

GBM8770 - Traitement de signaux et d'images

TP1 - Filtrage des signaux et qualité des images

Automne 2019

Professeure : Farida Cheriet

Chargés de laboratoire : Gabriel Lepetit-Aimon, Clément Playout

Objectifs :

Ce laboratoire se compose de deux parties. La première partie porte sur le filtrage spatial des images et la seconde partie porte sur le filtrage fréquentiel des images.

Remise du travail :

La date de remise est le 4 octobre à 23h30. Une pénalité de 3 points par jour sera appliquée lors d'un retard.

Documents à remettre :

Les exercices doivent être codés dans un fichier TP.m. Les réponses aux questions doivent être incluses dans le code. Les exercices doivent être séparés par des cellules (*Insert cell divider* ou *%%*). Vous devez bien identifier chaque exercice et sous-question, et bien commenter le code.

Créer un fichier html à l'aide de *Publish to html* de Matlab pour avoir un fichier html de votre code et de vos graphiques. Veuillez remettre tous vos fichiers (.m et dossier html) dans un seul fichier zip et nommez ce fichier selon le format: 'Nom1_1234567_Nom2_8901234_TP2.zip'.

Pour inclure les fonctions dans le html, ajouter 'type fonction.m' dans votre .m principal. Vérifier également que les graphiques et les figures sont lisibles dans le html.

Une pénalité de 3 points sera appliquée si ces consignes ne sont pas respectées.

Première Séance: Qualité des images

Exercice I (3 points) : Analyse de la résolution

Afin de recalibrer des images de fond d'oeil de la rétine avec d'autres modalités d'imagerie médicale, on a besoin de segmenter manuellement le disque optique (point de repère) identifié par un contour rouge (Figure 1).

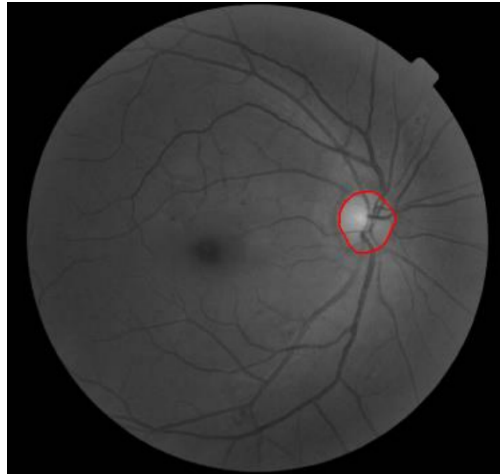


Figure 1 – Disque optique dans une image fundus.

Q1. (0,5 point) Lisez et affichez l'image `fundus_image.png`.

Q2. (0,5 point) Implémentez une fonction `reduire()` qui prend en entrée une image `I_originale` et un facteur de réduction `f` et en sortie retourne une image `I_reduite` qui a une résolution réduite d'un facteur `f`.

(Ne **pas** utiliser les fonctions prédéfinies de traitement d'image de Matlab!)

Q3. (1 point) Affichez l'image réduite pour les valeurs suivantes du facteur `f`: 2, 4, 8, 16, 32, 64, 150 et mettez un titre qui spécifie le facteur utilisé pour chaque image affichée.

Note: Vous pouvez utiliser la fonction `imshow(img, 'InitialMagnification', 'fit')` (`img` est l'image réduite) pour conserver la même taille d'affichage entre les images. Vous pouvez aussi placer le code d'affichage dans une boucle `for` terminant par la commande `pause(n)`.

Q4. (1 point) Qu'observez-vous dans l'image réduite au fur et à mesure que le facteur de réduction `f` augmente? À partir de quelle réduction la segmentation du disque optique sera-t-elle difficile à réaliser à partir de quelle réduction la segmentation des vaisseaux sera-t-elle difficile à réaliser?

Exercice II (3 points) : Analyse du contraste

Q1. (0,5 point) Lisez et affichez l'image originale I_1 : `crane1.png` et ses images dérivées I_2 et I_3 (respectivement `crane2.png` et `crane3.png`). Mettez un titre aux figures. Comparez visuellement ces 3 images.

Q2. (0,75 point) Affichez le profil le long de la même ligne diagonale du coin inférieur gauche au coin supérieur droit sur les 3 images dans un graphique. Comparez visuellement les trois profils en relation avec le contraste des trois images.

Note: Vous pouvez utiliser la fonction `ginput()` pour obtenir les coordonnées (x,y) de deux points sélectionnés directement sur l'image et la fonction `improfile()` pour obtenir le profil le long d'une ligne.

Q3. (0,5 point) Le contraste local C est donné par l'équation suivante:

$$C = \left| \frac{I_{cible} - I_{fond}}{I_{fond}} \right|$$

où I_{cible} représente la moyenne des intensités lumineuses à l'intérieur d'une région R_{cible} contenant des éléments anatomiques et où I_{fond} représente la moyenne des intensités à l'intérieur d'une région de fond R_{fond} avoisinant la région cible.

Implémentez une fonction `contraste()` qui calcule le contraste local C associé à une image en prenant comme région:

$$R_{cible} = \{y \in [120, 150], x \in [114, 144]\}$$

$$R_{fond} = \{y \in [155, 185], x \in [59, 89]\}$$

Note: Dans Matlab, les matrices sont indexées [ligne, colonne], c'est-à-dire [y, x].

Q4. (0,25 point) Affichez sur l'image I_1 le contour des deux régions R_{cible} et R_{fond} .

Q5. (0,5 point) Calculez les contrastes locaux C_1 , C_2 et C_3 associés aux images I_1 , I_2 , I_3 et comparez ces valeurs vis-à-vis des images.

Q6. (0,5 point) Discutez de l'impact du contraste d'une image pour le diagnostic de pathologies.

Exercice III (4 points) : Quantification du bruit

La formule du rapport SNR différentiel est donnée par l'équation suivante:

$$SNR_{diff} = \frac{A |f_{cible} - f_{fond}|}{\sigma_{fond}(A)}$$

où f_{cible} est la moyenne et $\sigma_{fond}(A)$ l'écart type des intensités lumineuses calculées sur un rectangle d'aire A de la région cible de l'image, f_{fond} la moyenne des intensités lumineuses calculées dans un rectangle d'aire A de la région du fond.

Q1. (1 point) Implémentez une fonction `rapport_SNR()` qui calcule le rapport SNR différentiel sur une image. La fonction doit retourner la valeur **en décibels** (dB).

Q2. (0,5 point) Calculez le rapport SNR différentiel SNR_{diff1} de l'image originale `crane1.png` pour les régions cible et fond définies dans l'Exercice II - **Q3.**

Q3. (0,5 point) Additionnez du bruit gaussien de moyenne nulle et de variance égale à 0.005 aux intensités de l'image I_1 en utilisant la fonction `imnoise()`. Affichez l'image bruitée (sans oublié le titre de la figure).

Q4. (0,5 point) Calculez le rapport SNR différentiel SNR_{diff2} de l'image bruitée.

Q5. (1 point) Augmentez la variance du bruit gaussien (variance=0.2; 0.5; 1). Affichez les images bruitées et recalculez le rapport SNR différentiel de chaque image bruitée.

Q6. (0,5 point) Qu'observez-vous lorsque le bruit augmente?

Seconde Séance : Filtrage de signaux

Exercice IV (5 points) : Filtre 1D

Un médecin intensiviste souhaite suivre l'activité électrique du diaphragme (AEDI) lors de l'expiration chez les patients sous une ventilation mécanique pour étudier l'impact de la pression d'oxygène donnée au patient sur cette activité. Cependant, au moment de la lecture, celui-ci se révèle illisible.

L'objectif est d'aider le docteur à lire le signal de l'AEDI en filtrant les rouces de bruits dues aux activités cardiaques et au mouvement des électrodes au cours de l'enregistrement du signal.

Q1. (1 point) À l'aide de la fonction `load()`, importez les données correspondant à l'AEDI bruité: `signal_diaph.mat`. Affichez le graphe de ce signal et rajoutez une grille au graphique; puis ajoutez un titre, une légende, et des annotations pour l'axe horizontal en secondes et l'axe vertical. Notez que le signal a une fréquence d'échantillonnage de 500 Hz.

Q2. (1 point) Calculez la transformée de Fourier Discrète (TFD), en utilisant la fonction `fft()` de Matlab. Tracez la magnitude et la phase du spectre ainsi générées en utilisant la fonctions `abs()` et `angle()` de Matlab.

Q3. (1 point) Réalisez un filtre à réponse impulsionnelle finie (RIF) de type passe haut en utilisant la fonction `fir1()` avec un ordre de 400 et une fréquence de coupure $f_c = 4\text{Hz}$ (utiliser la fréquence normalisée qui correspond à $w_c = 2\frac{f_c}{f_e}$).

Tracez sa réponse impulsionnelle en utilisant la fonction `impz()` de Matlab. Tracez sa réponse en fréquence à l'aide de la fonction `freqz()` de Matlab. Filtrez le signal original avec le filtre passe haut. Affichez le signal filtré (mettez un titre, légende...) dans en fréquentiel et en temporel. Qu'observez-vous dans chacun de ces domaines?

Q4. (1 point) Réalisez un deuxième filtre (RIF) de type passe-bas en utilisant la fonction `fir1()` avec un ordre de 35 et une fréquence de coupure $f_c = 50\text{Hz}$. Tracez sa réponse impulsionnelle et sa réponse en fréquence. Filtrez le signal déjà filtré à la question 3 avec le filtre passe-bas. Affichez le signal filtré (mettez un titre, légende...) en fréquentiel et temporel. Qu'observez-vous dans le domaine temporel et fréquentiel?

Exercice V (4 points) : Filtrage par fenêtres

Suite au filtrage réalisé à l'exercice précédent, on souhaite étudier la possibilité d'améliorer le résultat en exploitant le filtrage par fenêtres.

Q1. (1 point) En vous référant au cours (Filtrage de signaux 1D), pour quelles raisons le filtrage par fenêtre pourrait-il être une bonne stratégie pour cette application?

Q2. (0,5 point) Les filtres que vous allez comparer auront une fréquence de coupure normalisées de 0.12 et 0.28. En considérant un signal similaire à celui de l'exercice précédent (même fréquence d'échantillonnage), calculez la valeur de ces fréquences de coupures en Hz.

Pour les questions suivantes, vous utiliserez l'interface graphique Matlab `signal1D` fournie sur le Moodle du cours.

Q3. (0,5 point) Dans l'onglet Conception de filtres FIR: réalisez des filtres à réponse impulsionnelle finie de type passe bas pour un ordre $N = 20$ pour les différentes fenêtres (Rectangle, Blackman, Bartlett, Hamming, Hanning, Tukey) et pour les valeurs de fréquences de coupures normalisées suivantes: 0.12, 0.28. Sauvegarder les coefficients du filtre à chaque fois dans un fichier `.mat` puis filtrer le signal obtenu à la question **Q3.** de l'Exercice IV avec chacun de ces filtres. Affichez les signaux filtrés dans des figures séparées en dehors de l'interface (il faut d'abord enregistrer le signal de sortie à l'aide du bouton avec l'icône disquette).

Q4. (2 points) Commentez les résultats des filtres dans les deux domaines (fréquentiel et temporel) en fonction de la fréquence de coupure et du type de fenêtre utilisé.