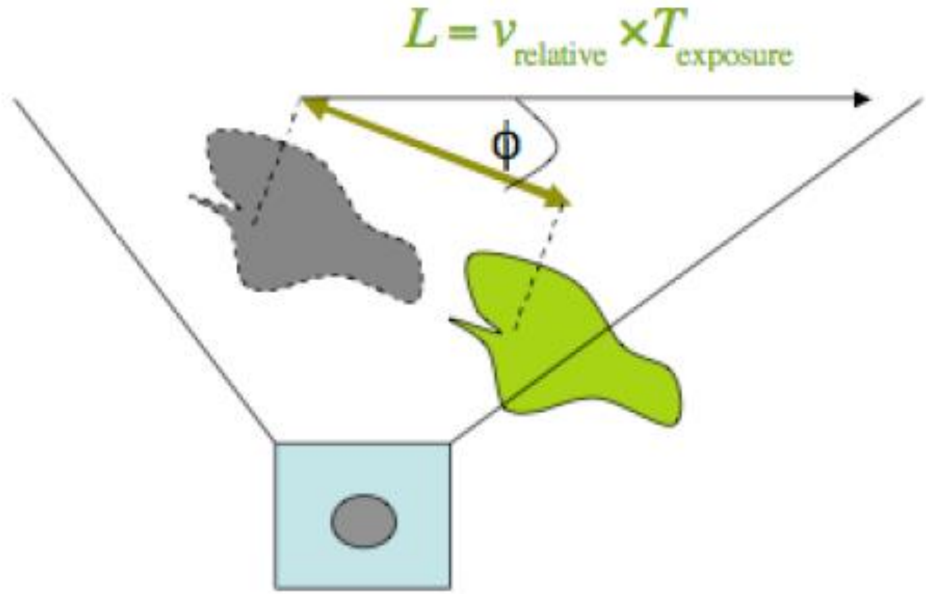
- On a alors :

$$h(x,y) = \begin{cases} \dfrac{1}{\pi R^2} & \text{si } x^2 + y^2 \leq R^2 \\ 0 & \text{si } x^2 + y^2 > R^2 \end{cases}$$

Paramètres de la dégradation : bougé

- Flou de bougé

  Il se modélise par un filtre linéaire dont la réponse impulsionnelle a la forme d'un segment.



- On a alors

$$h(x,y) = h(r,\theta) = \begin{cases} \frac{1}{L} \text{ si } \; 0 \leq r = \sqrt{x^2+y^2} \leq L \text{ et } \theta = \phi \\ 0 \;\; \text{sinon} \end{cases}$$

FILTRAGE INVERSE

Restauration par filtrage inverse

- On filtre l'image dégradée par un filtre $g(x, y)$ qui est l'inverse de $h(x, y)$.
- On passe dans le domaine des fréquences, en utilisant la transformée de Fourier.

En fréquentiel, on aura donc : $G(u,v) = \dfrac{1}{H(u,v)}$

- Pour restaurer l'image, on calcule le spectre de l'image restaurée :

$$\hat{I}(u,v) = G(u,v)F(u,v)$$

Ceci consiste à appliquer le filtre inverse dans la domaine des fréquences.

- Enfin, une transformée de Fourier inverse nous donne l'image restaurée $\hat{\imath}(x, y)$.

Sensibilité au bruit

- Afin de mieux comprendre le principe et les limites de cette méthode, nous allons examiner l'impact du bruit :

$$F(u,v) = H(u,v)I(u,v) + B(u,v)$$

$$\hat{I}(u,v) = G(u,v)F(u,v)$$

$$\hat{I}(u,v) = G(u,v)H(u,v)I(u,v) + G(u,v)B(u,v)$$

- Puisque $G(u,v)H(u,v) = 1$, on obtient

$$\hat{I}(u,v) = I(u,v) + G(u,v)B(u,v)$$

- Si le bruit était nul, on retrouverait exactement l'image originale.

Filtrage pseudo-inverse

- Pour un bruit non nul, ce qui sera toujours le cas en pratique, un problème se pose lorsque $H(u,v)$ devient très faible, car on a alors une forte valeur de $G(u,v)$, ce qui entraîne une forte amplification du bruit dans

$$\hat{I}(u,v) = I(u,v) + G(u,v)B(u,v)$$

- Solution : borner les valeurs que peut prendre $G(u,v)$:

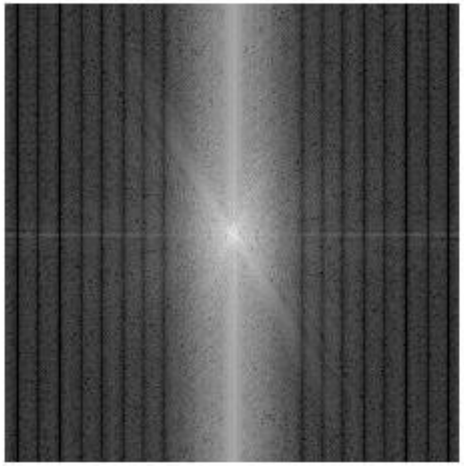
  Si $G(u,v) > S$ alors $G(u,v) = S$

  Si $G(u,v) < -S$ alors $G(u,v) = -S$

  où S est un seuil positif.

Exemple

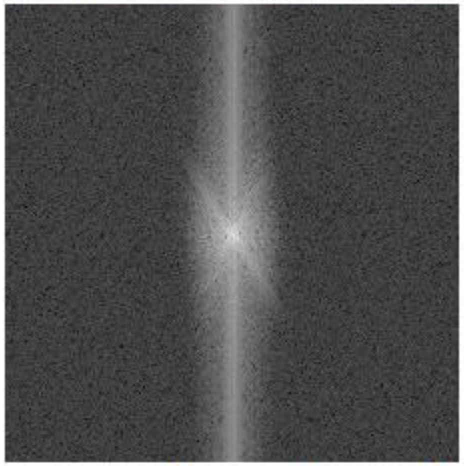
image avec un flou de bougé horizontal (float)



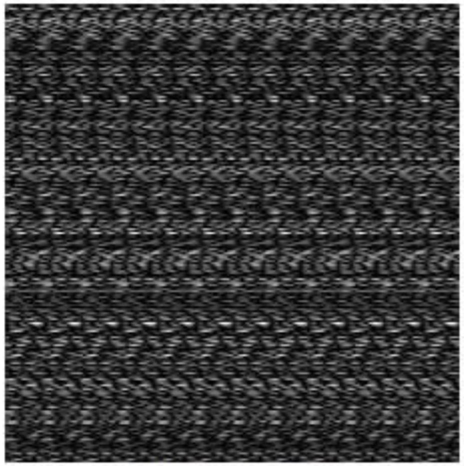
spectre de l'image flou : la distortion convolutive est visible

filtrage inverse

image avec un flou de bougé horizontal (byte)



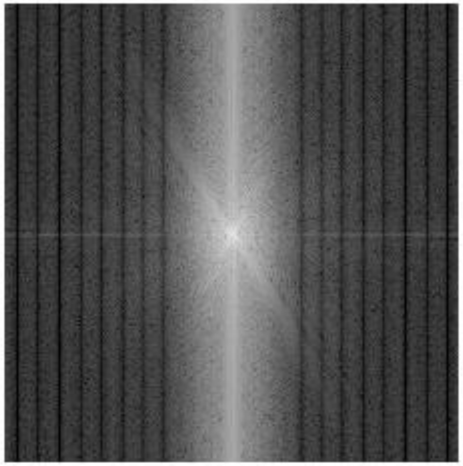
spectre de l'image flou avec le bruit de quantification



filtrage inverse

Exemple

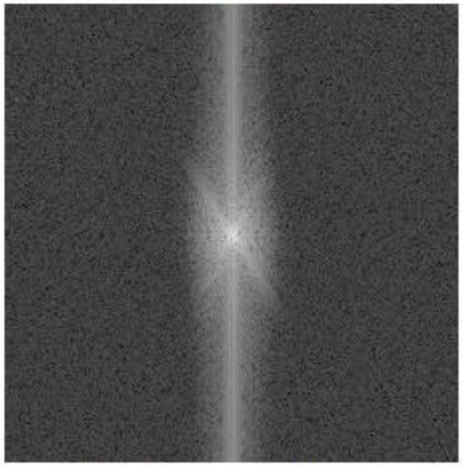
image avec un flou de bougé horizontal (float)



spectre de l'image flou : la distortion convolutive est visible

filtrage inverse

image avec un flou de bougé horizontal (byte)



spectre de l'image flou avec le bruit de quantification

filtrage pseudo-inverse

Filtrage de Wiener

- Le raisonnement qui vient d'être mené peut être rendu plus rigoureux : on aboutit à la notion de filtre de *Wiener*.

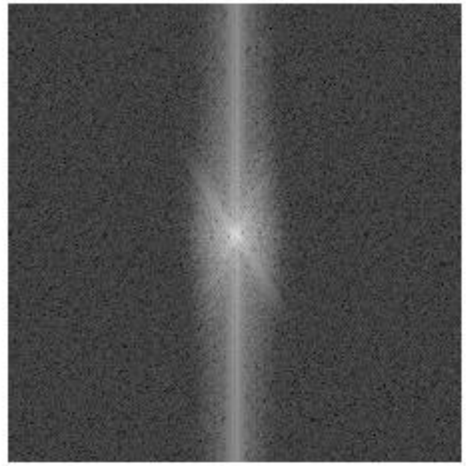
- On va déterminer le filtre $G(u,v)$ qui minimise l'erreur quadratique e_{QM} moyenne entre l'image idéale et l'image restaurée :

$$e_{QM}(u,v) = \boldsymbol{E}\left[\left(\hat{I}(u,v) - I(u,v)\right)^2\right]$$

Le filtre G est : $$G(u,v) = \frac{H^*(u,v)}{\left|H(u,v)\right|^2 + \rho}, \quad \rho = \frac{\left|B(u,v)\right|^2}{\left|I(u,v)\right|^2}$$

Exemple

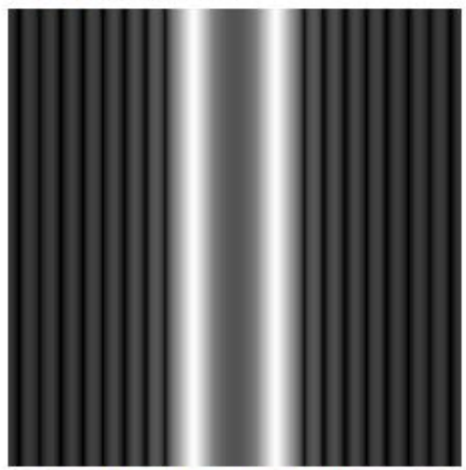
image avec un flou de bougé horizontal (byte)



spectre de l'image flou avec bruit de quantification

filtrage pseudo-inverse

image avec un flou de bougé horizontal (byte)



spectre du filtre de Wiener

filtrage de Wiener

CONCLUSION

Conclusion

- Une problématique très largement répandue
- Un besoin évident chez les utilisateurs
- De très nombreuses applications (satellites, photographie computationnelle, etc.)
- De nombreuses solutions…