



DIRO  
IFT 6150

## EXAMEN THÉORIQUE

*Max Mignotte*

DIRO, Département d'Informatique et de Recherche Opérationnelle  
[http: //www.iro.umontreal.ca/~mignotte/ift6150](http://www.iro.umontreal.ca/~mignotte/ift6150)  
*e-mail: mignotte@iro.umontreal.ca*

**Date:** 11/12/2024

I .....	Série de Fourier & Transformée de Fourier (31 pts)
II .....	Opérations Ponctuelles (33 pts)
III .....	Spectres et Filtrage (27 pts)
IV .....	Descripteurs, Classification & Segmentation (15 pts)
V .....	Questions relatives aux TPs (8 pts)
Total .....	114 points

TOUS DOCUMENTS, CALCULATRICES ET CALCULATEURS PERSONNELS AUTORISÉS

---

## I. Série de Fourier & Transformée de Fourier (31 pts)

### 1. TRANSFORMÉE DE FOURIER

Soit  $f(x)$ , une fonction (ou un signal spatial) et  $F(\nu)$  sa TF ou son spectre, *i.e.*;  $f(x) \rightleftharpoons F(\nu)$  où  $\mathcal{F}[f(x)] = F(\nu)$ .

- (a) Soit la fonction fonction triangle de largeur ou base  $2T$  et de hauteur  $(1/T)$  (et donc toujours d'aire unité) que l'on peut écrire:

$$\Lambda_T(x) = \frac{1}{T} \Lambda\left(\frac{x}{T}\right)$$

Pour rappel, avec cette notation, la fonction triangle classique que vous avez vu en cours jusqu'à présent est donc simplement:  $\Lambda_{T=1}(x) = \Lambda(x)$ .

En vous servant de vos connaissances, montrer que la limite de cette fonction quand  $T \rightarrow 0$  est la distribution de Dirac (notée  $\delta(x)$ ) d'une part, (-a-) en restant dans le domaine spatial et d'autre part, (-b-) en vous projetant dans le domaine fréquentiel.

<10 pts>

- (b) En cours, on a vu que:

$$\sum_{n \in \mathbb{Z}} \delta(x - nT) = \text{III}_T(x)$$

était le peigne de Diracs, c-à-d; la distribution composée d'une succession de Dirac espacés de  $T$ . Montrer maintenant que:

$$\text{III}_T(x) = \frac{1}{T} \sum_{n \in \mathbb{Z}} e^{2\pi j \frac{n}{T} x}$$

À cette fin, servez-vous de la propriété:  $\text{III}_T(x) \stackrel{\mathcal{F}}{\rightleftharpoons} \frac{1}{T} \text{III}_{\frac{1}{T}}(\nu)$  et plus précisément; simplifier mathématiquement, en utilisant deux propriétés du Diracs, l'expression  $\mathcal{F}^{-1}\left[\frac{1}{T} \text{III}_{\frac{1}{T}}(\nu)\right]$ .

<9 pts>

- (c) En utilisant entre autre et possiblement la dérivée d'une fonction connue (et une propriété de la TF), trouver la TF de la fonction:  $f(x) = x \exp(-\pi x^2)$ .

<7 pts>

- (d) Indiquer à quoi est égal:  $\delta\left(\frac{x}{2}\right) \cdot f(x)$

<5 pts>

---

### Réponse

1.(a)

[-a-]

Restons tout d'abord dans le domaine spatial. et pour cela il y a plusieurs façons de le démontrer:

- Lorsque  $T \rightarrow 0$ , la fonction triangle de largeur ou base  $2T$  et de hauteur  $1/T$  ( $\Lambda_T$ ) conservera son aire unité (la surface d'un triangle étant  $\text{BASE} \times \text{HAUTEUR} / 2$ ) comme le Dirac et quand  $T \rightarrow 0$ , la fonction  $\Lambda_T$  sera définie en un seul point, c-à-d;  $x=0$ .

De plus, lorsque  $T \rightarrow 0$ , la section gauche (supérieure) du triangle tendra vers une droite avec une pente ou dérivée infinie en  $0^-$  puis une pente  $-\infty$  en  $0^+$  pour la section droite (supérieure) du triangle tout comme le Dirac. Le Dirac étant **définie en ce seul point  $x=0$  avec cette même discontinuité de dérivée à gauche et droite du point  $x=0$  (mais de signe contraire), et sa largeur étant infiniment petite**

et sa hauteur infiniment grande quand  $T \rightarrow 0$ , avec cette particularité d'être aussi quand  $T \rightarrow 0$  de surface unité, et d'être toujours pair, on a donc nécessairement:

$$\lim_{T \rightarrow 0} \Lambda_T(x) = \lim_{T \rightarrow 0} \frac{1}{T} \Lambda\left(\frac{x}{T}\right) = \delta(x)$$

• Une autre façon de le démontrer consiste à dire que l'on sait (cf. cours) que  $\Pi(x) * \Pi(x) = \Lambda(x)$  et donc (ou/et aussi) que plus généralement, on a aussi:  $\frac{1}{T}\Pi_T(x) * \frac{1}{T}\Pi_T(x) = \Lambda_T(x) = \frac{1}{T} \Lambda\left(\frac{x}{T}\right)$  (avec  $\frac{1}{T}\Pi_T(x)$ ; la porte de largeur  $T$  et de hauteur  $T^{-1}$  et  $\Lambda_T(x)$ ; la fonction triangle, de base ou largeur  $2T$ , et de hauteur  $T^{-1}$ ).

On sait aussi (cf. cours) que  $\lim_{T \rightarrow 0} \left\{ \frac{1}{T}\Pi_T(x) \right\} = \delta(x)$  ce qui démontre que nécessairement:

$$\lim_{T \rightarrow 0} \Lambda_T(x) = \delta(x) * \delta(x) = \delta(x)$$

<5 pts>

[-b-]

Projetons nous dans le domaine fréquentiel maintenant pour le démontrer. En utilisant le fait que  $\mathcal{F}[\Lambda(x)] = [\sin(\pi\nu)/(\pi\nu)]^2$  ainsi que la propriété de dilatation-contraction, on trouve:

$$\Lambda_T(x) = \frac{1}{T} \Lambda\left(\frac{x}{T}\right) \stackrel{\mathcal{F}}{\iff} \left(\frac{\sin(\pi\nu T)}{(\pi\nu T)}\right)^2$$

Or:  $\lim_{T \rightarrow 0} \frac{\sin(\pi\nu T)}{(\pi\nu T)} = 1$ , donc:  $\lim_{T \rightarrow 0} \left\{ \frac{\sin(\pi\nu T)}{(\pi\nu T)} \right\}^2 = 1$ , : donc:

$$\lim_{T \rightarrow 0} \Lambda_T(x) = \lim_{T \rightarrow 0} \frac{1}{T} \Lambda\left(\frac{x}{T}\right) \stackrel{\mathcal{F}}{\iff} 1 \stackrel{\mathcal{F}^{-1}}{\iff} \delta(x)$$

Donc:  $\lim_{T \rightarrow 0} \Lambda_T(x) = \delta(x)$

**NOTA** : On aurait pu arrêter le raisonnement à  $\lim_{T \rightarrow 0} \left\{ \frac{\sin(\pi\nu T)}{(\pi\nu T)} \right\}^2 \rightarrow 1$ , et dire simplement que la TF tend vers un pour toutes les fréquences, ce qui est la définition de la TF du Dirac  $\delta(x)$ .

<5 pts>

**1.(b)**

• **Méthode[1]** Avec  $x \iff \nu$ , la solution la plus simple (utilisant la propriété de linéarité de la TF et la connaissance d'une TF<sup>-1</sup>) consiste à écrire:

$$\mathbb{I}\mathbb{I}\mathbb{I}_T(x) = \mathcal{F}^{-1}\left[\frac{1}{T} \mathbb{I}\mathbb{I}\mathbb{I}_{\frac{1}{T}}(\nu)\right] = \frac{1}{T} \mathcal{F}^{-1}\left[\sum_{n \in \mathbb{Z}} \delta(\nu - n/T)\right] = \frac{1}{T} \sum_{n \in \mathbb{Z}} \mathcal{F}^{-1}[\delta(\nu - n/T)] = \frac{1}{T} \sum_{n \in \mathbb{Z}} e^{2\pi j \frac{n}{T} x}$$

• **Méthode[2]** Une autre possibilité consiste à simplifier tout d'abord un peu mathématiquement:  $\mathcal{F}^{-1}\left[\frac{1}{T} \mathbb{I}\mathbb{I}\mathbb{I}_{\frac{1}{T}}(\nu)\right]$ :

$$\mathcal{F}^{-1}\left[\frac{1}{T} \mathbb{I}\mathbb{I}\mathbb{I}_{\frac{1}{T}}(\nu)\right] = \int_{-\infty}^{+\infty} \frac{1}{T} \sum_{n \in \mathbb{Z}} \delta\left(\nu - \frac{n}{T}\right) e^{+2\pi j \nu x} d\nu = \frac{1}{T} \sum_{n \in \mathbb{Z}} \int_{-\infty}^{+\infty} \delta\left(\nu - \frac{n}{T}\right) e^{2\pi j \nu x} d\nu$$

Maintenant, utilisons les deux propriétés du Dirac, à savoir le résultat du produit d'un Dirac avec une fonction (1ière ligne), et son intégrale unité (2ième ligne):

$$\begin{aligned} \mathcal{F}^{-1}\left[\frac{1}{T} \mathbb{I}\mathbb{I}\mathbb{I}_{\frac{1}{T}}(\nu)\right] &= \frac{1}{T} \sum_{n \in \mathbb{Z}} \int_{-\infty}^{+\infty} \delta\left(\nu - \frac{n}{T}\right) e^{2\pi j \nu x} d\nu = \frac{1}{T} \sum_{n \in \mathbb{Z}} \int_{-\infty}^{+\infty} \delta\left(\nu - \frac{n}{T}\right) e^{2\pi j \frac{n}{T} x} d\nu \\ &= \frac{1}{T} \sum_{n \in \mathbb{Z}} e^{2\pi j \frac{n}{T} x} \underbrace{\int_{-\infty}^{+\infty} \delta\left(\nu - \frac{n}{T}\right) d\nu}_1 = \frac{1}{T} \sum_{n \in \mathbb{Z}} e^{2\pi j \frac{n}{T} x} \end{aligned}$$

<9 pts> Or,  $\mathcal{F}^{-1}\left[\frac{1}{T} \mathbb{III}_{\frac{1}{T}}(\nu)\right] = \mathbb{III}_T(x)$  donc:  $\mathbb{III}_T(x) = \frac{1}{T} \sum_{n \in \mathbb{Z}} e^{2\pi j \frac{n}{T} x}$

**1.(c)**

On va donc utiliser la connaissance de la TF de  $\exp(-\pi x^2)$ , de la propriété de la dérivé et celle de la contraction dilatation soit:

$$\exp(-\pi x^2) \stackrel{\mathcal{F}}{\rightleftharpoons} \exp(-\pi \nu^2)$$

$$f'(x) \stackrel{\mathcal{F}}{\rightleftharpoons} 2\pi j \nu \cdot F(\nu) \quad \triangleright \text{donc: } -2\pi x \exp(-\pi x^2) \stackrel{\mathcal{F}}{\rightleftharpoons} (2\pi j \nu) \exp(-\pi \nu^2)$$

Donc;

$$\mathcal{F}[x \exp(-\pi x^2)] = -j \nu \cdot \exp(-\pi \nu^2)$$

<7 pts>

**NOTA** : On pouvait s'attendre un peu à cette réponse. En effet, puisque  $f(x)$  est *impair*, sa TF est forcément *imaginaires pure*, ce qui est bien le cas ici.

**1.(d)**

$$\delta\left(\frac{x}{2}\right) \stackrel{\mathcal{F}}{\rightleftharpoons} 2 \cdot \mathbf{1}(2\nu) = 2 \stackrel{\mathcal{F}^{-1}}{\rightleftharpoons} 2 \delta(x) \quad \text{donc: } \delta\left(\frac{x}{2}\right) \cdot f(x) = 2 \delta(x) \cdot f(x) = 2 f(0) \delta(x)$$

Ps: La fonction  $\mathbf{1}(\nu)$  désigne la fonction constante 1 de la variable  $\nu$ .

<5 pts>

## II. Opération Ponctuelle (33 pts)

### 1. INTERPRÉTATION D'ALGORITHME

Soit  $I(i, j)$  une image naturelle de départ de taille (length,width) en 256 niveaux de gris (dans laquelle  $I(i, j)$  correspond au niveau du gris du pixel de coordonnées ligne=  $i$  et colonne=  $j$ ). On réalise l'opération ponctuelle suivante (en C) (qui détruit l'image de départ  $I(i, j)$  [pour les besoins de l'algorithme] mais) qui permet de trouver l'image  $O(i, j)$  (initialement remplie de 0 et de même taille que  $I$ ):

```
. float oldpix, newpix, erpix;
. for(i=1;i<(length-1);i++) for(j=1;j<(width-1);j++)
. {
.     oldpix=I[i][j];
.     if (oldpix>128) newpix=255; else newpix=0.0;
.     erpix=oldpix-newpix;
.     O[i][j]=newpix;
.     \\ Version simplifiée de l'Algorithme on remplace les
.     \\ 4 dernières lignes par: I[i][j+1]=I[i][j+1]+erpix;
.     I[i][j+1]=I[i][j+1]+erpix*(7.0/16.0);
.     I[i+1][j-1]=I[i+1][j-1]+erpix*(3.0/16.0);
.     I[i+1][j]=I[i+1][j]+erpix*(5.0/16.0);
.     I[i+1][j+1]=I[i+1][j+1]+erpix*(1.0/16.0);
. }
```

On vous demande, en tant qu'expert en traitement d'images d'expliquer avec le plus de détails intéressants possibles ce que fait algorithmiquement cette procédure sur une image (pour cela imaginer que vous fassiez du *retro-engineering* de logiciels pour une compagnie [ou essayer de comprendre ce qu'a

fait algorithmiquement un ancien employé qui n'est plus dans l'entreprise] et que votre chef de projet (qui n'est pas un expert en traitement d'images comme vous) vous demande d'expliquer concrètement, brièvement et clairement, ce que font ces quelques lignes de code en matière de traitement d'images. Expliquez aussi ce que la version simplifiée de l'algorithme apportera comme *artefacts* ou *simplifications*.

<14 pts>

## 2. OPÉRATION PONCTUELLE

- (a) Quelle(s) opération(s) élémentaire(s) sur l'image du photograph permettrait d'avoir une meilleure visibilité des détails du gant et de la veste du photograph ?

<7 pts>



- (b) On fait l'opération ponctuelle suivante. On fait la différence (pixel par pixel et en valeur absolue) entre l'image du photograph et l'image du photograph qui a été comprimée par une compression JPEG (l'image comprimée JPEG est ensuite reconvertit en pgm pour être lu dans un programme. À la fin de cette différence en valeur absolue, on recale l'image de différence entre 0 et 255).

Décrire ce que l'on va obtenir.

<5 pts>

- (c) Donner le pseudo-code abrégé ou les différentes étapes permettant de changer une image avec une moyenne de  $\mu_A$  et de variance  $\sigma_A^2$  en une image de moyenne  $2\mu_A$  et de variance  $\sigma_A^2/2$  (les étapes faciles pourront être fortement résumées du style 1- ...on calcule d'abord  $\mu_A$  et ...

<7 pts>

---

### Réponse

1.(a)

Très simple! Prenons d'abord la version simplifiée de l'algorithme.

L'image obtenue O sera donc binaire avec des points blancs (*i.e.*, de niveaux de gris 255) ou noirs (niveaux de gris 0). Pour une image en niveaux de gris que l'on va donc parcourir dans le sens (habituel) lexicographique; lorsque ce balayage traversera une zone uniforme blanche, l'algorithme ne changera rien; même chose pour une zone uniforme noire. Lorsque le balayage sera sur une zone de gris moyen (*i.e.*, de niveaux de gris 128), l'algorithme la remplacera par un pixel blanc suivi d'un pixel noir etc. Lorsque le balayage balaiera une région de l'image de niveaux de gris moyen 170, l'algorithme remplacera la ligne de niveaux de gris 

170	170	170	170	170	170
-----	-----	-----	-----	-----	-----

... par 

255	0	255	255	0	255
-----	---	-----	-----	---	-----

... gardant ainsi la moyenne locale (linéairement dans cet exemple) de la région balayée mais **en utilisant seulement des points noirs ou des points blancs et ainsi en créant une illusion de niveaux de gris** en conservant la valeur moyenne de l'image (puisque la somme des coefficients est égale à un [conservation de l'énergie]) (Nota: C'est le principe d'un *tramage*).

En utilisant la version simplifiée de l'algorithme (qui propage l'erreur uniquement vers le pixel de droite donc horizontalement et accumulant ainsi excessivement cette erreur [de diffusion] dans cette direction) , on aura un tramage, simulant les niveaux de gris (continu) d'une région en utilisant seulement des points blancs et noirs, avec plus souvent des *artefacts* en ligne ou des *artefacts* de toutes sortes (faux contours, *pattern* psychédélique, et des discontinuités visibles, sous forme d'escalier, ou en ligne particulièrement visible sur une rampe de niveau de gris croissant [cf. Fig. ci-dessous]). La version améliorée (4 dernières lignes) passe

d'une opération ponctuelle à une opération locale à un voisinage et est plus efficace et intéressante car elle supprime ces *artefacts* de toutes sortes.

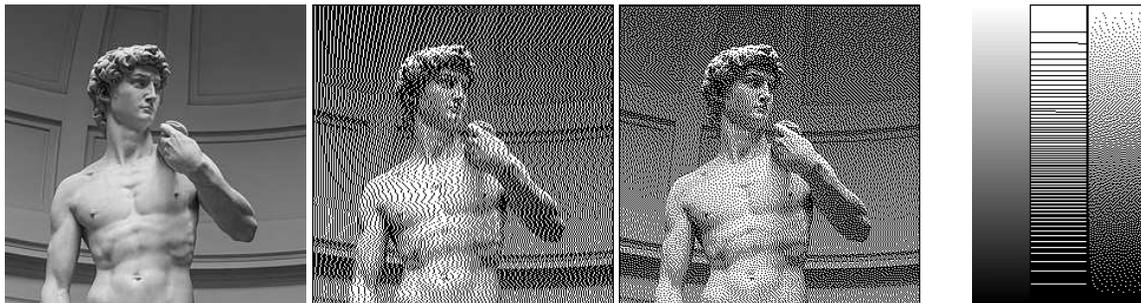
<14 pts>

NOTA -1- : Plus techniquement, l'algorithme réalise ce tramage que l'on appelle *halftoning* ou *dithering*, en utilisant le principe de diffusion d'erreur, ce qui signifie qu'il pousse (ajoute) l'erreur de quantification résiduelle d'un pixel sur ses pixels voisins, pour être traitée ultérieurement. Il répartit celle-ci selon cette répartition:

$$\left( \begin{array}{cccc} & & * & \frac{7}{16} \\ \dots & \frac{3}{16} & \frac{5}{16} & \frac{1}{16} & \dots \end{array} \right)$$

Le pixel indiqué par une étoile (\*) est le pixel en cours de balayage, et les pixels vides sont les pixels précédemment balayés. L'algorithme balaie l'image lexicographiquement, en quantifiant les valeurs des pixels un par un. À chaque fois, l'erreur de quantification est transférée aux pixels voisins, sans affecter les pixels qui ont déjà été quantifiés. Par conséquent, si un certain nombre de pixels ont été arrondis vers le bas, il est plus probable que le pixel suivant soit arrondi vers le haut, de sorte qu'en moyenne, l'erreur de quantification est proche de zéro. L'algorithme a été créé par Robert W. Floyd et Louis Steinberg en 1976. Cette technique a permis d'imprimer pour la première fois une image continue en dégradé de niveaux de gris en utilisant seulement des petits points (blancs ou noirs) et a été élaborée pour la première fois en imprimerie au milieu du XIXe siècle.

NOTA -2- : On obtient, pour, respectivement l'image originale (statue et pour une trame de niveaux de gris), la version simplifiée puis originale (*i.e.*, améliorée) de cet algorithme (de *dithering* de Floyd & Steinberg):



2.(a)

Une simple fonction LOGarithme,

...ou toute fonction ponctuelle  $g(x,y) = \mathbf{T}[f(x,y)]$  d'abscisse; les niveaux de gris de l'image avant transformation ( $f(x,y)$ ) et pour ordonnée; les niveaux de gris ( $g(x,y)$ ) de l'image traitée résultante (avec  $(x,y)$ ; les lignes et colonnes de cette image), du même type, c-a-d, avec une forte pente pour les faibles niveaux de gris puis une pente qui s'adoucit, de plus en plus, pour les niveaux de gris proches du blanc...

et un RECALage (entre 0 et 255) des niveaux de gris de cette image permettrait d'étirer l'histogramme dans les zones de réglons sombres et ainsi permettre d'avoir une meilleure visibilité des détails dans celles-ci.

<7 pts>

NOTA -1- : On obtient, pour l'image originale et l'image transformée:



NOTA -2- : Une égalisation d'histogramme aurait été moins bonne que la stratégie décrite précédemment (mais aurait donnée les deux tiers des points de cette question)

**2.(b)**

On va imager le bruit de compression JPEG en valeur absolue d'amplitude, *i.e.*, les *artefacts* spatiaux introduit par la compression JPEG.

La compression JPEG va agir comme une sorte de filtre passe-bas qui va éliminer en fait les coefficients hautes fréquences de la DCT de chaque petits blocs de  $8 \times 8$  pixels de l'image originale. On a vu en cours que la différence d'une image avec une image passe-bas est équivalent à un filtre passe-haut. La différence de l'image originale moins l'image compressée JPEG (en valeur absolue et recalé ensuite) permettra donc de mettre en évidence toutes les erreurs (en valeur absolue) de la compression JPEG qui sont essentiellement les hautes fréquences perdues de chaque petits blocs  $8 \times 8$  de l'image originale. On verra aussi les artefacts en blocs et les artefacts hautes fréquences caractéristiques créées par ce type de compression (essentiellement sur les contours et les variations rapides de l'image).

<5 pts>

NOTA : On obtient, pour l'image originale photograph et l'image différence: (et une version agrandie). On peut voir les artefacts en bloc  $8 \times 8$  de la DCT et les artefacts (erreurs) hautes fréquences caractéristiques créés par ce type de compression (les erreurs de compression sont d'autant plus importantes et visibles [de moyenne de niveaux de gris d'autant plus claire] que le bloc  $8 \times 8$ , extrait de l'image, contient des variations de niveaux de gris rapides.



**2.(c)**

1. On calcule  $\mu_A$  et  $\sigma_A^2$
2. On recentre l'image sur une moyenne nulle en retirant, à tous les niveaux de gris, la valeur  $\mu_A$ .
3. On divise l'amplitude des niveaux de gris de l'image par  $\sqrt{2}$ . L'image est ainsi toujours centrée sur une moyenne nulle mais a une variance de  $\sigma_A^2/2$ .
4. On ajoute à l'image précédente, à tous les niveaux de gris, la valeur  $2\mu_A$  pour obtenir finalement une image centrée sur une moyenne  $2\mu_A$ .

La seule difficulté de cette opération est l'étape 3. Il faut se souvenir qu'une variance est le carré de son écart type qui lui est homogène à une longueur ou une amplitude de signal (ou de niveaux de gris). Diviser l'amplitude des niveaux des gris d'une image par  $\sqrt{2}$  divisera ainsi la variance de l'image par deux.

<7 pts>

---

### III. Spectres & Filtrage (27 pts)

#### 1. Filtrage d'interférences

Voici une image bruitée par des interférences:



- (a) Expliquer de la façon la plus claire possible à quoi ressemblera la dégradation du spectre de cette image de taille  $256 \times 256$ . Vous devez être capable de me dire exactement la nature de cette dégradation et où elle se trouve au **pixel près** sur le spectre. En déduire une façon simple de restaurer cette image par filtrage fréquentiel et me dire quels sont les paramètres de ce filtre fréquentiel au **pixel près** dans le domaine fréquentiel.

<10 pts>

- (b) Qu'est-ce qui serait le plus difficile à restaurer; une image dégradée avec une interférence périodique additive qui s'ajoute sur l'image qui serait de haute fréquence ou de basse fréquence ? Expliquez en justifiant clairement pourquoi ?

<5 pts>

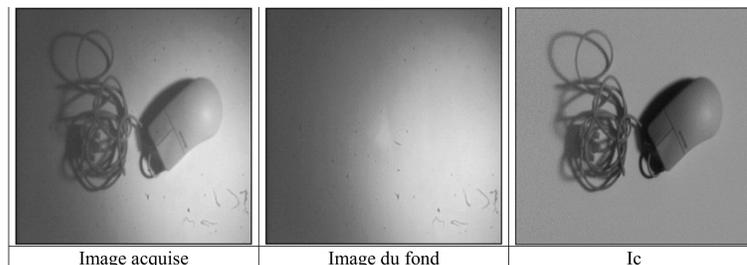
- (c) Supposons que le spectre de l'image sans interférence est modélisé par un ensemble de coefficient spectraux non nulles (à un bruit numérique près) qui s'étend sur une tache à peu près circulaire de rayon (approximativement) de 10 pixels (en fréquence ligne et colonne). À partir de quelle fréquence numérique (de la dégradation additive périodique), la restauration de l'image commencerait à être problématique et difficile. Justifiez bien votre réponse et en déduire pour être précis, le nombre minimal ou maximal de périodes de cette dégradation qui commencerait à être problématique et difficile.

**Ps:** je veux que vous puissiez finalement répondre finalement (à la fin de votre raisonnement) à l'assertion suivante: "*...le problème existe donc pour des dégradation engendrant donc plus/moins (?) de ? périodes ou franges sur l'image.*"

<7 pts>.

#### 2. Filtrage Adaptatif

Soit l'image acquise (de gauche)



- (a) Expliquer en premier lieu le type de dégradation de cette image. Expliquer ensuite comment faire pour obtenir l'image restaurée et l'image de fond.  
<5 pts>

### Réponse

#### 1.(a)

L'image de taille  $N \times N = 256 \times 256$  est dégradée par des interférences périodiques horizontales additives (presque sinusoïdales). On peut compter 25 périodes sur cette image, soit des périodes de largeur  $T \approx 10$  pixels qui correspondront à des dégradations de fréquences numériques  $F_d = 1/10 = 0.1$  dans le domaine fréquentiel.

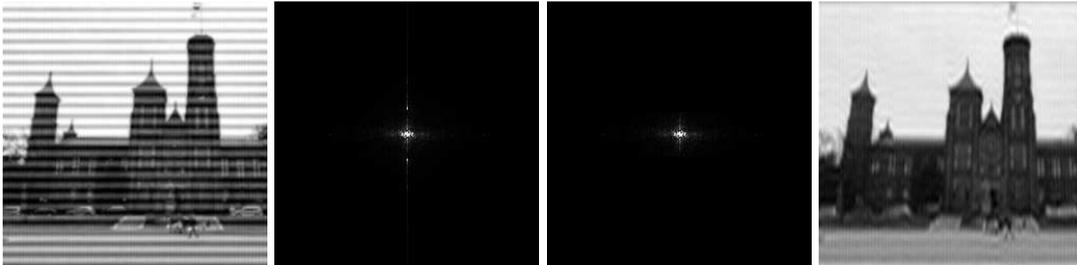
La plus petite fréquence détectable du spectre de cette image, de taille  $256 \times 256$ , est  $\Delta\nu = 1/T = 1/(N\Delta T) = 1/N = 1/256$  (cf. cours) en fréquence numérique ( $\Delta T = 1$ ). La fréquence numérique  $F_d \approx 0.1$  sera donc associée au 25-ième pixel ( $\pm 1$  pixel) du spectre de l'image à partir de la fréquence 0 (qui est au centre de l'image et qui en fait l'origine du spectre associée à la fréquence continue [fréquence numérique 0] <correspondant à la moyenne de l'image>).

Cette dégradation se verra sous la forme de deux points lumineux symétriques par rapport à l'origine du spectre, ou plus précisément; de deux petites taches lumineuses, de taille un à deux ou trois pixels de large (car la dégradation n'est pas tout à fait exactement sinusoïdale). Et ces deux petites taches se trouveront verticalement (perpendiculairement aux hautes fréquences créées par cette dégradation) de part et d'autre (fréquences positives et négatives) de l'axe vertical de ce spectre (recentré au centre de l'image), passant par l'origine.

Une façon simple de restaurer cette image par filtrage fréquentiel et de se débarrasser de ces deux petites taches lumineuses, par exemple, par un **filtre passe bande réjecteur de fréquence locale et centré sur la fréquence numérique (en ligne):  $0.1 \pm 1/256$  et  $-0.1 \pm 1/256$** .

<10 pts>

**NOTA** : On obtient, pour l'image originale, son spectre, le spectre filtré et l'image filtrée fréquentiellement:



#### 1.(b)

La dégradation additive faisant intervenir une interférence périodique de basse fréquence serait la plus difficile à enlever car celle-ci serait mélangée totalement au spectre de l'image non bruitée (lequel contient l'information structurelle essentielle et utile [contenu important] de l'image non bruitée).

Il est très difficile de retrouver les coefficients spectraux de l'image originale non bruitée lorsque ceux-ci sont altérés de valeurs (de bruit) aléatoires.

<5 pts>

#### 1.(c)

Puisque le spectre de l'image sans interférence est modélisé par un ensemble de coefficients spectraux non nulles (à un bruit numérique près) qui s'étend sur une tache à peu près circulaire de rayon 10 pixels (en

fréquence ligne et colonne), la fréquence la plus élevée de spectre de l'image (non bruitée par des interférence) est donc de fréquence numérique  $10 \times \Delta\nu = (10/256) \approx 0.039$ . Si la fréquence numérique de la dégradation additive qui s'ajouterait à cette image, était donc plus petite que cette fréquence numérique; il y aurait donc problème; car les coefficients spectraux de cette dégradation se mélangerait au spectre de l'image originale non bruitée (*et ce serait très difficile de les dé-mélanger !*). Le problème existe donc pour des fréquences numérique inférieures à 0.039, c-à-d, pour des périodes numériques supérieures à  $1/0.039 \approx 25.6$  pixels, c-à-d, pour des dégradation engendrant **strictement moins** de  $256/25.6 = 10$  périodes ou franges sur l'image.

<7 pts>

**NOTA** : Dans notre cas ou il y a 25 périodes sur l'images, le filtrage passe-bande se fera sans problème.

**2.(a)**

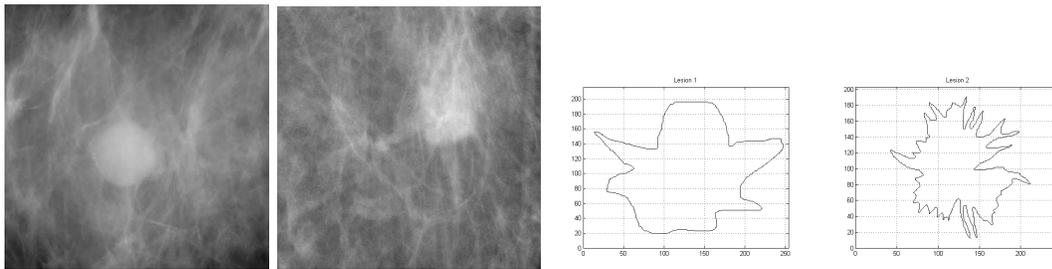
Il s'agit d'une dégradation multiplicative du type *défaut d'illumination* de très basse fréquence. Un Filtrage homomorphique de l'image permettrait de restaurer l'image. L'image de fond est ensuite simplement trouvé par l'opération ponctuelle: Image originale - Image restaurée, suivi d'un recalage des niveaux de gris (idéalement sur la moyenne de l'image non dégradée).

<5 pts>

#### IV. Descripteurs, Classification & Segmentation (15 pts)

##### 1. Descripteurs

Le cancer du sein est une tumeur maligne de la glande mammaire et ce type de tumeur est souvent repéré sur une radiographie du sein que l'on appelle mammographie. Les tumeurs bénignes sont généralement des masses de cellules rondes, lisse avec des frontières bien déterminées (Fig.1). En contre partie, les tumeurs malignes sont des masses de cellules avec des contours mal définis, irréguliers et des frontières présentant ce que les médecins appellent des spicules ou filaments s'étendant aux tissus voisins (Fig. 2 et 4.). Ainsi les contours de la tumeur montrées en Fig. 4 a plus de chance d'être associé à une tumeur maligne, car comportant de nombreux spicules comparées à celle de la Fig. 3. Une façon de réaliser un diagnostic automatique par traitement d'images est de segmenter cette tumeur et caractériser le mieux possible cette forme, ou plus précisément, caractériser les contours de chacune de ces tumeurs par un ou un jeu de paramètre qui caractérisera (ou résumera) au mieux cette propriété de spicularité et qui sera ou seront utilisés dans une procédure de classification réalisant ainsi un premier diagnostic automatique qui aidera le médecin dans son diagnostic final.



(a) Donner deux paramètres de formes, étudiés en cours qui peuvent permettre de caractériser et différencier une tumeur maligne, d'une tumeur bénigne.

<5 pts>

(b) Proposer un paramètre de forme, non étudié en classe qui pourrait être pertinent pour réaliser cette classification automatique.

<5 pts>

- (c) Indiquer quels pré-traitement sur l'image permettrait plus facilement de réaliser la segmentation de ces contours.  
<5 pts>

---

### Réponse

#### 1.(a)

Les paramètres de formes vus en classe qui permettrait de caractériser cette possible plus ou moins importante spicularité associée à chacune des tumeurs sont par exemple:

1. La compacité: plus il est proche de un, plus le contour de la masse est circulaire et celle ci est bénine. A l'inverse, un facteur de compacité proche de 0 indique une grande spicularité ou une forme peu compacte.
2. Les descripteurs de Fourier, en particulier les valeurs (d'amplitude ou d'énergie) associés aux coefficients hautes fréquences des descripteurs de Fourier sera plus importante si la tumeur est maligne donc très peu compacte.

<5 pts>

#### 1.(b)

On pourrait proposer par exemple une ou deux projections comme celle utilisée en tomographies. Le résultat obtenu pour une tumeur bénine, presque circulaire serait une courbe très lisse (somme des gradients sur cette courbe très faible) et pour une tumeur très spiculaire, une courbe présentant des forts gradients.

- On pourrait utiliser plusieurs projections comme celle utilisée en tomographies et calculées pour chacun des angles le nombre de fois que le rayon de projection intersecte les contours de la tumeur. La courbe obtenues (nombre d'intersections = fonction de l'angle de projection) aurait une variance importante dans le cas de formes spiculaires et une variance faible dans le cas d'une forme presque circulaire.

- On pourrait utiliser la vitesse de décroissance des coefficients des descripteurs de Fourier (rapide pour la tumeur bénine et plus lente pour la tumeur malignes).

- On pourrait utiliser aussi la rugosité ou l'écart-type des distances entre le centre de masse de la tumeur et les différents points de contour (pour la tumeur bénine: rugosité faible et pour celle maligne: rugosité plus importante).

<5 pts>

#### 1.(c)

Rehausseur de contour (un des trois vu en cours) suivi d'une égalisation d'histogramme (possiblement localement adaptative; *i.e.*, faite sur des petites fenêtres se chevauchant, centrée sur chaque pixel, et lesquelles seraient ensuite moyennées). Tout cela permettrait d'augmenter efficacement le contraste de l'image et l'information importante des contours de cette image.

<5 pts>

---

## V. Questions Relatives aux TPs (8 pts)

1. Dans le TP numéro 5 **SEGMENTATION MARKOVIENNE**.

Soit l'image ci-dessous représentant une page d'un article de revue. On peut voir sur cette image deux régions homogènes au sens de la texture; la première région dans laquelle il y a le texte ou les légendes des images et la seconde ou il y a des images en niveaux de gris. On supposera que le document est converti préalablement en niveaux de gris (initialement, il était en couleur).

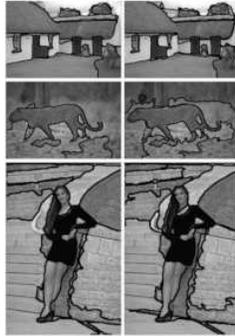


Fig. 3. Comparison of two segmentation results of our multiresolution fusion procedure (algorithm PPH [14], [15], [16], [17]) using respectively: left) a subgraphed and fixed number of connections ( $K$ ) regularly spaced and located within a square search window of size  $N_s = 30$  pixels, right) a fully connected graph computed on a search window two times larger and requiring a computational load increased by 100%.



Fig. 4. Segmentation image no. 385628 from Berkeley database. From top to bottom and left to right, segmentation map respectively obtained by: 1) our multiresolution optimization procedure ( $E = -3482960$ ), algorithm PPH ( $K_1=10, K_2=10, K_3=10, K_4=10$ ) ( $E = -3206127$ ), 2)  $S_N$  ( $N_{max}=1000$ ) ( $E = -3323704$ ), 3)  $S_N$  ( $N_{max}=1000$ ) ( $E = -3355732$ ), 4)  $S_N$  ( $N_{max}=1000$ ) ( $E = -3482960$ ).

with its four nearest neighbors and a fixed number of connections (88 in our application), regularly spaced between all other pixels located within a square search window of fixed size  $N_s = 30$  pixels centered around  $x$ . Fig. 3 shows comparison of segmentation results with a fully connected graph computed on a search window two times larger.

We decided to initialize the lower (or third upper) level of the pyramid with a sequence of 20 different random segmentations with  $K_1$  classes. The full resolution level is then initialized with the duplication (see Fig. 1) of the best segmentation result (i.e., the one associated to the lowest Gibbs energy  $E_k$ ) obtained after convergence of the ICM at this lower resolution level (see Fig. 2). We provide details of our optimization strategy in Algorithm 1.

Algo 1. Multiresolution minimization procedure (see also Fig. 2).

**Two-Step Multiresolution Minimization**

$\{S_k\}$  Set of  $L$  segmentations to be fused  
 $\{p_{ij}\}^{(l)}$  Pairwise probabilities for each pixel pair computed over  $\{S_k\}$  at resolution level  $l$

**I. Initialization Step**

- $\{P_k\}$  → Build  $L$  multiresolution Pyramids from  $\{S_k\}$
- $\{p_{ij}\}^{(3)}$  → Compute the pairwise probabilities from  $\{P_k\}$  at resolution level 3
- $\{p_{ij}\}^{(l)}$  → Compute the pairwise probabilities from  $\{S_k\}$  at full resolution ( $\{p_{ij}\}^{(l)} = \{p_{ij}\}$ )

**PIXEL LEVEL**

**Initialization** Random initialization of the upper level of the pyramidal structure with  $K_1$  classes

- ICM optimization on  $\mathcal{U}(\{p_{ij}\}^{(3)})$
- Duplication (cf. Fig. 1) to the full resolution
- ICM optimization on  $\mathcal{U}(\{p_{ij}\})$

**REGION LEVEL**

for each region at the finest level do  
 • ICM optimization on  $\mathcal{U}(\{p_{ij}\}, \beta, \tau) = \mathcal{U}(\{p_{ij}\}) + \beta|y|/\tau$

D. Comparison With a Monoresolution Stochastic Relaxation  
 In order to test the efficiency of our two-step multiresolution

On aimerait faire une segmentation en deux classes de cette image; soit la classe d'étiquette *noire* pour la première région (étiquette *texte* incluant les légendes des Figures et le texte d'un Tableau) et l'étiquette *blanche* pour la deuxième (étiquette *Figure*)

On aimerait aussi utiliser l'algorithme utilisé et programmé au dernier TP. Dans cette optique, où on désire changer le moins de chose possible (on pourra supprimer l'étape des K-moyenne) comment et/ou quelle partie et de quelle façon devrait-on changer cet algorithme pour réaliser cette opération de segmentation ?

< 8 pts >

**Réponse**

1.

Dans sa version initiale, le programme cherchait à trouver deux régions différentes dans l'image au sens des niveaux de gris. Dans la nouvelle version, la première région, correspondant au texte, est composée de niveaux de gris soit uniquement blancs ou soit uniquement noirs. Dans la deuxième région (correspondant aux images), les régions ont des niveaux de gris compris entre 0 et 255.

- Peut-être, le plus simple (qui impliquerait le moins de changement possible dans l'algorithme) serait de, d'une part, supprimer l'étape des K-moyenne et, d'autre part, de créer deux nouvelles vraisemblances différentes. La première donnerait une grande vraisemblance (ou probabilité  $P_{Y/X}(y/x = \text{texte})$ ) aux niveaux de gris qui sont soit blancs (ou très proche du blanc) ou soit (très proche du) noirs et la seconde donnerait une grande vraisemblance (ou probabilité  $P_{Y/X}(y/x = \text{image})$ ) aux niveaux de gris qui sont compris disons entre 20 et 235 (pour tenir compte du bruit de cette image). La seconde vraisemblance pourrait être une Gaussienne avec pour moyenne et variance:  $\mu_2 = 128, \sigma_2^2 = 500$  et la première serait soit une Gaussienne de moyenne 255 et de variance 500 si le niveaux de gris est supérieur à 128 ou soit une Gaussienne de moyenne 0 et de variance 200 si le niveaux de gris est inférieur à 128.

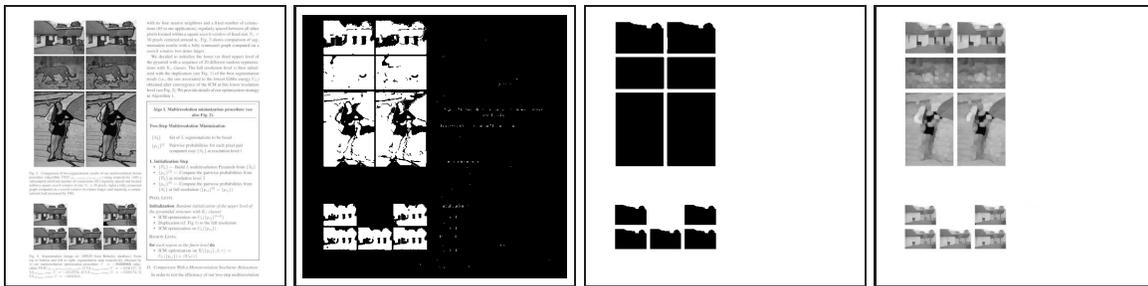
- Mais la stratégie la plus simple consiste à faire sur l'image un simple pré-traitement qui associe à chaque pixel de l'image; le niveaux de gris le plus clair (ou le plus important) dans un voisinage qui serait par exemple une croix, centrée sur le pixel, et de taille, par exemple,  $7 \times 7$  puis de lancer l'algorithme tel qu'il a été programmé avec le K-moyenne (cf. résultats ci-dessous).

- On pourrait utiliser comme vraisemblance aussi, des caractéristiques locales de textures comme par exemple le calcul de l'entropie, calculée sur une petite imagerie centrée sur le pixel à classer. L'entropie serait faible pour le texte (car son histogramme est *éparse*) et plus grande pour un pixel d'une image (histogramme *plus étalé*).

- On pourrait peut-être utiliser une vraisemblance avec deux probabilités *ad-hoc* basées sur un triple seuils du style; si on a un niveaux de gris d'amplitude  $< T_{1er\ seuil} = 30$  ou  $> T_{2eme\ seuil} = 225$ , alors on a la classe 1 (*texte*) avec la probabilité disons 0.75 et la classe 2 (*Figure*) avec 0.25 et dans le cas contraire  $> T_{1er\ seuil} = 30$  et  $< T_{2eme\ seuil} = 225$ , alors on a la classe 2 avec la probabilité disons 0.75 et la classe 1 avec 0.25. Ensuite, le modèle *a priori* de l'ICM permettrait de régulariser toutes les mauvaises classifications de cette vraisemblance.

- On pourrait aussi utiliser sur des fenêtres locales (du type  $7 \times 7$ ) à chaque pixel, la valeur de la moyenne du module du gradient (faible pour les images et plus élevé pour le texte).

**NOTA :** Avec cette simple stratégie on obtient pour l'ICM: (de gauche à droite; image originale, stratégie 1, stratégie 2 et image originale pré-traitée par la méthode expliquée ci dessus (stratégie la plus simple)).



<8 pts>