

ENDOSCOPIE VIRTUELLE

IMI - Projet Majeure - CPE Lyon

Camille Farineau (camille.farineau@cpe.fr) - Nicolas Ranc (nicolas.ranc@cpe.fr)



Contexte

Le but de ce projet est de développer un outil de rendu permettant de naviguer à l'intérieur d'un patient. Cette navigation virtuelle permet de diagnostiquer des symptômes évocateurs de tumeurs par exemple et ainsi d'éviter, dans certains cas, une endoscopie réelle.

L'endoscopie virtuelle est réalisée à partir d'image IRM du patient. Ces données IRM peuvent être visualisées par slice suivant les 3 axes d'orientation.

État de l'art

De nombreux outils existent afin de reconstruire un modèle 3D à partir de données 2D d'IRM et de naviguer dans celui-ci. Un logiciel très connu est 3D Slicer. Celui-ci possède un module d'endoscopie où l'utilisateur crée une trajectoire prédéfinie dans la cavité et il peut ensuite visualiser le parcours dans le modèle 3D.

Il n'existe pas d'outils où le praticien peut se déplacer, avec une manette par exemple, directement dans la cavité. De plus la reconstruction volumique ou surfacique doit être fait par l'utilisateur au préalable.

3D Slicer est un outil utilisant la bibliothèque VTK (Visualization Toolkit) développée par Kitware.

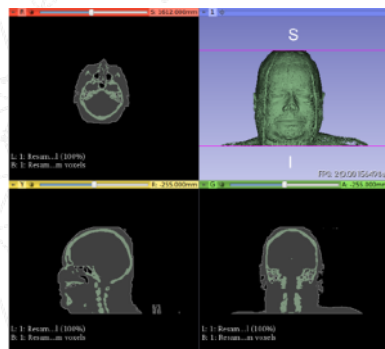


Fig.1 : Visualisation avec 3D Slicer

VTK

VTK est une bibliothèque libre de classes C++ spécialement dédiée à notre problématique. L'utilisation de cette librairie nous a permis de simplifier la visualisation des données IRM ou encore la reconstruction de surface à partir des données.

Implémentation

Nous avons pris le parti de réaliser un viewer simple en C++ utilisant Qt pour l'interface graphique et VTK pour la gestion des données 3D et IRM et des algorithmes de traitement d'image et visualisation 3D.

Ce viewer comporte 3 fenêtres permettant de visualiser les slices suivant les 3 orientations et une fenêtre correspondant à une scène 3D afin de visualiser la reconstruction surfacique.

Données IRM

Les données IRM utilisées correspondent à un champ scalaire 3D. Ce champ est organisé en slices suivant 3 orientations (XY, XZ, YZ).

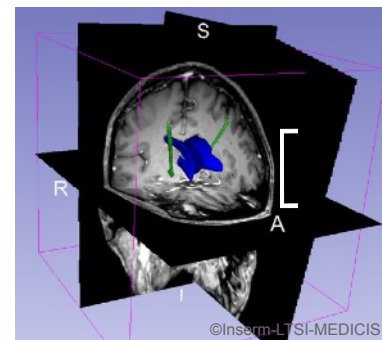


Fig.2 : 3 plans de coupe pour les slices (Slicer 3D)

Visualisation des slices

VTK possède une classe dédiée (vtkResliceImageViewer) à la visualisation des différentes slices. Cette classe utilisera un callback permettant de gérer les interactions avec la souris, ce qui permet de changer de slice ou de savoir à quel point du champ scalaire 3D correspondant la position de la souris.

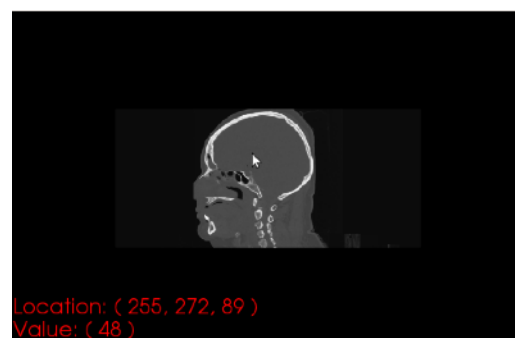


Fig.3 : Visualisation d'une slice suivant orientation XY

Reconstruction surfacique

Afin de visualiser les données en 3D, on procède à une reconstruction d'une isosurface grâce à l'algorithme de Marching Cubes. En effet les données IRM ont une valeur dépendant de la densité de l'élément associé ce qui permet de visualiser les organes, les tissus, les os etc. L'échelle de densité associée est l'échelle de Hounsfield.

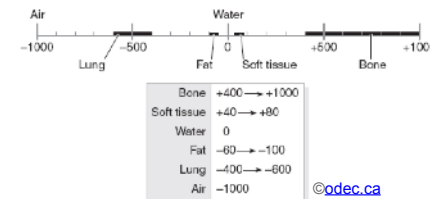


Fig.4 : Échelle de Hounsfield

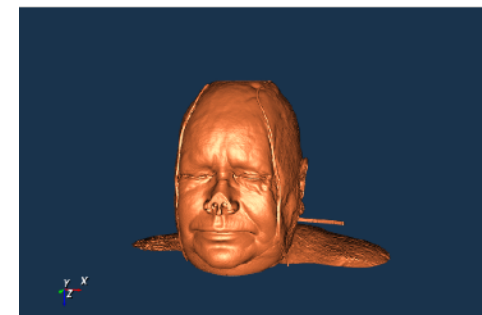


Fig.5 : Isosurface 3D après Marching Cubes

Navigation dans l'espace 3D

La navigation dans l'espace 3D se fait directement à partir d'une caméra associée à la fenêtre de rendu VTK. Cette caméra se déplace grâce au clavier ou à l'aide d'un joystick.

Afin d'explorer la cavité (intestins, trachée, cavité nasale etc.) d'intérêt pour le praticien, il suffit de sélectionner une zone sur les slices et la caméra se placera à l'endroit choisi.

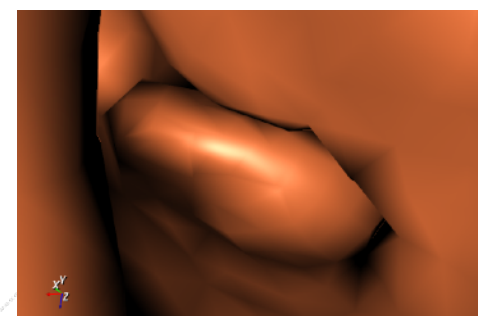


Fig.6 : Visualisation dans une cavité

Collision entre la caméra et la surface

L'application intègre une gestion de collision entre la caméra et la surface. Ainsi lorsque l'on déplace la caméra dans une cavité on ne pourra pas en sortir par erreur.

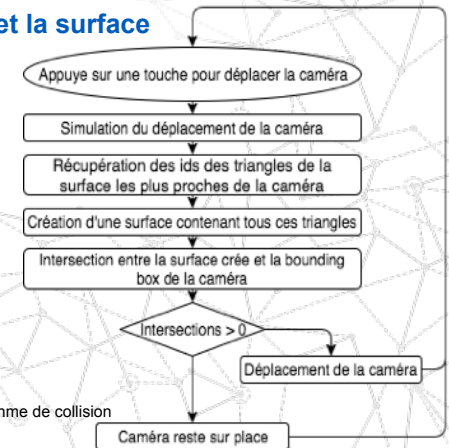


Fig.5 : Algorithme de collision

Limitations

- La reconstruction surfacique comporte des artefacts. Certains triangles apparaissent de manière non vraisemblable (zones non lisses ou non détaillées).
- L'implémentation sur GPU n'a pas été réalisée. L'utilisation classique de VTK a été favorisée afin d'obtenir rapidement des résultats.
- La reconstruction surfacique se fait sur tout le volume. Une segmentation de la zone d'intérêt et la reconstruction de cette seule zone serait plus judicieux (segmentation croissance de régions 3D)
- Ajout de curseur sur les slices afin de savoir où la caméra est située aurait été un plus.

Conclusion

Ce viewer final qui a été développé est facile à utiliser, simple et permet la visualisation de cavités corporelles en 3D.

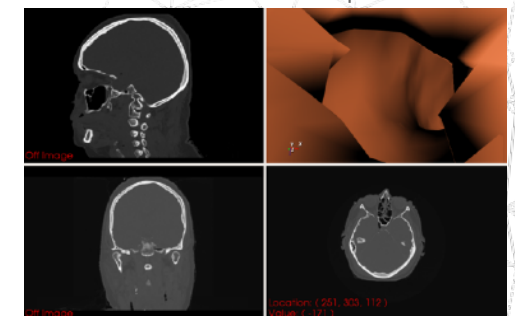


Fig.6 : Aperçu du viewer final

Bibliographie

1. VTK : <https://www.vtk.org/>
2. Example de VTK : <https://lorensen.github.io/VTKExamples/site/Cxx/>
3. Dataset : https://mri.radiology.uiowa.edu/visible_human_datasets.html
4. Module Endoscopy de Slicer : <https://www.slicer.org/wiki/Documentation/4.8/Modules/Endoscopy>