École Normale Supérieure Traitement du signal
26 mars 2015

TD n°6 - Spectrogrammes

Trois signaux audio et le code d'une fonction `rep_shift` sont disponibles à l'adresse : `http://www.di.ens.fr/~waldspur/tds/14_15_s2/td6.html`.

1. Calcul du spectrogramme d'un signal

a) Charger `gemshorn.wav` dans un signal `f` et sa fréquence d'échantillonage dans une variable `freq`.

> Utiliser la commande `[f,freq]=wavread(filename)`.

La fréquence d'échantillonage est exprimée en Herz. Une fréquence de 20000 Hz, par exemple, indique qu'un segment de 20000 points représente une seconde de son.
b) On va utiliser pour le calcul du spectrogramme des fenêtres de longueur 30ms, chaque fenêtre étant décalée de 5ms par rapport à la précédente.
Calculer en fonction de `freq` à quelle taille de signal ces valeurs correspondent. Les stocker dans des variables `win_size` et `step_size`.
c) Générer une fenêtre assez lisse de taille `win_size`.

> Plusieurs fonctions préimplémentées existent pour calculer des fenêtres, en particulier `hanning`.

d) Définir dans un fichier séparé une fonction `win_fourier`, qui accepte en entrée le signal `f`, la fréquence d'échantillonage `freq`, la fenêtre, la valeur de `step_size` et un paramètre de sur-échantillonage `ov` (qu'on pourra prendre égal à 5).
e) Dans la fonction `win_fourier`, commencer par remplacer le signal `f` (dont on notera N la taille) par le signal `g` tel que :

$$\mathtt{g}[k] = \mathtt{f}[k+1] - \mathtt{f}[k] \qquad \forall k = 0, ..., N-2$$

Quel est l'effet de ce remplacement sur la transformée de Fourier ?
f) Calculer `nb_win`, le nombre de fenêtres qu'on va considérer.
g) À l'aide de la fonction `rep_shift`, calculer la matrice de `nb_win` colonnes dont la colonne k contient les valeurs de `g` se trouvant dans le support de la k-ième fenêtre.
h) Multiplier chaque colonne de la matrice par la fenêtre puis calculer la FFT colonne par colonne, avec un facteur de sur-échantillonage égal à `ov`.

> Si `M` est une matrice de `m` lignes, la fonction `fft(M,m*ov,1)` calcule la FFT de `M` colonne par colonne avec un facteur de sur-échantillonage égal à `ov`.

i) Si h est un signal de taille N, de transformée de Fourier discrète égale à $\hat{h}$, alors la transformée de Fourier de h avec facteur de sur-échantillonage $a \in \mathbb{N}^*$ est un signal H de longueur Na tel que :

$$H[an] = \hat{h}[n] \quad \forall n = 0, ..., N-1$$

Les valeurs $H[m]$ pour m non multiple de a sont obtenues par interpolation des $H[an]$, ce qui donne un signal H plus « lisse » que $\hat{h}$.

Pour $k = 0, ..., \texttt{ov}\times\texttt{win\_size} - 1$, à quelle fréquence auditive correspond la ligne $k+1$ de la matrice calculée à la question précédente ?
j) Supprimer toutes les lignes qui ne correspondent pas à des fréquences comprises entre 0 et 5000 Hz. Faire renvoyer ce résultat à la fonction `win_fourier`.
k) Afficher la valeur absolue de la transformée de Fourier à fenêtre de `gemshorn.wav`.

> Pour afficher une image `M`, utiliser les instructions `figure (1) ; imagesc(M) ;`. On pourra appliquer la fonction `flipud` à la transformée de Fourier à fenêtre avant de l'afficher, de façon à ce que les basses fréquences soient en bas de la figure et les hautes fréquences en haut.

l) Écouter le signal `gemshorn.wav`. Le spectrogramme obtenu vous semble-t-il correspondre à ce que vous avez entendu ? Même question pour les signaux `coucou.wav` et `chirp.wav`.

2. Fréquence instantanée

a) Définir une fonction `inst_freq`, qui accepte en entrée une transformée de Fourier à fenêtre `S` et les paramètres qui ont servi au calcul de cette transformée (`freq`, la fenêtre, `step_size` et `ov`).
b) Imaginons que pour n autour d'une valeur $n_0.\texttt{step\_size}$, le signal f de la question 1. vaut à peu près :

$$\texttt{f}[n] = a \exp\left(2\pi i \nu \frac{n}{\texttt{freq}}\right)$$

(où a est un nombre complexe quelconque et ν un réel strictement positif)
Que vaut à peu près $\texttt{S}[k,n]$ pour n autour de n_0 et k quelconque ? Pour n fixé, quel est le k tel que $|\texttt{S[k,n]}|$ est maximal ?
[Indication : attention, il faut refaire le calcul par rapport au cours car on n'a pas défini la transformée de Fourier à fenêtre exactement de la même manière.]
c) Dans `inst_freq`, définir un tableau `S_inst` de même taille que `S` tel que $\texttt{S\_inst}[k,n]$ vaut 1 si :

$$|\texttt{S}[k,n]| > |\texttt{S}[k-1,n]| \quad \text{et} \quad |\texttt{S}[k,n]| > |\texttt{S}[k+1,n]|$$

et 0 sinon.

> Si `a` et `b` sont deux tableaux de même taille, alors `(a>b)` est le tableau dont la case $[l,m]$ vaut 1 si $\texttt{a}[l,m] > \texttt{b}[l,m]$ et 0 sinon.

d) Mettre à 0 les cases $[k,n]$ pour lesquelles $|\texttt{S}[k,n]|$ est inférieure à 2% de la valeur maximale de $|\texttt{S}|$.
e) Faire renvoyer à `inst_freq` ce résultat et l'afficher sur une autre figure que le spectrogramme. Que pensez-vous du résultat ?

> Utiliser `figure (2) ;` pour ouvrir une deuxième figure et `colormap('gray') ;` pour afficher en noir et blanc.

f) Si $\texttt{S}[k,n]$ correspond véritablement à une fréquence instantanée présente dans le signal `f`, que doit valoir à peu près $\texttt{S}[k,n]/\texttt{S}[k,n+1]$ (en fonction de k et n) ?
g) Utiliser ce résultat pour supprimer une partie des fausses détections de la question 2.e). Afficher le nouveau résultat sur une troisième figure.
h) Répéter la question précédente pour le signal `chirp.wav`. Commenter le résultat.