

Projet final : Depth Cueing sous LabVIEW

I – Objectif

Ce projet a pour but le rendu de profondeur (depth cueing) d'une acquisition volumique issue d'un scanner IRM 3T, sous LabVIEW. Il reprend de nombreux éléments des TP déjà réalisés (image volumique, traitement d'image) et doit permettre également d'introduire de nouvelles fonctionnalités en visualisation 3D (seuillage, rendu de profondeur, tables de couleur...).

Nous allons travailler sur une image volumique (un volume numérique (VN)) issue d'une acquisition IRM du cou et de la tête. Il s'agit d'un volume de 256 coupes composées chacune de 256*256 voxels.

Le rendu de profondeur s'effectuera dans trois conteneurs « Image » de LabVIEW, selon les trois axes classiques : axial, coronal, transversal et plus si affinités...

Le codage de l'intensité (luminance) de ces images monochromes est réalisé sur des entiers codés sur 16 bits (INT16) qui pourront être convertis sur 8 bits (U8).

On rappelle à toutes fins utiles que le depth cueing (DC) est une technique de rendu de volume nécessitant au préalable une segmentation du volume afin de dégager le volume d'intérêt que l'on souhaite visualiser.

La mise en œuvre qui suit n'est qu'une proposition de développement. Le but de ce projet est de s'approprier la technique de ce rendu puis d'ajouter un certain nombre de fonctionnalités qui seront proposées et/ou dont vous pourrez prendre l'initiative...

II – Mise en œuvre basique

On pourra créer un nouveau répertoire (syntaxe du type DC_Binôme1_Binôme2) et y créer un nouveau VI nommé « DC_1 »

On créera sur la face avant un conteneur « Image » (destiné à la visualisation des coupes scanner selon le plan natif (sagittal). On créera également un deuxième conteneur « Image » destiné à la visualisation en DC selon ce plan.

La figure 1 illustre l'apparence que pourrait prendre votre face avant.

Il s'agit là du programme de base, servant de point de départ à l'ensemble du développement.

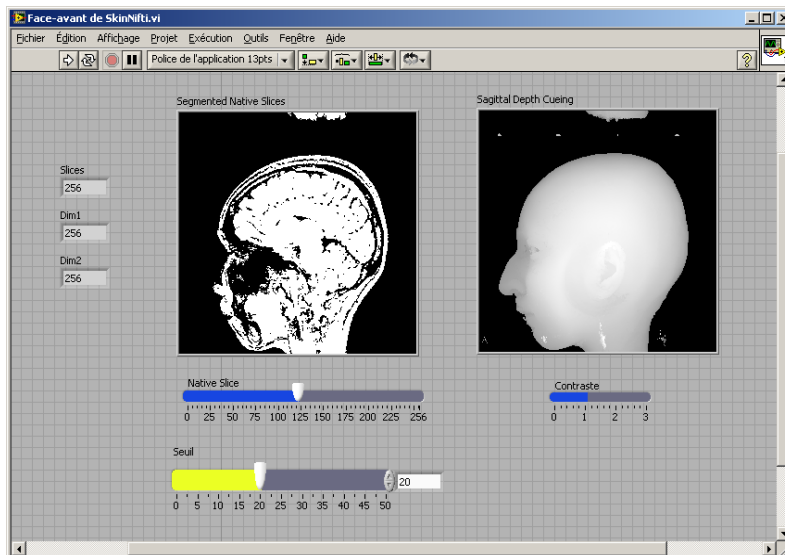


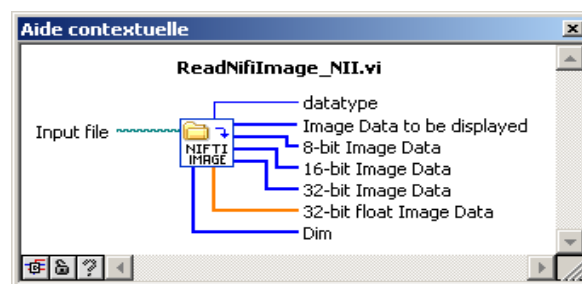
Figure 1 : Une proposition de face avant « basique »

a – Lecture et seuillage du volume

On ré-utilisera le sous-VI « ReadNiftImage_NII » qui vous a été fourni, dans la librairie LabVIEW : « NeuroImage.lib ». Il ouvre le fichier « .nii » donné en entrée et renvoie plusieurs données dont :

- Un tableau d'entiers : « Dim » dont les indices 1, 2 et 3 renvoient vers les trois dimensions du volume numérique (nombre de coupes et dimensions en X et Y de chaque de ces coupes)
- Un indicateur « datatype » du type de données codant les valeurs des voxels de l'image
- 4 sorties possibles selon la valeur de « datatype »
- L'image à afficher à l'écran (en niveaux de gris sur 8 bits)

L'aide en ligne de ce sous-VI figure ci-dessous.



On insistera sur le fait qu'il sera rapide d'utiliser la sortie « Image Data to be displayed » qui est la conversion en 8 bits de l'image originale sortie du scanner.

On pourra également écrire un sous-VI « Seuille_Volume » dont les entrées seront le volume original converti en 8 bits et un niveau de seuil (entier) et dont la sortie sera le volume seuillé (en 8 bits), binaire (0 ou 255) correspondant aux voxels dont la valeur est supérieure au seuil choisi.

On rappelle que le seuillage est une opération globale. Programmez-la la plus simplement possible.

b – Rendu en depth cueing selon le plan natif (sagittal)

On rappelle que le rendu en depth cueing consiste en une image 2D, construite de la façon suivante :

- L'image 2D à les dimensions d'une coupe (sagittale)
- On affiche les voxels du volume d'intérêt en partant des « coupes » les plus éloignées vers les plus proches, en augmentant régulièrement la luminances des pixels.

L'algorithme en lui même est très simple à implémenter. Calculer l'image avant de l'afficher. Pensez à correctement utiliser les outils de construction d'une image.

On pourra jouer (légèrement) sur le contraste de l'image de rendu en multipliant la valeurs de tous les pixels à afficher par un coefficient proche de 1.

III – Améliorations de base

Le projet, à cette étape, est correct mais largement perfectible. Nous allons donc y apporter des améliorations

a – Vue du rendu en depth cueing selon les deux autres plans : axial et coronal.

On pourra récupérer les fonctions implémentées lors du TP « Image volumique » pour les adapter à la génération des images DC selon les deux autres plans classique de la radiologie.

b – Modulation du seuillage

Le seuillage présenté lors de la première partie est très basique. A vous d'en imaginer des améliorations (simples)

c – Gestion de la couleur du depth cueing

Rien de bien compliquer. Il s'agit juste d'apporter un peu de couleur lors du calcul (ou de l'affichage) du depth cueing. Cela peut également améliorer le contraste.

III – Améliorations avancées

Ici encore, quelques propositions

- Les images IRM sont assez bruitées. Pouvez-vous imaginer (et implémenter) un petit filtre anti-bruit ???

- Les plans sagittal, coronal, axial sont assez simples. Pouvez-vous implémentez des plans obliques ?

IV – A vous de jouer

Une grande part est laissée à votre sens de l'initiative, et ceci tout au long de ce projet.

A vous donc d'être imaginatif et de proposer d'autres améliorations. Tant au niveau du fond que de l'ergonomie.

Même non finalisées, n'hésitez pas à proposer plusieurs versions de votre travail, selon les pistes que vous avez explorées.

Enfin, et surtout : commentez, expliquez et clarifiez votre code !!! Au delà de faciliter la compréhension (et donc la correction), cela vous permet également de structurer au mieux votre propre analyse.

Bon courage et bon TP....