

TP : Débruitage des signaux audio (3 heures)

Remarque : les questions dont le numéro est suivi du signe † sont des questions théoriques, ne nécessitant pas l'utilisation de Matlab, qui doivent impérativement être résolues avant la séance de TP.

I. Débruitage avec référence de bruit

Le fichier `noisy_sig.wav` contient un signal audio bruité ($d=s+n_0$) enregistré avec un microphone. Le fichier `noise_ref.wav` contient une référence de bruit (n_1), mesurée par un autre microphone. Les bruits n_0 et n_1 sont corrélés (liés par une opération de filtrage). On veut exploiter cette corrélation pour estimer le signal utile.

1. † Rappeler le schéma d'un système de débruitage avec référence de bruit, vu en cours. Comment peut-on estimer la réponse en fréquence du filtre de débruitage à partir des signaux enregistrés par les deux microphones ?
2. Ecouter les 2 signaux. Peut-on entendre le signal utile ? Ecouter la différence des 2 signaux.
3. En utilisant les fonctions `xcorr`, `fft` et `ifft` de Matlab, estimer la réponse impulsionnelle du filtre de débruitage $w[n]$. Filtrer la référence de bruit avec ce filtre et en déduire une estimation du signal utile. Ecouter le résultat. Conclusion ?
4. Quelles sont les limites de cette méthode ?

II. Débruitage par soustraction spectrale

On considère maintenant une autre configuration où on ne dispose pas de référence de bruit. Nous avons donc un seul enregistrement bruité, noté $x=s+b$. Le bruit b est supposé stationnaire et indépendant du signal utile s . Nous voulons exploiter les zones de silence pour estimer la DSP du bruit, puis l'utiliser pour débruiter le signal enregistré.

1. † Rappeler le principe de la méthode de débruitage par soustraction spectrale.

Comme on l'a vu en cours, cette méthode génère elle-même un bruit résiduel appelé « bruit musical ». Dans le cas d'un enregistrement extrêmement bruité, comme celui considéré dans la partie I, ce bruit résiduel est tellement fort que la méthode n'a presque pas d'intérêt. Nous allons donc nous intéresser à un cas plus réaliste où le rapport du signal sur bruit est plus raisonnable.

2. Le fichier `noisy_sig_fan.wav` contient l'enregistrement d'un signal de parole, pollué par le bruit d'un sèche-cheveux. Ecouter le signal.

La fonction `decoupage.m`, mise à votre disposition, permet de fenêtrer un signal pour le découper en plusieurs tranches partiellement recouvrantes, et les mettre dans les colonnes d'une matrice.

3. Lire la fonction pour la comprendre, puis l'utiliser sur le signal bruité. Vous pouvez choisir une fenêtre de Hamming pour découper le signal en tranches de 25 msec (→ combien d'échantillons ?), avec un décalage de 40% entre deux fenêtres successives.

En écoutant le signal, on peut constater qu'il y a au moins une demi-seconde de silence au début de l'enregistrement. Les tranches de cette première partie peuvent donc être utilisées pour obtenir une estimation du spectre du bruit.

4. Partant de ce constat, réaliser la méthode de débruitage. La fonction `reconstruction.m`, mise à votre disposition, permet de reconstruire un signal temporel à partir d'une matrice qui contient ses tranches recouvrantes. Ecouter le résultat du débruitage. Conclusion.

Pour réduire le bruit résiduel, plusieurs post-traitements peuvent être appliqués au résultat obtenu. Plus précisément, ces post-traitements seront appliqués à la matrice contenant le spectre du signal débruité, obtenue précédemment.

Si le débruitage était parfait, la valeur du spectre du signal débruité dans les zones de silence serait nulle. En pratique, ce n'est pas le cas : ces zones de silence contiennent encore un bruit résiduel.

5. Pour chaque fréquence, calculer le niveau maximum du spectre du bruit résiduel sur les zones de silence déjà identifiées (au début de l'enregistrement). Cette quantité sera notée $max_NR(f)$ dans la suite. L'ensemble de ces valeurs maximales pour toutes les fréquences sera stocké dans un tableau.
6. Pour chaque fréquence de chaque tranche, si le niveau du spectre du signal débruité est inférieur à $max_NR(f)$, on le remplace par le minimum du spectre sur 3 tranches :

$$|\hat{S}_i(f)| = \min(|\hat{S}_j(f)|, \quad j = i-1, i, i+1) \quad \text{si } |\hat{S}_i(f)| < max_NR(f)$$

7. Pour chaque tranche, si

$$20 \log_{10} \left(\text{mean}(|\hat{S}_i(f)| / |\overline{B(f)}|) \right) < -12$$

on remplace le module du spectre du signal débruité par 3 pourcents du module du spectre du signal observé.

8. Après avoir appliqué ces post-traitements au spectre du signal débruité, continuer avec les dernières étapes de la méthode d'origine pour retrouver un signal temporel débruité. Ecouter le résultat. Conclusion ?