



DIRO
IFT 3205

TRAVAIL PRATIQUE

Interpolation - Extrapolation

Max Mignotte

DIRO, Département d'Informatique et de Recherche Opérationnelle.

[http : //www.iro.umontreal.ca/~mignotte/ift3205](http://www.iro.umontreal.ca/~mignotte/ift3205)

e-mail : mignotte@iro.umontreal.ca

1 Introduction

L'interpolation et respectivement l'extrapolation sont des techniques très couramment utilisées en traitement du signal pour le rééchantillonnage et le calcul de données supplémentaires afin d'augmenter la résolution, agrandir, restaurer, reconstruire ou réaliser une transformation géométrique sur une image ou augmenter la qualité d'un son, etc. Sous la forme la plus élémentaire, l'interpolation permet d'estimer la valeur d'un signal f en x en fonction de ses échantillons passés et futurs et l'extrapolation en fonction seulement de ses échantillons passés. Une méthode d'interpolation à la fois simple et relativement efficace est l'interpolation d'ordre 2 ou bi-linéaire (c'est dire linéaire dans les deux dimensions de l'image et que l'on pourrait créer facilement à partir de trois interpolations linéaires successives). Cette dernière est meilleure qu'une simple interpolation utilisant le voisin le plus proche (dite d'ordre 0), introduisant des effets indésirables de crénelage. Dans le même style, il existe aussi des méthodes d'interpolations d'ordre supérieur, comme l'interpolation bicubique ou par spline (ordre 3).

La notion de noyau permet de généraliser l'interpolation pour les différents ordres existants. Le noyau est simplement la fonction que l'on multiplie avec les pixels connus de l'image et qui réalise l'interpolation. Mathématiquement, l'interpolation ou le signal interpolé y consiste à convoluer le noyau n choisi avec le signal x à interpoler ($y = x * n$). Plus ce noyau à un support large, i.e., plus ce noyau utilise beaucoup d'échantillons passés et futurs du signal, plus l'interpolation est de bonne qualité. L'interpolation idéale est d'ordre infini et cette interpolation dite idéale est l'interpolation spectrale réalisée grâce à la TF.

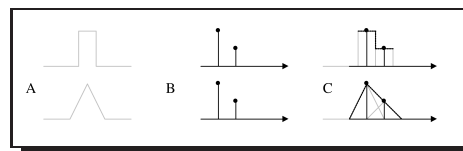


FIGURE 1 – De gauche à droite, noyaux d'interpolation, signal à interpoler et signal interpolé

2 Interpolation spectrale

Prenons l'exemple d'une interpolation de facteur 2. Pour un signal, cela consiste à créer un signal avec une résolution deux fois plus grande (i.e., avec deux fois plus d'échantillons). Pour ce faire, on va chercher à doubler la fréquence d'échantillonnage f_e et conserver les fréquences originales du signal, i.e., ses composantes fréquentielles comprises entre $[-f_m ; f_m]$. Cette opération qui consiste à conserver les fréquences originales du signal correspond, dans le domaine fréquentiel, à multiplier le

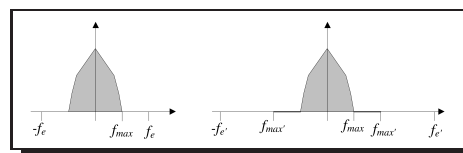


FIGURE 2 – Spectre du signal original et de celui utilisant deux fois plus d'échantillons

spectre du signal $S(f)$ par la fonction porte $\Pi_{\frac{1}{2f_m}}(f)$ centrée en $f = 0$ et de largeur $2f_m$. Cette opération dans le domaine spatial correspond donc à la convolution des échantillons originaux du signal avec une fonction "sinc". Pour réaliser, de façon équivalente, cette opération d'interpolation dans le domaine spatial, il faudrait convoluer les échantillons du signal de départ avec un noyau d'interpolation représenté par la "sinc" qui a un support (i.e., un espace de définition) infini. Par conséquent, il faudrait réaliser une infinité de combinaisons linéaires (pixels existants multipliés par le noyau "sinc") pour obtenir l'interpolation optimale. Étant donné que le nombre de pixels par ligne est limité aux pixels de l'image à interpoler, il est clair que la combinaison linéaire ne sera pas complète. L'image résultante aura donc des oscillations dues aux oscillations de grande amplitude du noyau "sinc". Une façon simple de réaliser cette interpolation optimale est donc de calculer la TFD du signal, de remplir de zéros le spectre (zéros padding) et de faire la TFD inverse.

1. À partir de l'image LENA.PGM de taille 128×128 , agrandir cette image par un facteur 4 avec
 - (a) une interpolation au plus proche voisin.
 - (b) avec une interpolation bi-linéaire¹.
 - (c) avec une interpolation spectrale (i.e., une interpolation idéale utilisant la TFD). Comparer avec le résultat précédent.
2. Expliquer comment (i.e., à partir de quel théorème sur la TF vu en cours) on peut prévoir à l'avance (c'est-à-dire calculer) le facteur de normalisation indispensable pour retrouver une image agrandie avec la même intensité moyenne que l'image quatre fois plus petite.
3. À partir de l'image LENA.PGM de taille 128×128 et sans utiliser/calculer l'image LENA.PGM agrandie par un facteur 4 et calculé précédemment. Définir une stratégie pour augmenter d'un facteur 4 le spectre de cette image.

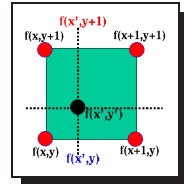
3 Interpolation spectrale par seuillage

On va maintenant utiliser une méthode qui utilise toujours la TFD mais aussi la notion de seuillage dur (*hard thresholding*) pour faire une interpolation qui permettra de restaurer une image dégradée par la présence de trous, de *scratches* ou encore de texte imprimé sur celle-ci comme l'image BARBARA.PGM de taille 512×512 que vous allez utiliser dans ce TP (cf. Fig. 3). La procédure de restauration que nous allons utiliser est itérative et consiste à alterner successivement des TFD de taille de plus en plus petite encerclant la dégradation avec un seuillage dur² sur le module de la TFD. Comme on ne dispose qu'une FFT dans cette question, qui ne marche que pour des longueur et largeur d'image qui sont des puissance de deux et pour simplifier le problème, on vous demande de procéder comme suit :

1. la première itération consiste à filtrer toute l'image :
 - (a) En prenant la TFD 512×512 de l'image dégradée.
 - (b) En utilisant un seuillage dur sur le module de celle-ci, avec un seuil égal à $\lambda = 4\%$ (et ensuite décroissant) de la valeur maximale de ce spectre (la valeur absolue en $[0][0]$ de la partie réelle).
 - (c) En prenant la TFD inverse de cette image filtrée.

-
1. On rappelle les formules d'interpolation bilinéaire

$$\begin{aligned}
 f(x', y) &= f(x, y) + (x' - x)[f(x + 1, y) - f(x, y)] \\
 f(x', y + 1) &= f(x, y + 1) + (x' - x)[f(x + 1, y + 1) - f(x, y + 1)] \\
 f(x', y') &= f(x', y) + (y' - y)[f(x', y + 1) - f(x', y)]
 \end{aligned}$$



2. On rappelle qu'un seuillage dur consiste à annuler tous les coefficients de la TFD qui sont en module inférieurs à ce seuil.

On combine ensuite l'image filtrée précédente et l'image dégradée en remplaçant les valeurs de pixels situées sur la dégradation de l'image (ceux dont la position est associée à un niveaux de gris égale à du blanc pur sur l'image dégradée) par les niveaux de gris de l'image filtrée.

2. On répète ensuite l'opération précédente en prenant une TFD de taille $[256 \times 256]$ différemment centrée sur la dégradation avec un seuillage dur deux fois plus petit à chaque fois. Pour obtenir les résultats montrés à la Fig. 3, j'ai pris une TFD de taille $[256 \times 256]$ partant (pixel le plus à gauche et haut) respectivement sur [ligne = 128 col = 128] avec $\lambda = 2\%$ (2ième itération), partant de [ligne = 240 col = 128] avec $\lambda = 1\%$ (3ième itération), [ligne = 168 col = 128] avec $\lambda = 0.5\%$ (4ième itération), [ligne = 200 col = 128] avec $\lambda = 0.25\%$ (5ième itération).



FIGURE 3 – Image dégradée et résultat de la restauration à la 1ière et 5ième itération

3. Expliquer qualitativement pourquoi cette méthode d'interpolation marche.

4 Extrapolation spectrale par seuillage

La méthode d'interpolation par seuillage précédente aurait été plus efficace si on avait mis en place une série de filtrages (i.e., TFD + seuillage dur + TFD inverse) avec toujours un seuil dur décroissant mais

surtout avec une TFD de taille de plus en plus petite encerclant la dégradation. Nous n'avons pas utilisé cette stratégie précédemment à cause du fait que la dégradation était grande et que nous disposions pour cette question que d'une FFT qui ne pouvait fonctionner qu'avec des longueurs et largeurs d'image qui sont des multiples de la puissance 2 (256×256 était ainsi la plus petite FFT encerclant la dégradation affectant l'image précédente). Pour cette question on dispose d'une Transformée de Fourier Discrète (TFD) qui peut être utilisée quelle que soit la taille de l'image mais qui, en contre-partie, est beaucoup plus lente. On vous demande dans cette question de

1. Définir une stratégie et l'implémenter, utilisant la TFD permettant d'extrapoler le haut (i.e., les 30 ou 50 lignes suivantes) de l'image (de taille 256×256) MONTROYALPARC.PGM pour trouver un résultat d'extrapolation comparable (ou meilleur) que celui montré en Fig. 4.

NOTA : On vous conseille de

- (a) remplir initialement chaque ligne à interpoler avec la moyenne des 30 dernières lignes de chaque colonne de l'image (plutôt que du noir).
- (b) de prendre initialement pour l'image filtrée, une image dans laquelle il n'y a que le ciel et les arbres.
- (c) faire une procédure itérative (avec plusieurs itérations) et à chaque itération de mettre tous les pixels dont le niveau de gris est négatif par 0 et ceux dont le niveau de gris est supérieur à 255 par 255.



FIGURE 4 – Image originale et image dont les 30 lignes du haut ont été extrapolées

Nota :

L'extrapolation pour les images (ou signaux 2D) reste ludique et sans application utile ; Néanmoins en traitement du signal 1D, cette extrapolation est utile pour la prédiction de trajectoires, des cours boursiers, d'un signal quelconque, l'évolution de la démographie, etc.



FIGURE 5 – *Image originales et image extrapolées*



FIGURE 6 – *Image originale et image extrapolée*

Remise & Rapport

Vous devez rendre physiquement au démonstrateur le rapport et électroniquement le(s) programme(s) fait en C avant la date de remise spécifiée dans le fichier *barème* situé sur la page web du cours. Pour la remise électronique, utilisez le programme *remise* (*man remise* pour plus de détails) pour remettre votre code dans le répertoire TP<Numéro du Tp>. N'oubliez pas d'inscrire vos noms, courrier électronique en commentaire en haut du fichier .c remis. Les noms des programmes à remettre devront avoir le format suivant *Tp3-IFT3205-<Numéro de la question>.c*. Les programmes devront se compiler et s'exécuter sur Linux tel qu'indiqué dans le barème.

Le rapport (si celui-ci est demandé dans le fichier barème) doit être concis, clair et devra contenir (en plus des éléments et des discussions que vous jugerez importants), la description brève du problème et de la technique utilisée (lorsque cela est nécessaire ou demandé), les réponses aux questions posées, les images résultats et les commentaires que l'on vous demande et que vous jugerez utile.