

Rapport de Synthèse

Développement d'outils de visualisation d'images médicales 3D

Claude Stéphane Vienne

Centre de lutte contre le cancer Léon Bérard
28 rue Laennec
69 373 Lyon Cedex 08

Laboratoire LIRIS (FRE CNRS 2672),
Université Lumière Lyon 2
5 Av. Pierre Mendès-France - 69676 Bron

Enseignant(s) responsable(s) :

Stéphane Bres – INSA
David Sarrut – Lyon 2

Résumé

Une partie des chercheurs du département de radiothérapie du Centre Léon Bérard travaillent sur des techniques d'imagerie afin d'améliorer l'efficacité des traitements. Pour cela, ils ont besoins d'outils de visualisation d'images tridimensionnelles. Ce projet de fin d'études s'est organisé autour de deux activités principales :

- **Application de visualisation d'images médicales** : Conception et développement d'une plate-forme logicielle dédiée à la visualisation et la manipulation d'images volumiques médicales.
- **Algorithme de visualisation de champs de vecteurs tridimensionnels** : Pour se rendre compte des mouvements et des déformations des organes des patients, des champs de déformation sont calculés à partir d'images volumiques. Les lignes de champs de vecteur, une méthode de visualisation de champs de vecteurs a donc été étudiée et évaluée.

Mots clefs

Imagerie médicale – Images volumiques – Champs de vecteurs – Lignes de champs

Abstract

Some researchers of the Léon Bérard Center radiotherapy department work on imagery techniques to improve treatments efficiency. Therefore, they need tools to visualise three-dimensional images. This project is composed of two major activities :

- **Medical images visualisation application** : Design and development of a software platform to visualise and manipulate three-dimensional medical images.
- **Three-dimensional vector fields visualisation algorithm** : Vectors Fields are computed from 3D images to take into account patient's organs deformations and movements during treatment. Line Integral Convolution, a method to visualise such vectors fields has been studied and evaluated.

Key words

Medical Imaging – Three-dimensional images – Vector Fields – Streamlines.

I. Introduction

Le PFE "Développement d'outils de visualisation d'images médicales 3D" s'est déroulé au Centre de lutte contre le cancer Léon Bérard (CLB), au sein du service de Radiothérapie. Dans le cadre de projets de recherche, le CLB travaille en collaboration avec des chercheurs du LIRIS¹ chargés d'effectuer de la recherche fondamentale pour assister informatiquement la radiothérapie. Plus spécifiquement, les travaux de recherche se situent au niveau du traitement des images.

La radiothérapie est l'une des trois principales techniques de traitement contre le cancer, avec la chirurgie et la chimiothérapie. Elle consiste à délivrer une dose de rayon à haute énergie aux tumeurs, tout en essayant d'épargner les cellules saines voisines. La planification du traitement et sa réalisation repose sur l'imagerie et des logiciels d'analyse et de traitement de l'image.

Les axes des travaux de recherche sont d'une part le contrôle du positionnement du patient sur la table d'irradiation, et d'autre part l'intégration du mouvement et de la déformation des organes avant et pendant le traitement.

Pour leurs travaux, ils ont besoin d'outils spécifiques de visualisation et de manipulation d'images 3D médicales. Ces outils sont très liés aux besoins des chercheurs et nécessitent l'utilisation d'une application dédiée, à la fois évolutive et maintenable. Le premier objectif de ce PFE était de concevoir et de réaliser cette application.

La deuxième partie du PFE s'est inscrite plus directement dans le monde de la recherche. Il s'agissait d'étudier et de mettre en œuvre un algorithme de visualisation de champs de déformation tridimensionnels.

Ce document présente d'abord le format des images volumiques qui constituent la base de ce PFE. Il s'attache ensuite plus particulièrement à synthétiser le travail effectué, dans un premier temps sur l'application de visualisation puis sur les champs de déformation. Enfin, la conclusion donne mon avis sur ce PFE ainsi que les perspectives issues de ce projet...

¹ Laboratoire d'InfoRmatique en Images et Systèmes d'information, CNRS FRE 2672

II. Imagerie médicale

II.1. Acquisition

Dans le cadre de la radiothérapie, pour localiser les tumeurs par rapport aux organes, il est nécessaire d'obtenir une image volumique du patient. Les médecins et les physiciens utilisent une méthode d'acquisition volumique, la tomographie. Un scanner émet des rayons X sur la zone à radiographier. L'atténuation de ces rayons permet de quantifier la densité de la matière traversée dans le corps humain, et donc son identification.

II.2. Contenu des images

Les données issues de cette méthode d'acquisition sont ensuite stockées dans un fichier d'image volumique. Les différents voxels² de l'image correspondent à un échantillonnage régulier et uniforme du volume réel. La résolution des voxels est donnée par la résolution des scanners.

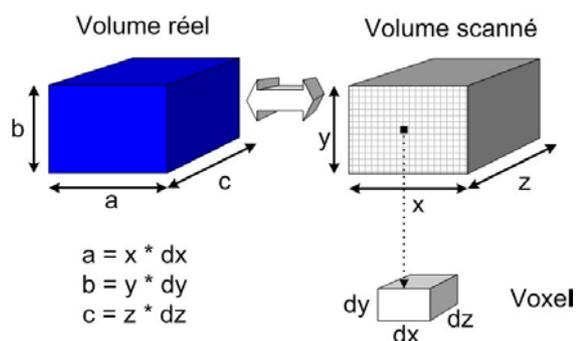


Figure 1 : représentation des images volumiques
 a , b et c sont les dimensions réelles du volume en mm.
 x , y et z représentent les dimensions du volume en voxel.
 dx , dy et dz sont les dimensions réelles des voxels en mm.

Les informations portées par les voxels indiquent des niveaux de densité de matière exprimée en unités *Hounsfield* [Cli00], liés à la densité réelle. Ces intensités peuvent ensuite être converties en niveau de gris pour l'affichage des images.

Une image volumique classique peut par exemple représenter un volume réel de 480x480x335mm, pour une résolution de 512x512x67 voxels. La taille des voxels peut en être déduite (figure 1) : 0,98x9,98x5. La dernière dimension étant assez espacée, on parle souvent de coupes successives, comme l'illustre la figure 2.

² Pixel en 3 dimensions

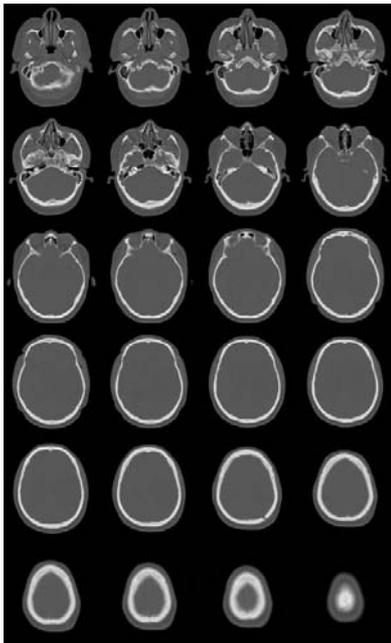


Figure 2 : Quelques coupes d'un volume crânien

III. Développement de l'application

III.1. Contraintes particulières

Outre les fonctionnalités souhaitées pour l'application, certaines exigences non fonctionnelles sont rapidement apparues au début du PFE et ont dû être prises en compte.

- Les chercheurs de l'équipe image du CLB travaillent sous un environnement Linux. Ainsi, l'application à développer doit donc fonctionner sous Linux pour réutiliser les nombreux outils de manipulations d'images issus des travaux de recherche.
- Les plus importants critères non fonctionnels de l'application étaient son évolutivité et sa maintenabilité. En effet, les travaux des chercheurs se poursuivant durant le PFE et au-delà, de nouvelles fonctionnalités ou besoins apparaissaient régulièrement. Prendre en compte ces modifications étaient donc nécessaire, tout en essayant de limiter au maximum leur influence sur le reste de l'application.

Pour prendre en compte ces exigences, les mesures suivantes ont été prises :

- *Développement orienté objet* : une définition précise des objets et des classes mises en jeu permet à la fois de mieux localiser les différentes fonctionnalités et d'en intégrer de nouvelles.
- *UML³ [Sou01]* : Au cours de la spécification, UML a permis de bien définir les interactions qui existent entre les différents

³Unified Modeling Language

objets ainsi que les mécanismes de fonctionnement.

- *Documentation fournie* : Utilisation d'un outils de génération de documentation automatique, Doxygen [Dox].
- *Règles de codage* à respecter dans un soucis de faciliter la compréhension du code associé à l'application.

III.2. Librairie graphique

Le développement d'une application pour un environnement Linux a nécessité d'étudier et de choisir une bibliothèque de composants graphiques. Les solutions étudiées furent GTK+ [Gtk], Java [Java] et Qt. [Qt].

Qt, la librairie C++ proposée par TrollTech a finalement été retenue. D'abord, le langage C++ a été préféré à Java, d'une part pour son efficacité et d'autre part parce que beaucoup d'outils des chercheurs étaient déjà en C++. Par rapport à GTK, la documentation et les outils fournis par Qt sont beaucoup plus riches. Enfin, Qt reste très portable car il est possible de recompiler l'application dans différents environnements en prenant soin de ne pas utiliser de bibliothèques spécifiques à un système.

Le principal avantage de Qt est son mécanisme de "signals and slots". Il permet de connecter dynamiquement un signal d'un objet à un ou plusieurs slots d'un autre. Lorsque certains événements surviennent sur un objet, une fonction spécifique d'un autre objet peut être déclenchée grâce à l'émission d'un signal. La figure 3 présente le fonctionnement de ce mécanisme. Il repose sur l'utilisation d'un Meta Object Compiler qui génère automatiquement les liens entre les objets.

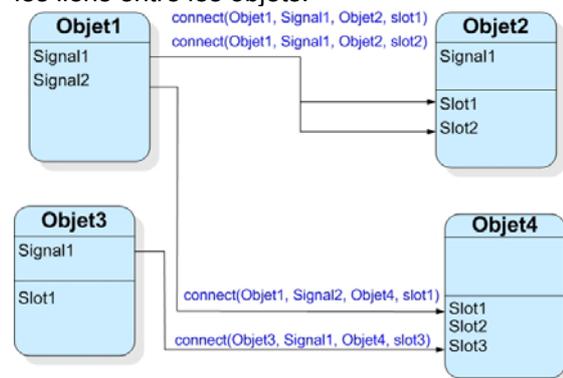


Figure 3: Signals and Slots

Qt Designer, une application de développement rapide, a été utilisée. Cependant, la partie la plus importante du travail effectué concerne essentiellement les fonctions de traitements d'images.

III.3. L'application

L'application, baptisée *ViewVox*, doit permettre d'afficher des coupes en deux dimensions issues d'un volume de données. Pour récupérer une coupe à partir d'un volume, il faut sélectionner les voxels appropriés afin d'extraire une image 2D. Pour répondre aux besoins des médecins, les coupes peuvent être extraites selon l'un des 3 axes du volume. L'efficacité de l'algorithme associé à l'extraction de coupes a été particulièrement travaillée car il est l'un des centres de l'application.

L'application peut ouvrir simultanément plusieurs volumes d'images. Elle propose 3 emplacements configurables pour l'affichage des coupes désirées et 1 pour afficher des informations sur l'image courante. La figure 4 présente l'application lorsqu'une seule image volumique est ouverte, mais affichée selon les 3 axes possibles - coupes axiales, sagittales et coronale.

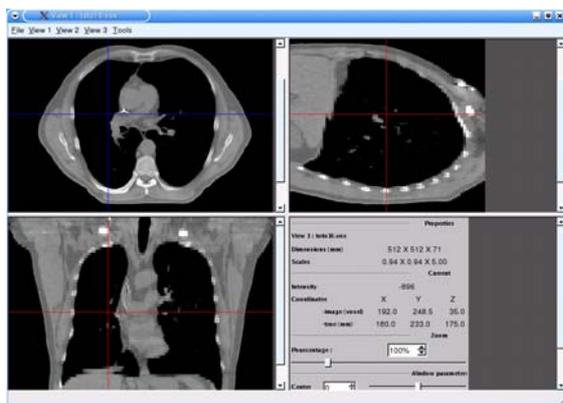


Figure 4 : Application

Les outils de manipulation proposés aux utilisateurs se répartissent en trois catégories : les manipulations géométriques des images, les manipulations sur les intensités et celles sur les liens entre les images.

Les manipulations géométriques se basent sur l'utilisation de matrices de transformation. Une image 2D est d'abord extraite du volume - image d'entrée. On désire lui faire subir une transformation géométrique afin d'afficher l'image résultante - image de sortie. Les matrices de transformation fournissent les coordonnées d'un pixel de l'image de sortie à partir des coordonnées du pixel de l'image d'entrée. Ces matrices permettent notamment de faire subir aux coupes des zooms, des rotations et des mises à l'échelle pour la taille des voxels.

Les manipulations sur les intensités sont assez délicates. Afin de pouvoir afficher des images il

faut convertir les informations de densité des fichiers sur 256 niveaux de gris. Pour améliorer la fluidité de l'application, ces conversions ont été optimisées à la conception et durant le développement de l'algorithme. Les images volumiques peuvent atteindre des tailles de 20 millions de voxels (35Mo), et cette conversion doit être rapide.

A l'ouverture d'un fichier, une deuxième image volumique en niveaux de gris est créée. La quantité de données en mémoire est plus importante, mais le nombre de conversion à effectuer s'en trouve largement diminué. La conversion des densités s'effectue de manière linéaire vers des niveaux de gris et ce sur une fenêtre paramétrable - figure 5. Le volume en niveau de gris peut-être recalculé de manière interactive en changeant les paramètres le fenêtrage et grâce à un système de prévisualisation : Seules les coupes courantes sont actualisées - 3 images maximum. La conversion totale ne se fait que lorsque l'utilisateur est satisfait de ces réglages.

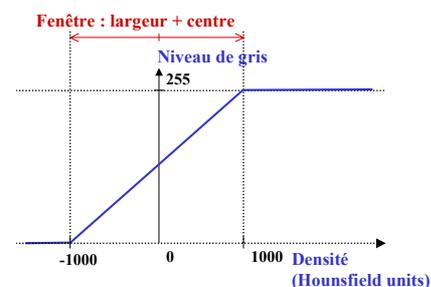


Figure 5 : Fenêtrage, Conversion intensité

Les liens entre les images ont été les plus délicats à mettre en œuvre. L'application peut afficher différentes images volumiques et différentes coupes de ces images. Les coupes affichées peuvent être liées entre elles : un déplacement sur la coupe courante peut actualiser les coupes des autres emplacements de manière à toujours voir sur les 3 emplacements le point courant - ou son correspondant.

Lorsque l'application affiche 3 coupes d'une même image, ce mécanisme donne l'impression de se déplacer dans un volume et permet de mieux s'y situer. Pour des coupes d'images différentes, il facilite la comparaison des volumes car on visualise le correspondant du point courant dans les autres volumes.

La correspondance entre des images différentes est d'autant plus compliquée que les volumes peuvent ne pas avoir les mêmes résolutions, ne pas posséder les mêmes échelles ou encore avoir des transformations géométriques différentes...

Il est à noter que dans l'application, toutes les informations médicales affichées sont directement issues du volume original. La conservation de l'intégralité de l'information étant primordiale dans le domaine médical [Bres].

L'application a rapidement été jugée satisfaisante par les chercheurs, qui l'utilisent fréquemment. Elle leur offre de nombreux avantages : comparaisons d'images, mise en évidence de certaines densités, repérage de points, outils géométriques... Bien que quelques ajouts furent nécessaires, la fin du stage a pu être consacrée à un travail plus orienté vers la recherche.

IV. Visualisation des Champs de déformation

IV.1. Problématique

L'un des axes de recherche du CLB est l'étude du mouvement et de la déformation des organes. L'exemple des poumons est très représentatif. Il est possible d'acquérir des images scanners à des instants différents du cycle respiratoire. Le but est d'évaluer un champ de déformation entre 2 acquisitions à travers des méthodes de mise en correspondance non rigide d'image 3D [Bo102]. Les vecteurs de déplacement obtenus pourront être utilisés par la suite comme information pertinente pour la modélisation de la déformation des poumons pendant la respiration ainsi que pour des études de dosimétrie.

Les champs de vecteurs ainsi calculés sont très volumineux et difficiles à représenter. Plusieurs méthodes peuvent être envisagées : la projection 2D ou superposition 3D des vecteurs sur une coupe – figure 6. Cependant ces deux méthodes présentent quelques inconvénients : affichage de quelques vecteurs uniquement, difficultés de manipulation, vue coupe par coupe des images...

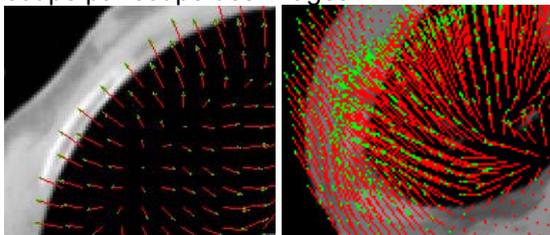


Figure 6 : projection et superposition de vecteurs

La deuxième partie de ce PFE a donc été d'étudier et d'implémenter une méthode pour visualiser ces champs de vecteurs et la comparer aux autres... Cette méthode est la représentation des lignes de champs.

IV.2. Line Integral Convolution

La représentation des champs de vecteurs par des lignes de champs a été introduite en 1993 par B. Cabral et L. Leedom [Lic93] sous le nom de *Line Integral Convolution* (LIC). Elle a par la suite été améliorée afin d'augmenter son efficacité.

Cette méthode a pour but de générer une image volumique représentant les lignes tangentes au champ de vecteurs en déformant une image de texture aléatoire. La figure 7 illustre le fonctionnement de la méthode : la couleur de chaque voxel de l'image de sortie est donnée par la moyenne des couleurs de la texture aléatoire recouverte par la ligne de champ locale.

Des voxels de l'image de sortie partageant une partie d'une même ligne de champ auront des couleurs très proches et constitueront une ligne. Dans le cas contraire, la texture aléatoire fournie en entrée devrait permettre de distinguer les différentes lignes de champ.

L'aspect délicat de cette méthode provient de sa complexité mathématique et du temps de calcul nécessaire. En effet, le calcul des lignes de champs locales est relativement long car il faut interpoler de nombreux vecteurs.

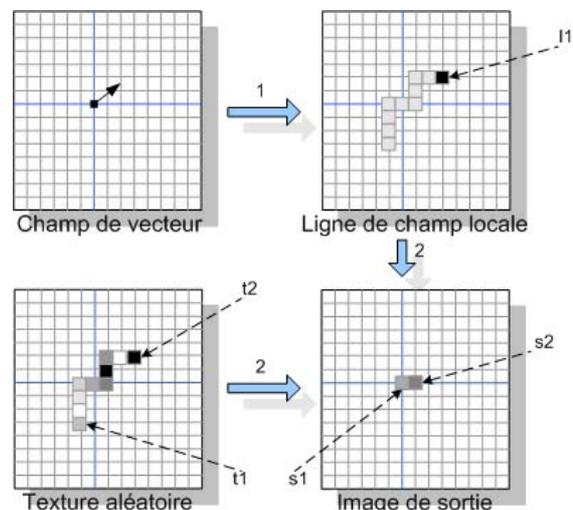


Figure 7 : Illustration fonctionnement LIC

Afin d'améliorer la méthode, de nombreuses propriétés ont été utilisées [Lic95] :

- Une fois une ligne de champ locale calculée, on peut approximer la couleur de quelques points de l'image de sortie qui utilisent cette même ligne de champs en augmentant légèrement sa longueur. Si l_1 prolonge la ligne de champ, on peut déduire la couleur de s_2 à partir de celle de s_1 en tenant compte de t_2 au lieu de t_1 – figure 7.
- Cette approximation peut être rendue plus précise en effectuant les calculs de couleurs

plusieurs fois à partir de lignes de champs différentes puis en moyennant.

c. Combiné au a., répartir les calculs des lignes de champs dans l'image permet de recouvrir plus rapidement l'ensemble des points de l'image de sortie.

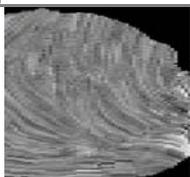
d. Application de masques binaires pour ne calculer les lignes de champs que dans les régions intéressantes (les poumons...).

IV.3. Résultats

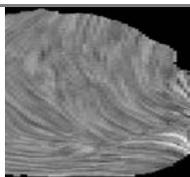
Même si les optimisations apportées à l'algorithme ont complexifié l'algorithme, elles ont cependant grandement contribué à diminuer les temps de génération. Par exemple, pour une même image, il a été possible de passer d'un temps de génération de 12mn30s à 30s.

Afin de rendre compte des résultats obtenus par cette méthode de représentation des champs de vecteurs, il est plus parlant de présenter quelques tests avec leurs paramètres. Ils ont été réalisés dans les mêmes conditions : une texture aléatoire commune, le même champ de vecteur (déformation d'un poumon pendant la respiration) et la même taille pour l'image de sortie. Seuls certains paramètres de l'algorithme ont été changés : la longueur des lignes de champ exprimée en voxel, le nombre d'approximation réalisée sur une même ligne de champ en voxel, l'espacement entre les lignes de champ calculées, et le nombre maximal de calcul pour la couleur d'un voxel de l'image de sortie.

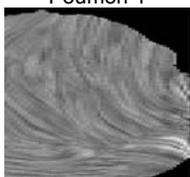
Poumon	1	2	3	4
Longueur des lignes	50	50	50	80
Nombre d'approximation	20	30	20	20
Espacement des lignes	10	15	10	10
Compteur Maximum	5	5	10	10
Temps de calcul	33s	1mn30s	2mn22s	3mn11s



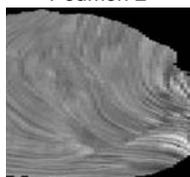
Poumon 1



Poumon 2



Poumon 3



Poumon 4

Les résultats obtenus paraissent instinctivement cohérents car on observe bien un soulèvement de la cage thoracique et un mouvement d'extension au niveau du dos.

L'image 4 paraît de meilleure qualité que les autres, mais au vue des temps de calcul, il est préférable d'adapter les paramètres en fonction des résultats désirés.

L'avantage déterminant de cette méthode est qu'elle se sert de tous les vecteurs du champ de déformation, alors que pour des raisons de lisibilité, les autres méthodes sont obligées de ne pas tous les représenter. On obtient également une image volumique visualisable dans ViewVox. Par contre, les images obtenues sont parfois difficiles à interpréter...

V. Conclusions

Même si le travail réalisé au cours de ce Projet de fin d'études répond aux besoins, de nombreuses améliorations peuvent encore être apportées au travail qui a été effectué au cours de ce projet de fin d'études. Pour la partie applicative, d'autres outils peuvent être ajoutés : la possibilité de configurer le nombre d'emplacements, augmenter les formats d'images supportés, intégrer les outils des chercheurs... Ces améliorations n'ont pas eu le temps d'être intégrées à l'application, mais restent possibles. Pour la partie sur la représentation des champs de vecteurs, l'algorithme lui-même peut encore être optimisé. Une étude détaillée sur l'influence de chaque paramètre pourra être menée.

Ce projet de fin d'étude a vraiment suscité mon intérêt car il m'a permis de découvrir plusieurs aspects de l'informatique : le domaine de la recherche, le monde de l'imagerie ainsi que celui de la l'informatique scientifique et médicale. De plus il m'a permis de travailler sur un projet concret dont les applications étaient visibles.

Références bibliographiques

- [Gtk] GTK+ <http://www.gtk.org>
- [Java] Java – The source for Java technology <http://java.sun.com/>
- [Lic93] B. Cabral & L. Leedom, *Computer Graphics Annual Conference Series*, 1993. "Imaging Vector Fields Using Line Integral Convolution"
- [Lic95] D. Stalling & HC. Hege, *Computer Graphics Annual Conference Series*, 1995. "Fast and resolution independent line integral convolution"
- [Qt] TrollTech – The multi-platform C++ GUI / API <http://www.trolltech.com/>
- [Dox] Doxygen <http://www.doxygen.org/>
- [Bres] S. Bres, 2002, séminaire 5IF "Traitement et analyse de l'image"
- [Cli00] S. Clippe, 2000, Rapport de DEA "Aide au positionnement du patient en radiothérapie conformationnelle par l'utilisation de techniques de recalage d'images 3D et 2D"
- [Bol02] V. Boldea, 2002, Rapport de DEA "Étude de la déformation du poumon lors de la respiration par recalage volumique élastique"