Synthèse théorique Traitement des signaux et données

Table des matières

Introduction

Le traitement de signal a de nombreuses applications bien connues tel que la reconnaissance vocale, l'imagerie en radiologie, l'imagerie satellite, …

1. Définitions

- Signal : représentation physique de l'information qu'il transporte

- Bruit : phénomène perturbateur qui gêne la perception ou l'interprétation du signal
  attention : ce qui peut être un bruit dans un domaine peut être un signal utile dans un autre

- Théorie du signal : description mathématique du signal mettant en évidence ses caractéristiques (sa formule)

- Traitement du signal : discipline technique basée sur l'électronique, l'informatique et la physique appliquée ayant pour objet l'élaboration et l'interprétation des signaux

- Traitement de l'information : ensemble de concepts pour évaluer les performances des systèmes de transfert d'informations

2. Principales fonctions

- Élaboration des signaux
  - Synthèse : création de signaux de forme appropriée
  - Modulation : adaptation du signal pour sa transmission (surtout réduire la bande passante qu'il utilise)
  - Codage

- Interprétation de signaux
  - Filtrage : élimination de certaines composantes fréquentielles
  - Fenêtrage : élimination de certaines composantes temporelles
  - Détection : extraction du signal du bruit
  - Identification : classification du signal
  - Analyse : isolement de composantes essentielles ou utiles
  - Mesure : estimation d'une grandeur caractéristique d'un signal

NB : pour le filtrage et le fenêtrage, l'idée est la même c'est juste histoire de spécifié si on travaille dans le fréquentiel ou dans le temporel

Chapitre 1 – Représentation des signaux analogiques

1. Modélisation des signaux

En pratique	En théorie
Un signal est une grandeur physique et doit donc être réalisable. Caractéristiques : • Énergie < ∞ • Amplitude < ∞ • Continu temporellement • Causal : $s(t) = 0$ pour $t < 0$ • Spectre du signal borné tend vers 0 quand la fréquence tend vers l'infini	Pour faciliter les calculs, on travaille avec des signaux représentés par des fonctions. Caractéristiques : • Énergie théorique infinie • Avec des discontinuité (signal carré) • Définie sur l'ensemble des réels (non causaux) • Spectre infini • A valeurs complexes : $s(t) = Ae^{j\omega t}$ $= A[\cos(\omega t) + j\ \sin(\omega t)]$

NB : en fait la différence entre les signaux théorique et pratique, c'est qu'en théorie on travaille avec des signaux qui passent d'un état à un autre instantanément (un signal carré qui passe d'un état à un autre sans transition) alors que dans la pratique c'est impossible, il y a toujours un temps de transition.

2. Classification des signaux

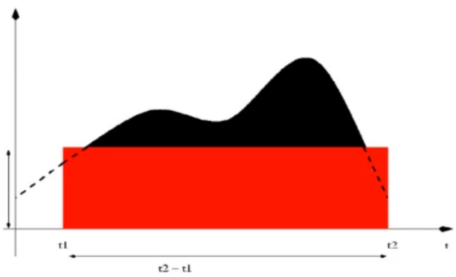
a. Classification temporelle

i. Signaux certains ou déterministes
Évolution du signal parfaitement décrit par une équation mathématique. Peuvent être périodique ou non.

ii. Signaux aléatoires
Comportement temporel imprévisible (ex : bruit)

b. Classification énergétique

i. Signaux à énergie finie
S'applique aux signaux transitoires



$$\int_{-\infty}^{+\infty} s^2(t)dt < \infty$$

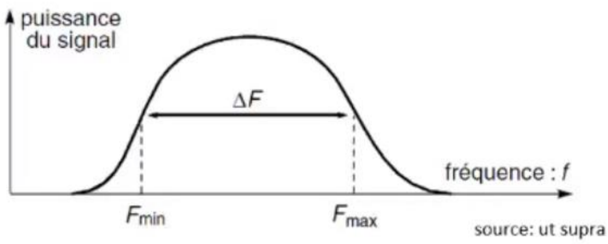
$$P_{moy} = \frac{\int_{t_1}^{t_2} s^2(t)dt}{t_2 - t_1}$$

ii. Signaux à puissance moyenne finie
S'applique aux signaux périodiques ou quasi-périodique

$$\lim_{T\to\infty} \frac{1}{T}\int_{-T/2}^{T/2} s^2(t)dt < \infty$$

c. Classification spectrale

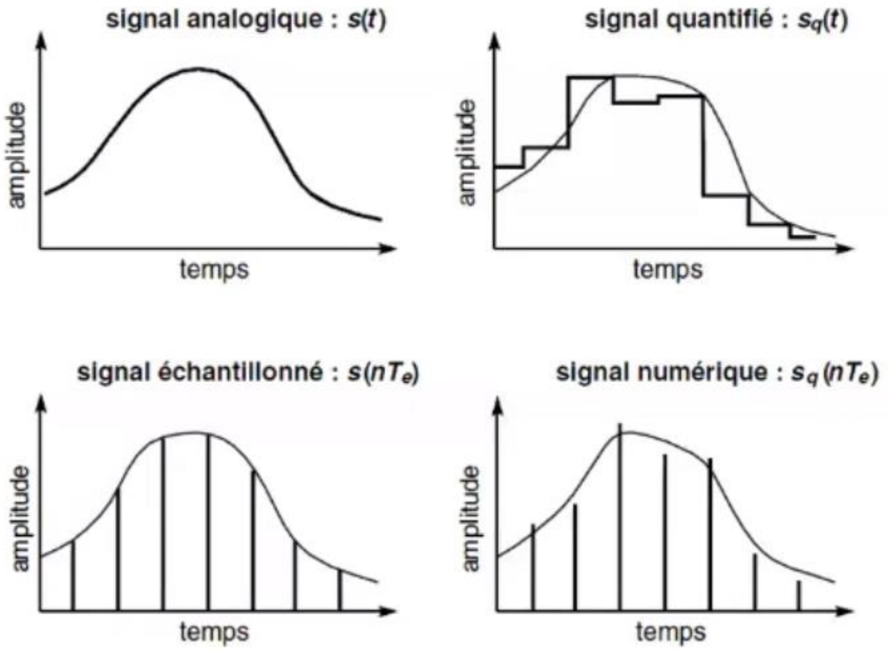
Selon la distribution de son énergie ou de sa puissance en fonction de la fréquence (spectre du signal). Le domaine des fréquences occupé par ce signal est appelé la largeur de bande du signal.



$$\Delta F = F_{\max} - F_{\min}$$

d. Les signaux numériques

Numérisation => opération faisant passer le signal du domaine des temps et amplitudes continus au domaine des temps et amplitudes discrets. (Echantillonnage + Quantification)



3. Quantification

- On doit « discrétisée » l'information pour pouvoir la stocker … pour cela on utilise des convertisseurs analogique/numériques

$$\Delta V = \frac{V_{ref}}{2^n}$$

n = nb de bits

V_{ref} = gamme de conversion en tension

- Principe de base : le pas de quantification détermine le delta-valeurs (ΔV) entre 2 échantillons proches. Ce pas est en relation avec la quantité de bits du convertisseur

- A chaque échantillonnage, le CAN décide si la tension du signal à l'instant t est plus proche de $(j-1)\ \Delta V$ ou de $j\Delta V$

- Dans une quantification uniforme dans toute la gamme de tension, l'erreur de quantification $\varepsilon_q(t)$ est toujours de $-\frac{\Delta V}{2} \leq \varepsilon_q(t) \leq \frac{\Delta V}{2}$

- $\left({}^{S}/_{N}\right)_q = 6\left(\frac{A}{Vref}\right)^2\ Nb\ max^2$

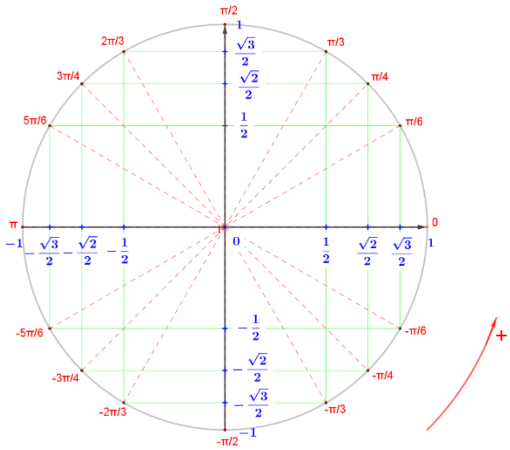
- Le rapport S/N dépend du carré de l'amplitude du signal d'entrée … la qualité de la quantification dégrade donc considérablement mes performances S/N des signaux à faibles dynamiques (rapport A/Vref faible)

- Le rapport S/N s'améliore en fonction du carrée de nombre de pas de quantification, càd en fonction de la puissance 2 du nombre de bits du convertisseur A/N

Chapitre 2 – Transformée de Fourier

1. Rappels

a. Trigo



x	0	$\frac{\pi}{6}$	$\frac{\pi}{4}$	$\frac{\pi}{3}$	$\frac{\pi}{2}$	π	2π
$\cos x$	1	$\frac{\sqrt{3}}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{1}{2}$	0	-1	1
$\sin x$	0	$\frac{1}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{\sqrt{3}}{2}$	1	0	0
$\tan x$	0	$\frac{\sqrt{3}}{3}$	1	$\sqrt{3}$	Non définie	0	0

b. Euler & Dirac

Les formules d'Euler

1e équation d'Euler => $\cos\theta = \frac{e^{j\theta}+e^{-j\theta}}{2}$

2e équation d'Euler => $\sin\theta = \frac{e^{j\theta}-e^{-j\theta}}{2j}$

Distribution de Dirac

Distribution de Dirac : $f(x) = \begin{cases} 0 \; quand \; x \neq 0 \\ +\infty \; quand \; x = 0 \end{cases}$

Sa surface unitaire (appelée poids de la distribution) vaut 1 : $\int_{-\infty}^{+\infty} \delta(x)\,dx = 1$

2 propriétés :

$$\boxed{\begin{array}{c} f(x) * \delta(x) = f(0) * \delta(x) \\ f(x) * \delta(x-a) = f(a) * \delta(x-a) \end{array}}$$

c. Intégrales - primitives – dérivées

Tableau des dérivées

f	f'
$constante$	0
x	1
x^2	$2x$
x^3	$3x^2$
x^n	$n \times x^{n-1}$
$\frac{1}{x}$	$-\frac{1}{x^2}$
$\sqrt{x}$	$\frac{1}{2\sqrt{x}}$
$sin(x)$	$cos(x)$
$cos(x)$	$-sin(x)$
e^x	e^x
$ln(x)$	$\frac{1}{x}$

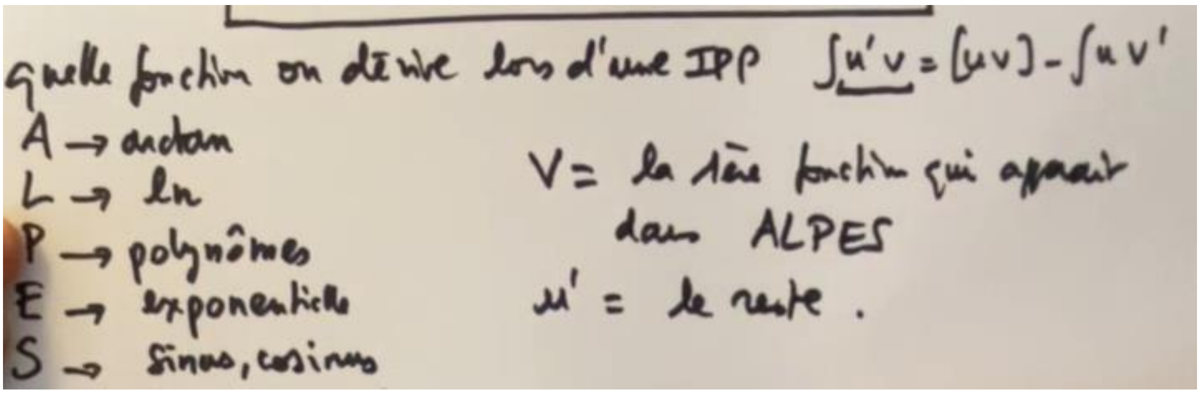
Tableau des primitves composées

f	F (LA PRIMITIVE)
$u' \times u$	$\frac{u^2}{2}$
$u' \times u^2$	$\frac{u^3}{3}$
$u' \times u^3$	$\frac{u^4}{4}$
$u' \times u^n$	$\frac{u^{n+1}}{n+1}$
$\frac{u'}{u}$	$ln(u)$
$\frac{u'}{u^2}$	$-\frac{1}{u}$
$\frac{u'}{u^n} = u' \times u^{-n}$	$\frac{u^{-n+1}}{-n+1} = \frac{1}{(-n+1) \times u^{n-1}}$
$\frac{u'}{2 \times \sqrt{u}}$	$\sqrt{u}$
$u' \times e^u$	e^u
$u' \times sin(u)$	$-cos(u)$
$u' \times cos(u)$	$sin(u)$

Tableau des primitives

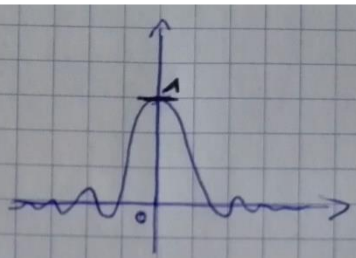
f	F (LA PRIMITIVE)
0	$constante$
1	x
x	$\frac{x^2}{2}$
x^2	$\frac{x^3}{3}$
x^3	$\frac{x^4}{4}$
x^n	$\frac{x^{n+1}}{n+1}$
$\frac{1}{x}$	$ln(x)$
$\frac{1}{x^2}$	$-\frac{1}{x}$
$\frac{1}{x^n} = x^{-n}$	$\frac{x^{-n+1}}{-n+1} = \frac{1}{(-n+1) \times x^{n-1}}$
$\frac{1}{2\sqrt{x}}$	$\sqrt{x}$
e^x	e^x
$sin(x)$	$-cos(x)$
$cos(x)$	$sin(x)$

Intégration par partie : https://www.youtube.com/watch?v=cEW-nI5mbn0



d. sinus cardinal $sinc()$

- $\dfrac{\sin(x)}{x} = \mathrm{sinc}(x)$



$\mathrm{sinc}(0) = \dfrac{\sin(0)}{0}$

ce n'est pas une indétermination (div par 0)

$\Rightarrow f(x) = \sin(x)$ et $g(x) = x$ sont des Infiniments Petits Equivalents (IPE)

$\lim\limits_{x \to 0} \dfrac{\cancel{\sin}\, x}{\cancel{x}} = 1$ (on peut simplifier $\sin(x)$ et x)

sinc() est une fonction paire

$\mathrm{sinc}(-x) = \dfrac{\sin(-x)}{-x} = \dfrac{-\sin(x)}{-x} = \dfrac{\sin(x)}{x}$

2. Fourier

a. Séries de Fourier

Matière 2e

(Pas d'explications, c'est le 1er exercice de chap_2_Fourier.pdf)

b. Coefficients de Fourier

$$a_0 = \frac{1}{T}\int_{-T/2}^{T/2} s(t)\, dt$$ (Valeur moyenne du signal)

$$a_n = \frac{2}{T}\int_{-T/2}^{T/2} s(t).\cos(2\pi\,.n\,.F_0\,.t)\, dt$$

$$b_n = \frac{2}{T}\int_{-T/2}^{T/2} s(t)\,.\,\sin(2\pi\,.n\,.F_0\,.t)\, dt$$

ASTUCE

- Si la **fonction est impaire**, càd qu'on peut lui appliquer une symétrie centrale de 180° sans la modifier, tous les coefficients $\boldsymbol{a_n}$ **valent 0**.
(Attention : a_0 n'est pas pris en compte dans les a_n)

- Si la **fonction est paire**, càd que l'on peut lui appliquer une symétrie orthogonale d'axe y sans la modifier, tous les coefficients $\boldsymbol{b_n}$ **valent 0**.

c. Expression générale de Fourier

Transformée de Fourier : $S(f) = \int_{-\infty}^{+\infty} s(t)\,.\,e^{-j\,.\,2\pi\,.\,f\,.\,t}\; dt$

Transformée de Fourier inverse : $s(t) = \int_{-\infty}^{+\infty} S(f)\,.\,e^{j\,.\,2\pi\,.\,f\,.\,t}\; df$

d. Quelle formule utiliser ?

- Selon le type de signal.
- Signal périodique à énergie infinie ET sans points anguleux (signal sinus, cos)

  ⇨ Séries de Fourier, distribution de Dirac, formules Euler

- Signal périodique à énergie infinie ET avec points anguleux (signal carré, dents de scies, …)

  ⇨ Coefficients de Fourier

- Si aucun des 2 types de signaux ci-dessus

  ⇨ Expression générale de Fourier

NB : énergie infinie signifie qu'il est à durée illimité (sans bornes)

3. Exercices

Cf. : chap_2_Fourier.pdf

4. Existence et propriétés

a. Existence

La transformée de Fourier $S(f)$ d'un signal n'existe que si l'énergie du signal est finie.

$$\int_{-\infty}^{+\infty} s^2(t)dt < \infty$$

b. Linéarité

La transformée de Fourier d'une somme de signaux, c'est la somme des transformées de Fourier de chacun de ses signaux.

$$a.s1(t) + b.s2(t) \; <=> a.S1(f) + b.S2(f)$$

c. Translation dans le temps

Déphasé un signal induit un déphasage apparaissant au niveau du spectre

$$s(t-a) \; <=> S(f).e^{-j2\pi af}$$

d. Translation en fréquence

Faire une translation en fréquence induit un déphasage dans la représentation temporelle du signal

$$s(t).e^{j2\pi bt} <=> s(f-b)$$

e. Théorème de la modulation de Fourier

Lorsque l'on multiplie un signal par une porteuse, on voit apparaitre le spectre du signal de base autour des fréquences de la porteuses (ce qu'il se passe dans le dernier exercice de Fourier … Cf. : chap 2 Fourier.pdf)

Propriété permettant de modifier la fréquence d'un signal (modulation)

De l'équation de la propriété de translation en fréquence, nous déduisons que

$$x(t)e^{-j2\pi bt} \leftrightarrow X(f+b)$$

Calculons maintenant la demi-somme de ces deux équations,

$$\frac{x(t)e^{j2\pi bt} + x(t)e^{-j2\pi bt}}{2} \leftrightarrow \frac{X(f-b)+X(f+b)}{2}$$

$$x(t)\cos(2\pi bt) \leftrightarrow \frac{1}{2}(X(f-b)+X(f+b))$$

f. Cosinus limité dans le temps

(exo de fin de chap 2 Fourier.pdf)

Chapitre 3 – Systèmes de transmission

1. Définitions

- Un système de transmission fait correspondre à signal d'entrée $e(t)$ à un signal de sortie $s(t)$
- $s(t)$ dépend de $e(t)$ et des caractéristiques du ST
- On peut caractériser le rapport entre $s(t)$ et $e(t)$

$$A_{db} = 10\log_{10}\left({}^{s(t)}\!/_{e(t)}\right)$$

- Gain/perte en puissance : $A_{db} = 10\log_{10}\left({}^{P_s}\!/_{P_e}\right)$

- Gain/perte en tension : $A_{db} = 20\log_{10}\left({}^{V_s}\!/_{V_e}\right)$

Rapport des tensions V_s/V_e	$1/10$	$1/2$	$1/\sqrt{2}$	2	10	100
Gain ou affaiblissement en db	-20	-6	-3	6	20	40

- Bande passante : tranche des fréquences pour lesquelles l'affaiblissement de la puissance de sortie, à puissance entrante constante, est inférieur de 3dB par rapport à sa valeur maximale.

2. Propriétés des systèmes de transmission

a. Linéarité

$$a.e_1(t) + b.e_2(t) \quad \overset{S.T}{\rightarrow} \quad a.s_1(t) + b.s_2(t)$$

« Si on fait passer dans un ST une somme de signaux, en sortie, on va retrouver la somme de s1 et s2. »

« Lorsqu'on injecte une somme de signaux dans un ST, on peut évaluer, séparément les effets du ST sur chacun des signaux et en sortie effectué à nouveau la somme des résultats. »

b. Continu

$$\lim_{n \to +\infty} e_n(t) \quad \overset{S.T}{\rightarrow} \quad \lim_{n \to +\infty} s_n(t)$$

« Pour tout échantillon d'entrée doit correspondre un échantillon de sortie. »

c. Stationnaire

$$e(t-\theta) \quad \overset{S.T}{\rightarrow} \quad s(t-\theta)$$

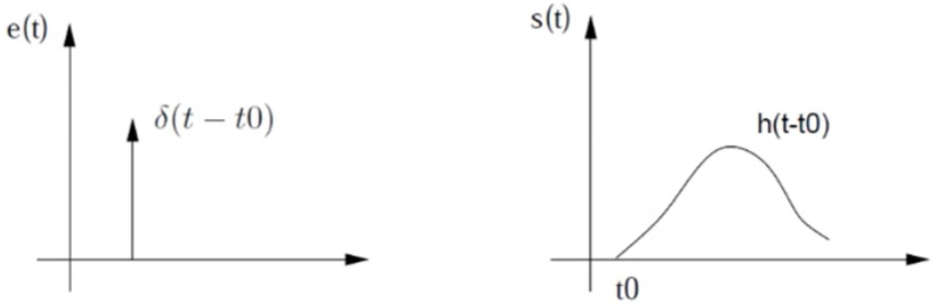
« Si son comportement est indépendant de l'origine du temps »

« Un système qui conserve les mêmes caractéristiques physiques au cours du temps »

3. Filtres et convolution

- Filtres : ST linéaires, continus et stationnaires
- Réponse impulsionnelle : une impulsion brève, injectée dans un ST linéaire, continu et stationnaire, ne donne jamais en sortie une impulsion infiniment brève mais un signal de durée finie.

$$h(t) = L.\delta(t)$$



- Opération de convolution :
    - Notée *
    - Réponse à un signal quelconque à partir de celle à un signal type (réponse impulsionnelle). Elle dépend du filtre h(t) et de l'instant T

$$e(t) \xrightarrow{\text{S.T.-L.C.S.}} s(t) = \int_{-\infty}^{+\infty} e(\tau) \cdot h(t-\tau) \cdot \, \mathrm{d}\tau = e(t) * h(t)$$

*S.T-L.C.S => Système de Transmission Linéaire, Continu et Stationnaire (un filtre quoi ….)

Remarque: convolution des signaux périodiques

$$P_{\text{conv}}(t) = \frac{1}{T_0} \int_0^{T_0} x(\tau) \; y(t-\tau) \; \mathrm{d}\tau$$

- Les filtres sont donc des systèmes de convolution
- Donc un filtre atténue ou amplifie les composantes fréquentielles de e(t)

(les opérations de convolution sont très lourdes, c'est pour cela qu'on utilise des outils comme Matlab)

- Propriétés de la convolution
  - Commutativité

$$x * y = y * x$$

  - Distributivité

$$x * (y + z) = x * y + x * z$$

  - Associativité

$$x * (y * x) = (x * y) * z$$

  - Élément neutre : distribution de Dirac

$$x * \delta = \delta * x = x$$

(* représentant l'opération de convolution et non une multiplication)

- **<u>Théorème de Plancherel</u>**
  - <u>La transformée de Fourier d'un produit de convolution est un produit simple et réciproquement</u>.

$$x(t) * y(t) \overset{F}{\leftrightarrow} X(f).Y(f) \qquad \text{et} \qquad x(t).y(t) \overset{F}{\leftrightarrow} X(f) * Y(f)$$

- **<u>Théorème de Parseval</u>**
  - L'énergie totale d'un signal ne dépend pas de la représentation choisie (temporelle ou fréquentielle)

$$E = \int_{-\infty}^{+\infty} [x(t)]^2 dt = \int_{-\infty}^{+\infty} |X(f)|^2 df$$

Exemples de convolution discrète

Convolution discrète

$x = [1 ; 2 ; 3]$

$y = [4 ; 5 ; 6] \xrightarrow{(1)} [6 ; 5 ; 4]$

$z = x * y$

$[1 \quad 2 \quad 3]$
$[6 \quad 5 \quad 4]$ — $1.4 = 4$

$[1 \quad 2 \quad 3]$
$[6 ; 5 ; 4]$ — $1.5 + 2.4 = 13$

$[1 \quad 2 \quad 3]$
$[6 \quad 5 \quad 4]$ — $1.6 + 2.5 + 3.4 = 28$

$[1 \quad 2 \quad 3]$
$[6 \quad 5 \quad 4]$ — $2.6 + 3.5 = 27$

$[1 \quad 2 \quad 3]$
$[6 \quad 5 \quad 4]$ — $3.6 = 18$

(2) (3) (4)

(1) Symétrie sur le vecteur y $(-\tau)$

(2) Translation

(3) Multiplication des termes alignés

(4) Addition des multiplication

$\Rightarrow z = [4 ; 13 ; 28 ; 27 ; 18]$

$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$ $h = \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}$ $-h = \begin{pmatrix} 8 & 7 \\ 6 & 5 \end{pmatrix}$

① conv (A, h)

$$\begin{matrix} 8 & 7 & \\ 6 & 5\,1 & 2 \\ & 3 & 4 \end{matrix} \qquad 5.1 = 5$$

$$\begin{matrix} 8 & 7 \\ 6\,1 & 5\,2 \\ 3 & 4 \end{matrix} \qquad \begin{aligned} 6.1 &= 6 \\ 5.2 &= 10 \; + \\ \hline &16 \end{aligned}$$

$$\begin{matrix} & 8 & 7 \\ 1 & 6\,2 & 5 \\ 3 & 4 & \end{matrix} \qquad 6.2 = 12$$

on descend d'une ligne

$$\begin{matrix} 8 & 7\,1 & 2 \\ 6 & 5\,3 & 4 \end{matrix} \qquad \begin{aligned} 7.1 &= 7 \\ 5.3 &= 15 \; + \\ \hline &22 \end{aligned}$$

$$\begin{pmatrix} 1\,8 & 2\,7 \\ 3\,6 & 4\,5 \end{pmatrix} \qquad \begin{aligned} 1.8 &= 8 \\ 2.7 &= 14 \\ 3.6 &= 18 \\ 4.5 &= 20 \; + \\ \hline &60 \end{aligned}$$

$$\begin{matrix} 1 & 2\,8 & 7 \\ 3 & 4\,6 & 5 \end{matrix} \qquad \begin{aligned} 2.8 &= 16 \\ 4.6 &= 24 \; + \\ \hline &40 \end{aligned}$$

on descend encore...

$$\begin{matrix} & 1 & 2 \\ 8 & 3\,7 & 4 \\ 6 & 5 & \end{matrix} \qquad 3.7 = 21$$

$$\begin{matrix} 1 & 2 \\ 3\,8 & 4\,7 \\ 6 & 5 \end{matrix} \qquad \begin{aligned} 3.8 &= 24 \\ 4.7 &= 28 \; + \\ \hline &52 \end{aligned}$$

$$\begin{matrix} 1 & 2 & \\ 3 & 4\,8 & 7 \\ & 6 & 5 \end{matrix} \qquad 4.8 = 32$$

$$\Rightarrow \begin{pmatrix} 5 & 16 & 12 \\ 22 & 60 & 40 \\ 21 & 52 & 32 \end{pmatrix}$$

Pour faire une convolution sur une image nous n'utilisons pas conv(A, h) car cette fonction agrandi l'image (cf. Screenshot ci-dessus, il y a plus de données dans z que dans A et h). Pour les images nous utilisons imfilter() car nous ne voulons pas forcement les agrandir …. Mais le principe de la convolution est le même partout.

For example, suppose the image is

```
A = [17  24   1   8  15
     23   5   7  14  16
      4   6  13  20  22
     10  12  19  21   3
     11  18  25   2   9]
```

and the convolution kernel is

```
h = [8   1   6
     3   5   7
     4   9   2]
```

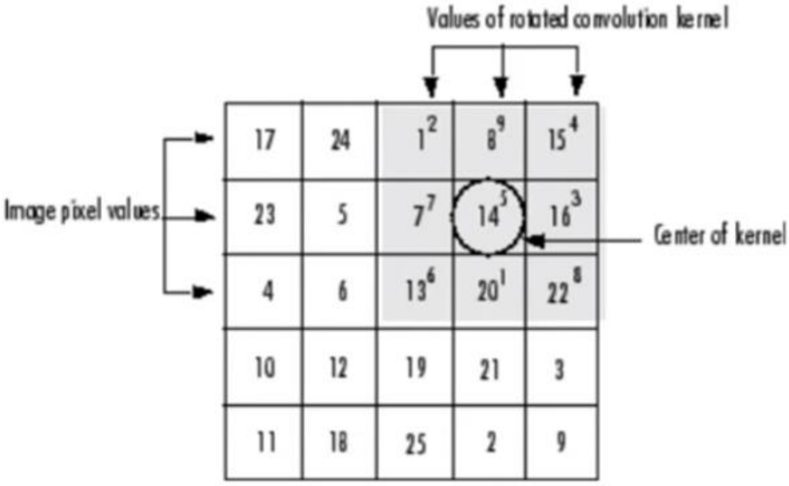
The following figure shows how to compute the (2,4) output pixel using these steps:

1. Rotate the convolution kernel 180 degrees about its center element.
2. Slide the center element of the convolution kernel so that it lies on top of the (2,4) element of A.
3. Multiply each weight in the rotated convolution kernel by the pixel of A underneath.
4. Sum the individual products from step 3.

Hence the (2,4) output pixel is

$$1 \cdot 2 + 8 \cdot 9 + 15 \cdot 4 + 7 \cdot 7 + 14 \cdot 5 + 16 \cdot 3 + 13 \cdot 6 + 20 \cdot 1 + 22 \cdot 8 = 575$$

Computing the (2,4) Output of Convolution



17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	21	3
11	18	25	2	9

x translat° vers la droite plus tard...

17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	21	3
11	18	25	2	9

ligne suivante

17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	21	3
11	18	25	2	9

x translat° vers la droite plus tard...

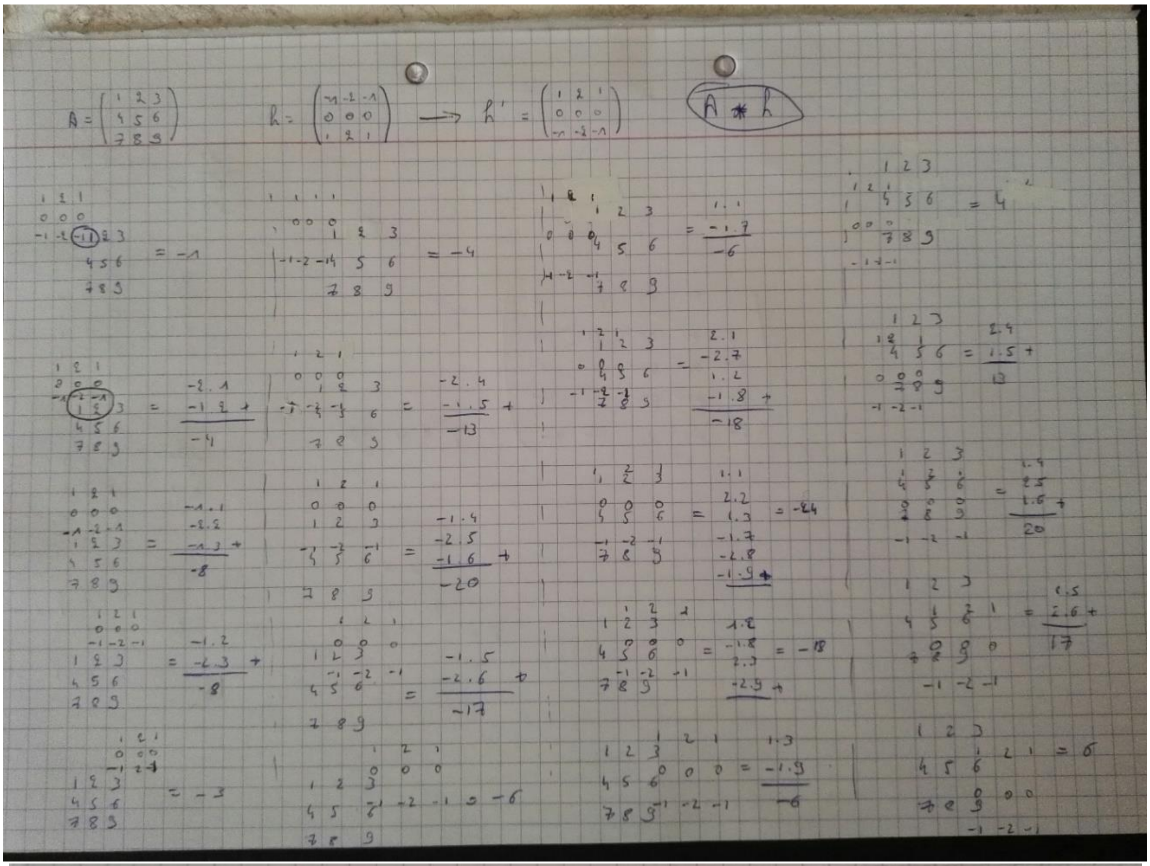
17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	21	3
11	18	25	2	9

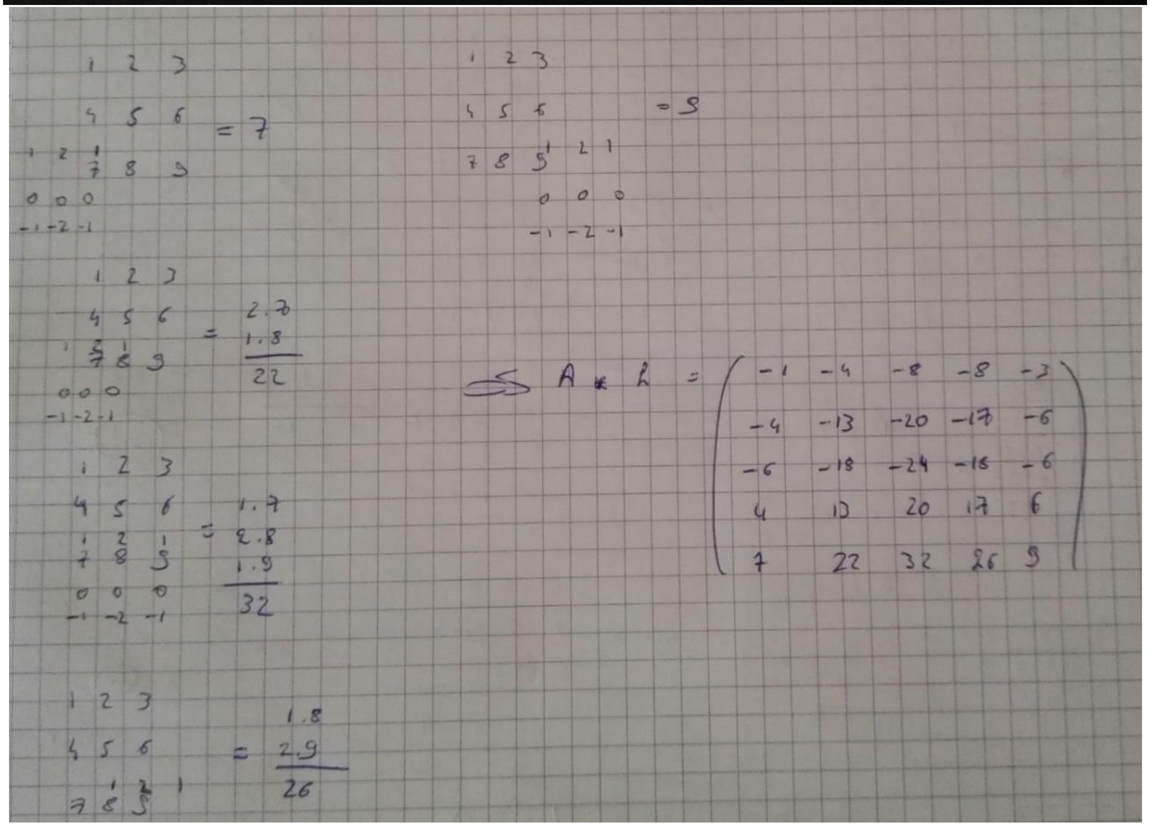
Je vais direct à la dernière ligne pour vous montrer la fin

17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	21	3
11	18	25	2	9

x translat° vers la droite plus tard

17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	21	3
11	18	25	2	9





4. Convolution et traitement d’images

- Les filtres appliqués aux images par des logiciels, tel que Photoshop et Gimp, ne sont en fait que des matrices de convolution.

5. Notion de corrélation

- Dans l'opération de convolution, la matrice h est retourné … dans l'opération de corrélation, elle ne l'est pas.

$$C_{xy} = \int_{-\infty}^{\infty} x(\tau)\,.\;y(\tau - t)\,d\tau$$

- Si x et y sont identiques, on parle d'autocorrélation et se note alors C_{xx}

- Les fonctions de corrélation traduisent la similitude d'un ou deux signaux au niveau de la forme et de la position en fonction du paramètre de translation t.
- Dans le cas d'une fonction d'autocorrélation, c'est une étude de la ressemblance du processus avec lui-même au cours du temps. Par conséquent, si le signal est périodique, la fonction d'autocorrélation permettra de détecter cette périodicité.

- Propriétés importantes de la fonction d'autocorrélation :

  - Pour les signaux réels, la fonction d'autocorrélation est paire
  $$C_{xx}(t) = C_{xx}(-t)$$

  - La fonction d'autocorrélation a sa valeur max pour t=0
  $$C_{xx}(t) \leq C_{xx}(0)$$

  - Si x(t) est périodique, la fonction d'autocorrélation possède toutes les fréquences comprises dans le système initial et seulement celles-ci. Elle conserve l'information fréquence mais pas les informations de phase et d'amplitude.

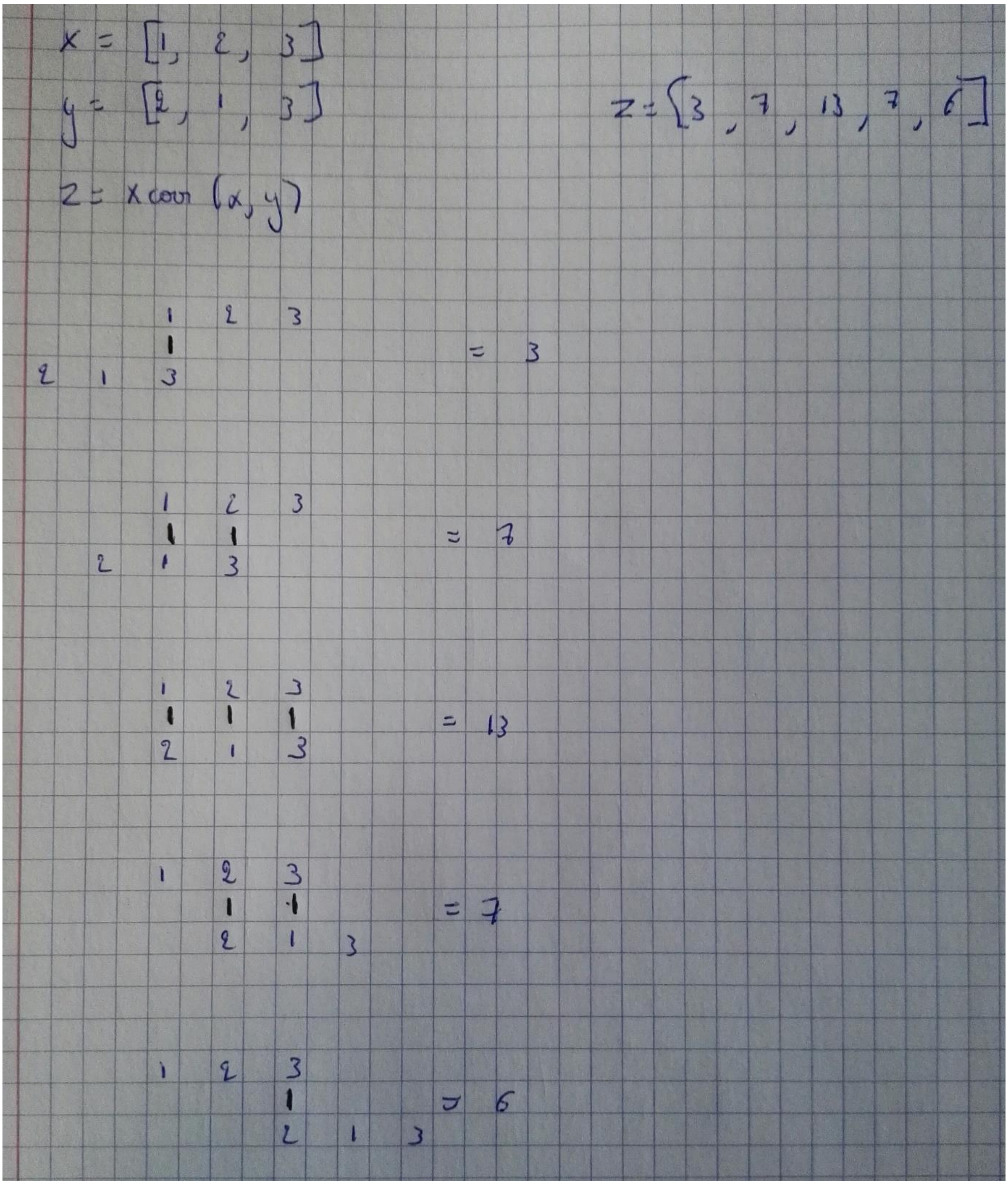
  - L'autocorrélation d'un signal aléatoire est la distribution δ

- Relation entre la densité spectrale :
$$C_{xx}(t) \xleftrightarrow{Fourier} S_{xx}(f)$$

  Et pour 2 signaux x(t) et y(t) :
  $$C_{xy}(t) \xleftrightarrow{Fourier} S_{xy}(f) \text{ ou } C_{yx}(t) \xleftrightarrow{Fourier} S_{yx}(f)$$

  La transformée de Fourier de la fonction de corrélation du signal représente la densité spectrale de l'énergie, càd la redistribution de l'énergie sur l'axe des fréquences.
  Ainsi, il sera souvent plus facile de calculer la fonction d'autocorrélation ou l'intercorrélation d'un signal en passant par son spectre ou sa densité spectrale.

- Le passage de la corrélation à la corrélation discrète se fera de la même manière que pour la convolution.
  Attention, on ne retourne pas le vecteur h dans l'opération de corrélation



x = [12 3 4 1 2 3 5 6]

y = [1 2 3]

↳ [1 2 3 0 0 0 0 0]

[x et y doivent avoir la m̂ taille

Z = xcorr (x, y)

```
                12 3 4 1 2 3 5 6     = 0
1 2 3 0 0 0 0 0

              12 3 4 1 2 3 5 6       = 0
1 2 3 0 0 0 0 0

            12 3 4 1 2 3 5 6         = 0
1 2 3 0 0 0 0 0

          12 3 4 1 2 3 5 6           = 0
1 2 3 0 0 0 0 0

        12 3 4 1 2 3 5 6             = 0
1 2 3 0 0 0 0 0

      12 3 4 1 2 3 5 6               = 36
1 2 3 0 0 0 0 0

    12 3 4 1 2 3 5 6                 = 33
1 2 3 0 0 0 0 0

12 3 4 1 2 3 5 6                     = 30
1 2 3 0 0 0 0 0

12 3 4 1 2 3 5 6                     = 14
   1 2 3 0 0 0 0 0

3 4 1 2 3 5 6                        = 12
  1 2 3 0 0 0 0 0

4 1 2 3 5 6                          = 14
  1 2 3 0 0 0 0 0

1 2 3 5 6                            = 23
1 2 3 0 0 0 0 0

2 3 5 6                              = 31
  1 2 3 0 0 0 0 0

3 5 6                                = 17
  1 2 3 0 0 0 0 0

6                                    = 6
1 2 3 0 0 0 0 0
```

Z = [0 0, 0, 0, 0, 36, 33, 30, 14, 12, 14, 23, 31, 17, 6]

6. Décorrélation et ICA (Independent Component Analysis)

- ACP (Analyse des Composantes principales)
    - Consiste à transformer des variables liées entre elles (corrélées) en nouvelles variables indépendantes les unes des autres (non corrélées).
    - Ces variables indépendantes sont nommées « composantes principales »

- ICA
    - Méthode statistique permettant de transformer un ensemble de valeurs aléatoires en sous-ensembles qui sont statistiquement indépendants les uns des autres.

    - En Statistique, 2 variables sont considérées indépendantes lorsque le coefficient de corrélation calculé entre ces deux variables tend vers 0.

$$C_{xy}(t) \to 0$$

    - Application : Cocktail Party
        - ICA permet de résoudre ce problème en considérant que les différents signaux mélangés sont des signaux aléatoires statistiquement indépendants (conversations indépendantes les unes des autres)

Chapitre 4 – Filtrage des signaux analogiques

Ces 2 opérations sont en faites les mêmes, c'est juste qu'on parlera de « fenêtrage » quand on travaille dans le domaine temporel et de « filtrage » quand on travaille dans le domaine fréquentiel.

1. Fenêtrage temporel

- <u>Opération consistant à prélever, interrompre ou atténuer un signal ou une partie du domaine temporel d'un signal</u>

$$s(t) = gfen(t).e(t) \overset{T.F}{\leftrightarrow} S(f) = Gfen(f) * E(f)$$

En français ….

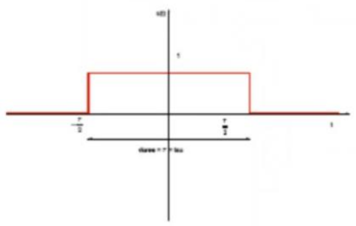
- Le signal de sortie s(t) correspond au produit du signal d'entrée e(t) par la fonction temporelle du filtre
- Le spectre en sortie S(f) correspond au produit de convolution du spectre du signal d'entrée E(t) par le spectre du filtre. (Théorème de Plancherel)
- On peut passer d'une égalité à l'autre via le théorème de Fourier

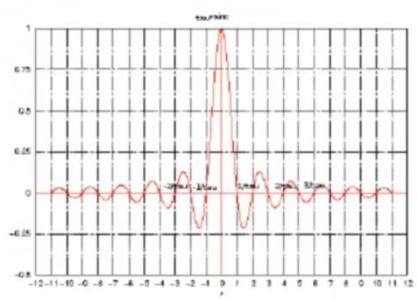
Il faut également limiter le signal dans le temps. (Ça n'a pas de sens d'avoir une fenêtre temporelle infinie)

Pour cela, on multiplie le signal d'entrée e(t) par un signal limité dans le temps, tel qu'une fonction porte ($\pi_\tau(t)$).

Et nous savons que la transformée de Fourier de cette fonction porte est un sinc.

$$s(t) = \pi_\tau(t).e(t) \leftrightarrow S(f) = \tau.sinc(f\tau) * E(f)$$



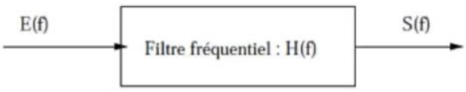


Conclusion :

Lorsqu'on prélève une partie d'un signal, on étale l'énergie autour de la fréquence de base du signal et plus on utilise une porte chère et complexe à mettre en place, moins le spectre sera affecté et inversement.

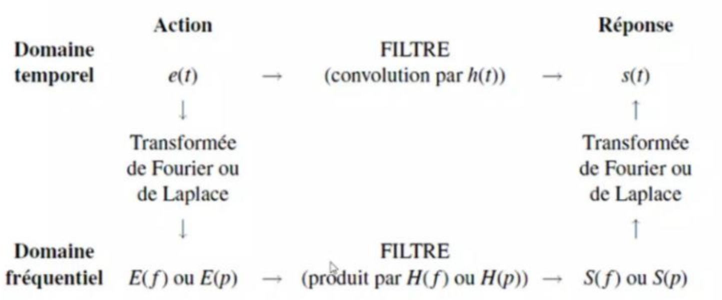
2. Filtrage fréquentiel

- Opération consistant à <u>prélever, interrompre ou atténuer un signal ou une partie des composantes fréquentielles d'un signal</u>



$$s(t) = h(t) * e(t) \overset{T.F}{\Leftrightarrow} S(f) => H(f).E(f)$$

« Fenêtrage temporelle, il suffit de multiplier dans le domaine temporel et c'est un produit de convolution dans le domaine fréquentiel.
Filtrage fréquentielle, il suffit de multiplier dans le domaine fréquentiel et c'est un produit de convolution dans le domaine temporel. »



Source: Cottet F., « Traitement des signaux et acquisition des données » 2ème édition. Dunod 2002

Explications du schéma :

- Dans le domaine temporel, si on veut connaitre la réponse $s(t)$, on doit appliquer une convolution $e(t) * h(t)$
- Dans le domaine fréquentiel, si on veut connaitre la réponse $S(f)$, on doit faire un produit simple $E(f).H(f)$

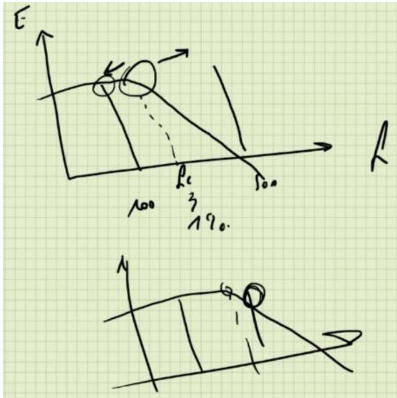
- Par définition, <u>à la fréquence de coupure, l'atténuation d'un filtre est toujours de -3dB</u> peut importe l'ordre de ce dernier.

- <u>Un filtre Butterworth d'ordre n atténuera de</u> (Pour des fréquences > fréquence de coupure)
  - <u>-6n dB par octave et par ordre</u>
  - <u>-20n dB par décade et par ordre</u>

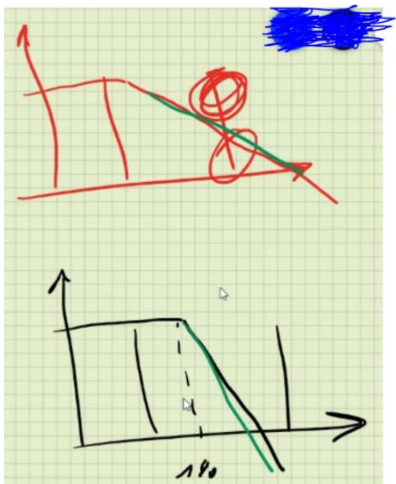
- <u>Un filtre est réalisable s'il est causal</u> càd si sa réponse $h(t) = 0 \forall t < 0$
  L'effet ne peut précéder la cause. (Cf. Chapitre 5 … principe de causalité)

- **<u>Question d'examen : Voici un signal à 1000Hz et un filtre avec une fréquence de coupure à 100Hz. Si le filtre est d'ordre 2 qu'elle sera l'atténuation du 1000Hz.</u>**
  - ⇨ **Entre 100 et 1000 je multiplie par 10 (décade).**
  - ⇨ **On sait qu'on atténue de 20dB par décade**
  - ⇨ **Le filtre est d'ordre 2 donc j'atténue de $2 \cdot 20dB$ donc j'atténue de – 40dB**

- Comment déterminer la fréquence de coupure et l’ordre du filtre ?
    - Fréquence de coupure :
        - Par essai-erreur … il n’y a pas de formule pour déterminer la fréquence parfaite …
        - Cependant, il y a une logique à avoir, il faut prendre une fréquence de coupure raisonnable (120Hz sur l’image) pour que le filtre ait « le temps » de filtrer jusqu’à 500Hz
        - Si on donne une fréquence de coupure trop grande (trop proche de la fréquence à filtré), le filtre n’aura « pas le temps » de faire son travail et ne va atténuer que partiellement le 500Hz



    - Ordre :
        - Plus l’ordre est élevé, plus la pente du filtre sera raide
        - On peut alors se dire qu’il suffit de mettre un ordre très élevé pour que le filtre « s’applique très vite » … mais il faut faire attention que plus l’ordre est élevé, plus le temps de traitement sera long (et demandera plus de matériel pour filtrer dans la vrai vie)



Propriétés :

- Tout filtre, réalisable et physique, déphase et atténue.
- Un filtre idéal qui couperait une bande de fréquence données (mur de briques) ne serait pas réalisable.
- Un filtre analogique, continu et réalisable est composé de résistances, d’inductances, de condensateurs et d’ampli op.
  Leur équivalent numérique c’est un quotient de polynôme (cf. img ci-dessous)
  Question d’examen : C’est quoi l’équivalent numérique d’un filtre analogique ? R : Un quotient de polynôme (l’image à coté)

Si $p = j\omega$, ils sont **caractérisés par**

$$H(p) = \frac{\sum_{i=0}^{m} b_i p^i}{\sum_{k=0}^{n} a_k p^k}$$

Paramètres d'un filtre (un filtre est caractérisé par…)

- Son gabarit
  - Passe-bas, passe-haut, passe-bande, coupe-bande

- Son type
  - Butterworth, Bessel, Chebychev
    (On utilise que Butterworth dans le cours)

- Sa fréquence de coupure f_c

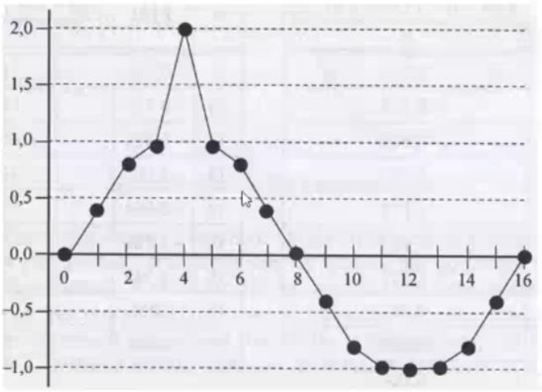
- Son ordre n

Chapitre 5 – Filtres numériques

1. Approche intuitive

- Filtre à moyenne mobile

  Prenons ce signal sinusoïdal dans lequel se trouve un craquement …



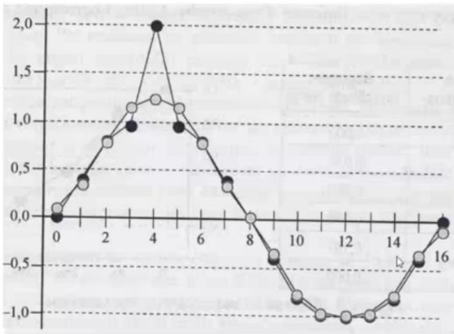
Appliquons-lui un algorithme calculant la valeur moyenne de part et d'autre du craquement.

$$g(n) = \frac{f(n-1) + f(n) + f(n+1)}{3}$$

De cette façon, le craquement ne disparaitra pas complétement mais son amplitude sera réduite (on atténue donc son effet sur le signal)

Du coup on obtient…

n	f(n)	g(n)	n	f(n)	g(n)
-1	0,000	0,000	9	-0,383	-0,363
0	0,000	0,128	10	-0,707	-0,671
1	0,383	0,363	11	-0,924	-0,877
2	0,707	0,671	12	-1,000	-0,949
3	0,924	1,210	13	-0,924	-0,877
4	2,000	1,283	14	-0,707	-0,671
5	0,924	1,210	15	-0,383	-0,363
6	0,707	0,671	16	0,000	-0,128
7	0,383	0,363	17	0,000	0,000
8	0,000	0,000			

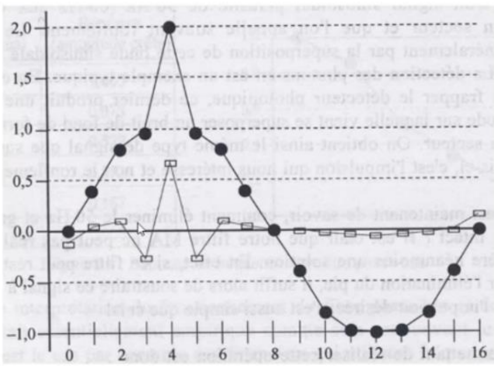


Ce type d'implémentation est l'équivalent numérique d'un filtre passe-bas.

- Filtre différenciateur

  Il suffit de soustraire le signal reconstitué par le filtre précédent au signal lui-même.

$$g(n) = f(n) - \frac{f(n-1) + f(n) + f(n+1)}{3}$$



On obtient alors ceci.

Ce type d'implémentation est l'équivalent numérique d'un filtre passe-haut puisqu'il ne laisse passer que les hautes fréquences.

- En pratique, les filtres utilisé sont beaucoup plus sophistiqués (voilà pourquoi on utilise Matlab). Leur fonction de transfert peut s'écrire :

$$H(z) = \frac{B(z)}{A(z)} = \frac{b(1) + b(2)z^{-1} + \cdots + b(n+1)z^{-n}}{1 + a(2)z^{-1} + \cdots + a(n+1)z^{-n}}$$

(Notez que les coefficients s'appellent A et B, ce qui correspond aux valeurs de retours de l'instruction butter() dans Matlab)

Dans Matlab pour filtrer un signal, on utilise les instructions

```
[b,a]=butter(n,Wn,'ftype')
filter(b,a,s)
```

butter signifie qu'il s'agit d'un filtre de Butterworth
n est l'ordre du filtre
ftype est son type (par défaut passe-bas)
s est le signal à filtrer
Wn est la fréquence de coupure normalisée qui $= 1$ si $F_c = F_s/2$
On aura donc $Wn = \frac{F_c}{F_s/2}$
Exemple: On désire filtrer un signal *s* avec un *filtre passe-bas* de *Butterworth* d'ordre *2*, avec *Fc = 400* et *Fs = 2000*

```
»[b,a]=butter(2,400/1000)
»y=filter(b,a,s)
```

2. Principe de causalité

- Le principe de causalité suggère que le fonctionnement d'un filtre s'accomplisse en temps réel.

Si on revient sur le filtre passe-bas à moyenne mobile et qu'on regarde les indices des échantillons utilisés dans son expression, on voit qu'on utilise des échantillons d'indice n-1 et/ou n+1. Vu que l'indice n correspond à l'échantillon courant (celui qui vient immédiatement), un problème apparaît sur la signification de l'échantillon n+1 de l'équation car cet échantillon est en fait la valeur d'un échantillon futur.

Si on suppose un signal enregistré (donc dont les valeurs futures sont accessibles en mémoire), il n'y a aucun souci.
Par contre, pour les applications en temps-réel, on est contraint d'utiliser uniquement les valeurs courantes et passées....on n'a pas accès aux valeurs futures.

- Les filtres basés sur les valeurs présentes et passés sont appelés des filtres causaux.

- Les filtres nécessitant des valeurs futures sont appelés des filtres non-causaux. Ces filtres ne peuvent être construits que de façon numérique puisqu'il suffit d'aller en mémoire pour connaître la valeur future.

- Tout filtre du monde réel est obligatoirement causal, puisqu'on ne connait pas le futur.

Un filtre, pour être causal donc réalisable, ne peut donc se baser que sur des valeurs présentes et passés.

Chapitre 6 – Echantillonnage

1. Échantillonnage idéal

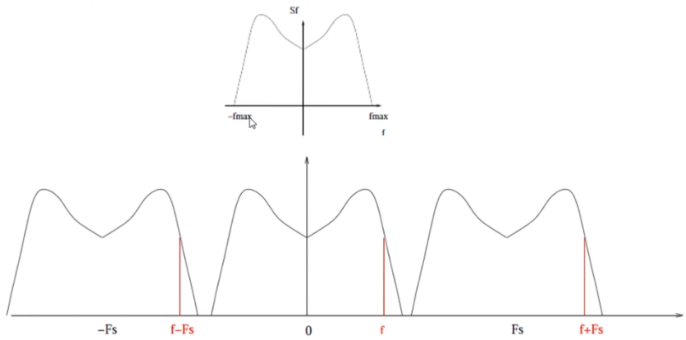
Echantillonnage : faire une découpe temporelle d'un signal continu
Période d'échantillonnage T_s : intervalle de temps entre 2 découpes
Fréquence d'échantillonnage : inverse de la période d'échantillonnage $F_s = \frac{1}{T_s}$

- Echantillonner un signal, c'est le multiplier par un peigne de Dirac
    - Un « peigne de Dirac » c'est une somme de distribution de Dirac espacé à intervalles réguliers correspondant à la période d'échantillonnage T_s.
    - Le poids de chaque distribution correspond à la valeur du signal à l'instant où elle se trouve
    - La Transformée de Fourier d'un peigne de Dirac c'est un peigne de Dirac
- Propriété : le spectre d'un signal échantillonné est périodique de période F_s

$$s_e(t) = s(t) \cdot \mathrm{Pgn}_{T_e}(t)$$

- Pour savoir si le signal échantillonné contient bien la même info que le signal initial, on va comparer les spectres des deux signaux

- Paroles du prof :
    - L'idée c'est de dire « j'ai un signal temporel » et j'en fait la transformée de Fourier, j'obtiens alors son spectre borné. Quand je vais échantillonner le signal, le spectre devient « le motif de base du spectre et le même motif apparait aussi à $\pm F_s$, $\pm 2F_s$, $\pm 3F_s$ … ». Et donc le spectre initialement borné devient non borné à cause de l'échantillonnage.

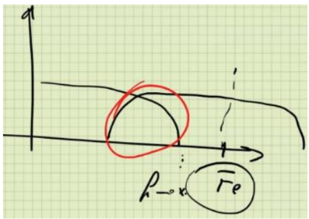


- L'opération d'échantillonnage rend un spectre borné … non borné

Théorème de Shannon-Nyquist : Lors d'un échantillonnage, pour ne pas déformer le motif de base du spectre du signal initial, il faut et il suffit que $F_s \geq 2.f_{max}$

($2.f_{max}$ = 2 fois la plus grande fréquence du signal initial)

Si on ne respecte cette règle, le motif vont se superposer (on aura un repliement de spectre)

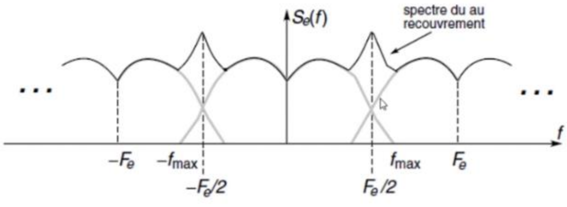


- Pour récupérer le signal initial à partir du signal échantillonné, il suffit alors d'isoler un des motifs grâce à un filtre passe-bas idéal dont $f_c = {F_s}/{2}$

2. Repliement de spectre

- Ce produit lorsqu'on ne respecte pas le théorème de Shannon-Nyquist (Fs < 2fmax)

- Une fréquence se trouvant dans la zone de repliement est susceptible d'appartenir à la fois au spectre de base du signal initial et à son spectre image décalé de $\pm F_e$



Source: Cottet F., « Traitement des signaux et acquisition des données » 2ème édition. Dunod 2002

- En gros on <u>modifie l'information</u> alors qu'on ne devrait pas …

- <u>Fréquence fantôme</u> : <u>fréquence qui n'existait pas dans le spectre d'origine mais qui y est apparue car on n'a pas respecté Shannon-Nyquist</u>
-
- Pour éviter tout ça …
  - Si on peut toucher à la fréquence d'échantillonnage, on fait en sorte de respecter Shannon-Nyquist
  - Si on ne peut pas toucher à la fréquence d'échantillonnage, on doit limiter le spectre du signal de base pour ne garder qu'une partie de l'information de départ

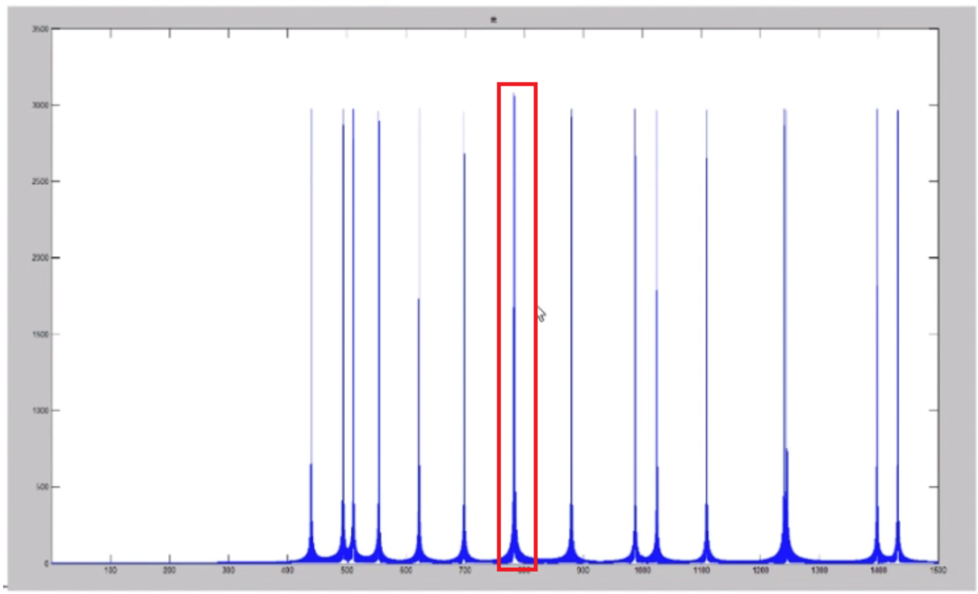
3. Exemples et exercices (question d'examens)

- Si on écoute un « la » de la 3e octave (f = 440Hz) avec une fréquence d'échantillonnage Fs de 600Hz, quelles fréquences va-t-on entendre ?

f	f+Fs	f-Fs	f+2Fs	f-2Fs
440	1040	-160	1640	-760
-440	160	-1040	760	-1640

- Quelles fréquences va-t-on entendre si on écoute une succession de notes commençant avec un « la » de la 3e octave sachant que la fréquence d'échantillonnage vaut 3000Hz. La fréquence suivante est obtenue en faisant $f = f_{-1} * \sqrt[6]{2}$
    - Comme on échantillonne à 3000Hz, jusqu'à f = 1500Hz, on respecte le théorème de Shannon-Nyquist.
    - Une fois qu'on dépasse f = 1500Hz, on voit l'apparition d'une fréquence fantôme (regardez le cadre rouge, la valeur de la colonne Fs-f est inférieure à la valeur de f donc on va l'entendre)
    - Dans le spectre, on voit (cadre rouge pour l'exemple) que certaines « barres de fréquences » sont plus épaisses que la barre de base à 440Hz … c'est pcq on a la superposition de la fréquence de base avec la fréquence fantôme

f	-f	Fs-f	f-Fs	2Fs-f
440,00	-440,00	2560,00	-2560,00	5560,00
493,88	-493,88	2506,12	-2506,12	5506,12
554,37	-554,37	2445,63	-2445,63	5445,63
622,25	-622,25	2377,75	-2377,75	5377,75
698,46	-698,46	2301,54	-2301,54	5301,54
783,99	-783,99	2216,01	-2216,01	5216,01
880,00	-880,00	2120,00	-2120,00	5120,00
987,77	-987,77	2012,23	-2012,23	5012,23
1108,73	-1108,73	1891,27	-1891,27	4891,27
1244,51	-1244,51	1755,49	-1755,49	4755,49
1396,91	-1396,91	1603,09	-1603,09	4603,09
1567,98	-1567,98	1432,02	-1432,02	4432,02
1760,00	-1760,00	1240,00	-1240,00	4240,00
1975,53	-1975,53	1024,47	-1024,47	4024,47
2217,46	-2217,46	782,54	-782,54	3782,54
2489,02	-2489,02	510,98	-510,98	3510,98
2793,83	-2793,83	206,17	-206,17	3206,17
3135,96	-3135,96	-135,96	135,96	2864,04
3520,00	-3520,00	-520,00	520,00	2480,00



Chapitre 7 – Analyse spectrale des signaux discrets

1. Transformée de Fourier rapide (FFT)

- Série d’algorithmes permettant de minimiser/réduire le nombre d’opérations nécessaire pour calculer la transformée de Fourier

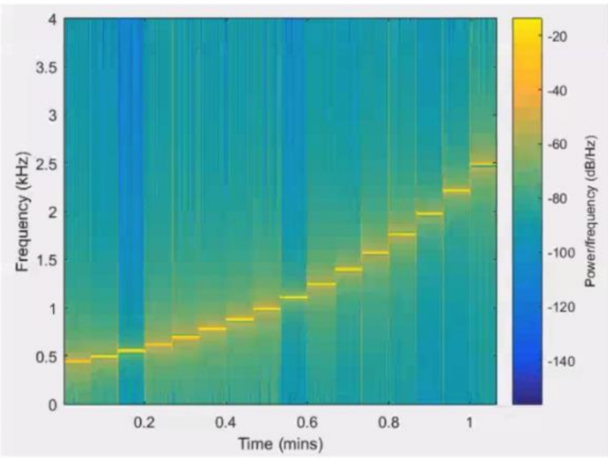
- Sous Matlab on utilise les instructions :
  - fft() pour calculer la transformée de Fourier d’un signal
  - fftshift() pour recentrer d’un point de vue fréquentiel le signal que l’on veut représenter
  - ifft() pour calculer la transformée de Fourier inverse

2. Spectrogramme

- Permet de calculer et visualiser l’évolution du spectre d’un signal dans le temps.

- Se base sur une STFT (Short Time Fourier Transform), l’idée c’est de considérer une fenêtre glissante qui va parcourir le signal dans le temps et à chaque position de la fenêtre elle va en calculer une transformée de Fourier.

```
spectrogram(s,1024, [],512, Fs,'yaxis');
```



- L’instruction spectrogram() de Matlab prend plusieurs paramètres :
  - s => nom du signal s
  - Fs => fréquence d’échantillonnage Fs
  - 1024 => nombre d’échantillons de la NFFT

    - on peut également utiliser une fenêtre temporelle afin de minimiser l’apparition de fréquences non-désirées dues au fenêtrage rectangulaire. Par défaut, Matlab utilisera une fenêtre de Hamming
    - un recouvrement des intervalles de calcul de la FFT (“overlap”) permettra une plus grande précision des calculs. Par défaut, Matlab choisira un overlap de 50%.

- Dualité temps-fréquence :
  - Si on essaie d'avoir une meilleure précision fréquentielle, on va perdre en précision temporelle
  - Si on essaie d'avoir une meilleure précision temporelle, on va perdre en précision fréquentielle

Le temps et la fréquence sont 2 grandeurs conjuguées et il y aura toujours un compromis à accepter entre résolution temporelle et résolution fréquentielle.

Chapitre 8 – Introduction au traitement d'images

1. Niveau de gris

- A chaque pixel correspond une valeur qui est traduite par un niveau de gris qui se trouvent entre le blanc et le noir.

- Sur 8 bits, la valeur se trouve entre 0 (noir) et 255 (blanc) … on a 256 intensités de gris (plus ou moins claires selon la valeur)

- Ont converti souvent les images couleurs en image en niveau de gris pour leur appliquer un traitement par la suite

2. Inversion

- L'inversion c'est <u>créer un négatif de l'image</u>
  - Dans un négatif, le blanc devient noir et le noir devient blanc

- Cette fonction est représentée par l'expression $p(i) = (2^k - 1) - p(i)$
  - Chaque pixel de l'image $p(i)$ est remplacé par une valeur égale à $(2^k - 1)$ moins la valeur actuelle du pixel.

    K représente le nombre de bits avec lequel on travail … 8bits par exemple

1.Image de base	2.On la met en niveau de gris	3.On applique la formule

3. Filtre de bruit par moyenne et filtre de bruit par médiane

- <u>Corrigent les images bruitées</u>

- Dans une image, un <u>bruit c'est des hautes-fréquences</u> càd que l'on a de fortes variations de couleurs ou de niveaux de gris entre des pixels qui sont les uns à côtés des autres.

- Ces 2 filtres sont des filtres passe-bas, ils vont donc adoucir les variations de valeurs entre les pixels.

- <u>Ceci a aussi pour effet de flouté l'image</u> (puisqu'on réduit les variations entre les pixels)

On peut voir sur l'image ci-contre que plus on applique un filtre, plus l'image devient floue. En effet, le 1ere image est nette mais bruitée alors que la dernière est moins nette mais bien moins bruitée.

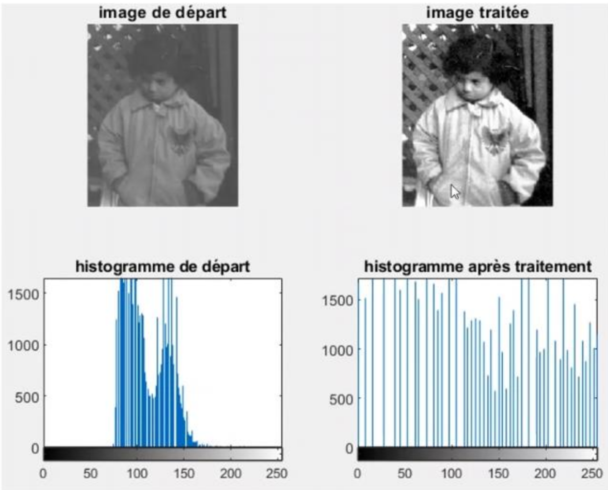
4. Détection de contours

- Repérer les grandes variations de couleur entre des pixels proches.

- Il existe plusieurs méthodes comme gradient, Roberts, Prewitt, Sobel, Freishen, Canny, …
  Les méthodes Sobel et Canny sont très utilisées (on les utilise à un moment en TP Matlab)

5. Augmentation de contraste

- Pour améliorer le contraste, on va équilibrer l'histogramme de l'image



Histogramme de départ : on voit qu'il y a beaucoup de valeurs entre 75 et 175 (environ) et que le reste des valeurs possibles est très, voir pas du tout, utilisé.

Histogramme après traitement : Après avoir utilisé la fonction histeq() de Matlab sur l'image, on voit que les valeurs de l'histogramme on était étalées sur l'histogramme sur toutes les valeurs possibles entre 0 et 255

6. Binarisation

- Transformé un pixel codé sur plusieurs bits en pixel codé sur 1 bit.

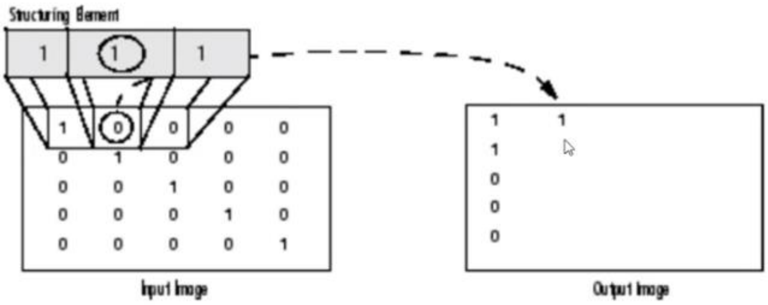
- On passe sur tous les pixels de l'image … si sa valeur est inférieure à un seuil fixé sa nouvelle valeur sera 0 … si sa valeur est supérieure ou égale au seuil sa valeur sera 1

7. Érosion/Dilatation

- L'état de sortie de chaque pixel est déterminé en appliquant un traitement à lui et ses voisins.

- Dilatation : la valeur du pixel de sortie est la valeur max de tous les pixels dans le voisinage du pixel

- Erosion : la valeur du pixel de sortie est la valeur min de tous les pixels dans le voisinage du pixel



- La forme et la taille du voisinage sont déterminé par un élément structurant

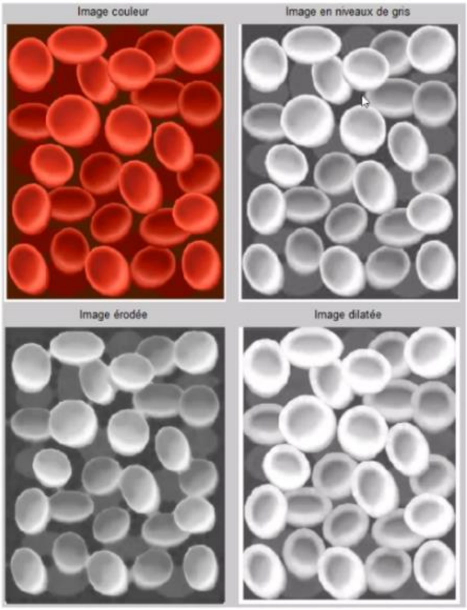
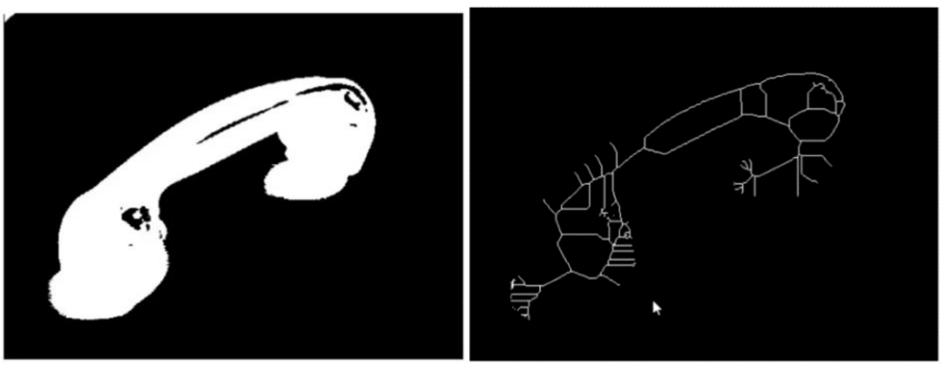


Image érodée : on voit que les globules ont été « mangés » (un peu comme les bords des rivières avec l'érosion)

Image dilatée : on voit que les globules ont été grossis (un peu comme la dilatation du métal)

8. Squelettisation

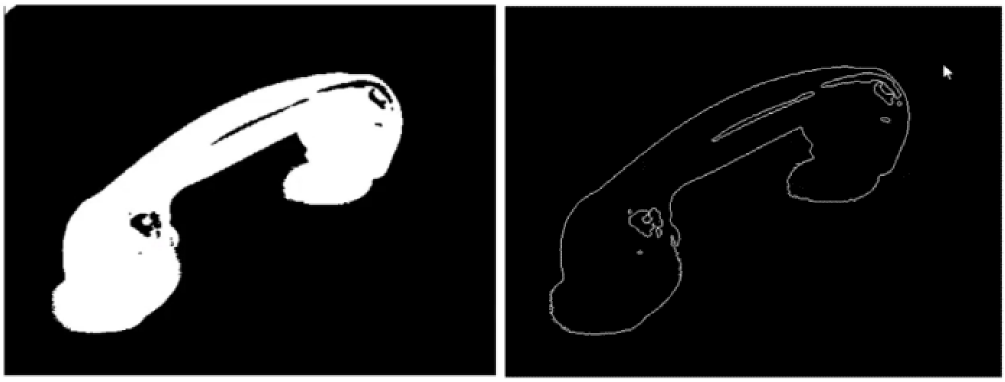
- Permet de réduit tous les objets d'une image à des lignes sans changer la structure essentielle de l'image. Cette opération est en fait basée sur les fonctions de dilatation et d'érosion.



(Ici on a une image d'un téléphone à droite qui a été squelettisée à gauche … oui le squelette obtenu est chelou quoi …)

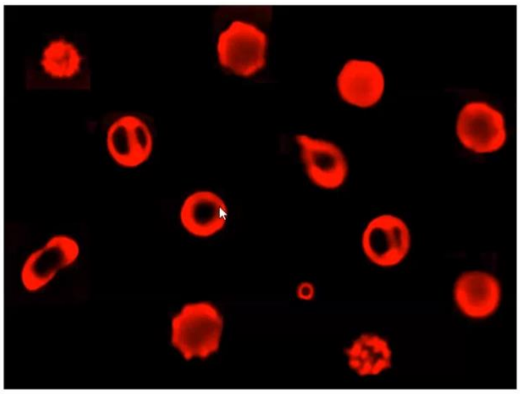
9. Détermination de périmètre

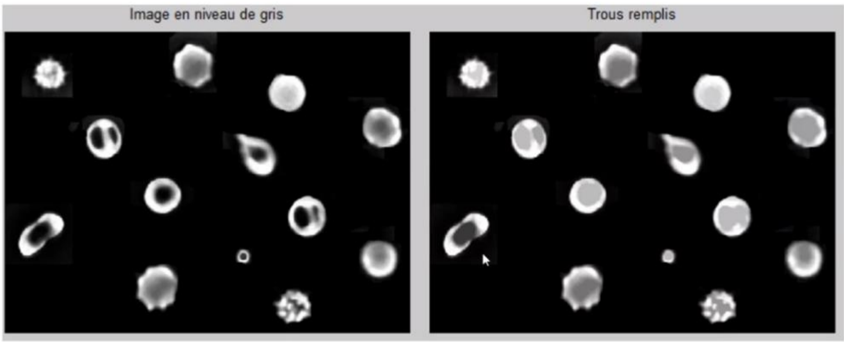
- Semblable à la détection de bords mais <u>marque cette fois les contours</u>.



10. Remplissage de trous

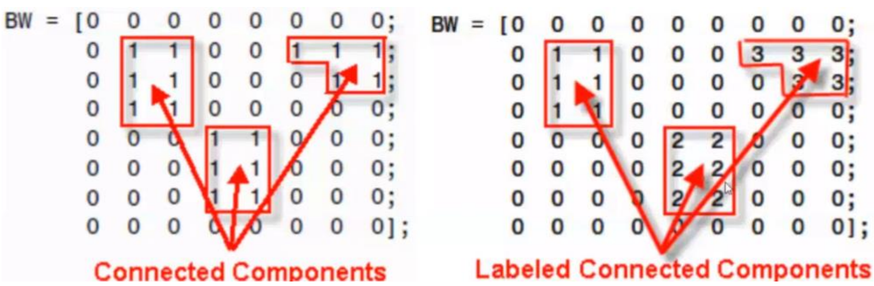
```
clear all; close all;
filename = 'blood2.tif';
x = imread(filename);

if (length(size(x))>2)
    x=x(:,:,1);
end

y = imfill(x,'holes');

subplot(131)
imshow(x)
title('Image en niveau de gris')
subplot(132)
imshow(y)
title('Trous remplis')
```





11. Labellisation

- Méthode qui va <u>identifier des objets (groupements de 1)</u> présents dans une image binaire. Chaque objet est identifié par un numéro unique.



Chapitre 9 – Introduction au Machine Learning

1. Machine Learning

- Apprendre aux ordinateurs ce qui est naturel chez l'humain
  ⇨ Apprendre à partir d'expériences

- On utilise le ML quand on doit réaliser une tâche complexe ou un problème impliquant une grande quantité de données mais qu'il n'y a pas de formule ou équation mathématiques pour nous aider

- Associé au Big Data, le ML permet de résoudre des problèmes dans de nombreux domaines tel que la finance (trading, …), le traitement d'images (reconnaissance faciale, …), production d'énergie (prédiction de la charge réseau à une heure h, …), maintenance prédictive, …

- Pour entrainer les modèles à résoudre ces problèmes, on utilisera soit l'apprentissage supervisé (AS) soit l'apprentissage non-supervisé (ANS)

2. Apprentissage Supervisé (AS)

- Entrainer le modèle avec des entrées et des sorties connues de manière à pouvoir prédire les résultats futurs
  - « Si tu as cette entrée, tu dois avoir cette réponse »

- Son objectif est de construire un modèle de prédiction basées sur des preuves tout ça, face à un certain degré d'incertitude.

- 2 grandes techniques :
  - Classification : présente des réponses discrètes. Les données d'entrées sont classifiées en différentes catégories.
  - Régression : présente des réponses continues. On étudie la relation des données d'entrées par rapport à d'autres données.

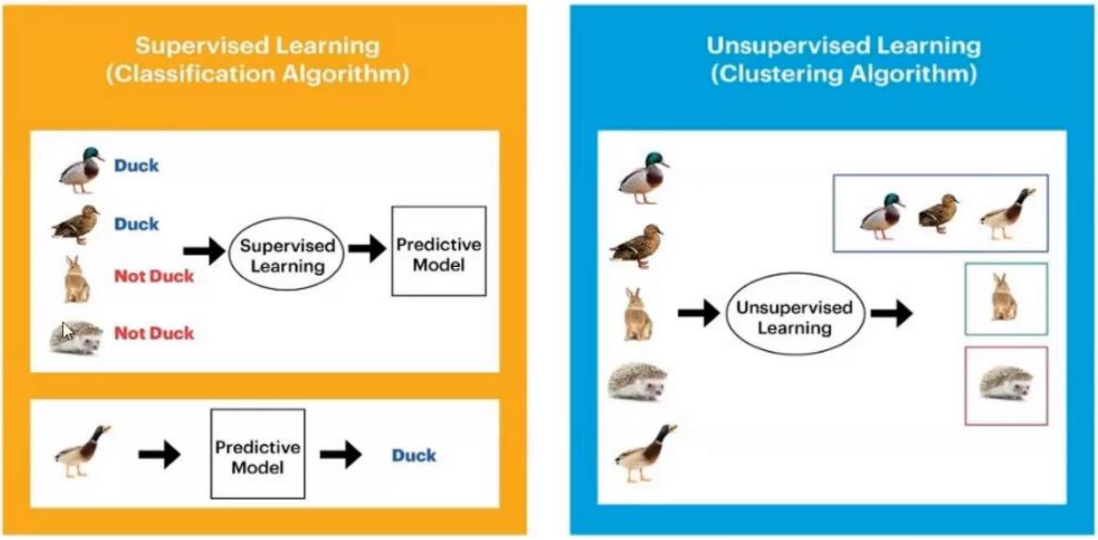
3. Apprentissage Non-Supervisé (ANS)

- Trouver des motifs cachés ou des structures dans un ensemble de données
  - « Prends ces données et démerdes-toi pour me faire des groupes »

- Technique la plus utilisée :
  - Clustering : on sépare les données en un ensemble de groupes de sorte que dans chaque groupe les données se ressemblent davantage qu'à celles des autres groupes

- 2 grands groupes d'algo de clustering :
  - Hard clustering : chaque donnée n'appartient qu'à un cluster
  - Soft clustering : chaque donnée peut appartenir à plusieurs clusters

4. Comparaison AS et ANS



Le modèle prédictif est obtenu en disant « ça c'est un truc, ça c'est un machin et ça c'est un bidule » On donne alors de nouvelles données au modèle prédictif et on voit s'il nous donne la bonne réponse (le canard quoi …)	On lui dit « Démerdes-toi pour me faire des groupes »

- L'AS est plus complexe à mettre en place que l'ANS mais il produit des résultats plus précis

5. Quel algo choisir ?

- Selon la taille et le type de données
  (Y a un gros tableau qui montre des exemples mais on s'en fout)