

医用画像可視化技術 ～VTKによる医用画像アプリケーション開発

Medical Image Visualization Technique-Software Development by VTK (Visualization Tool Kit)

田中 真幸* 馬場 明子* 山本 啓二*

Masayuki Tanaka, Akiko Baba, Keiji Yamamoto

数ある放射線治療の中でも、「粒子線がん治療」はその治療効果の高さなどから関心が高い。当社では、粒子線治療装置の国内トップメーカーである三菱電機株式会社（以下、三菱電機）の下で、粒子線治療装置関連の医用画像ソフトウェア開発の実務を担当してきた。先進的な可視化ソフトウェアの基盤として広く使われているVTK (Visualization Tool Kit) ※1を使用することにより、各種表示機能の実現、及び描画性能の確保が容易になっている。本報告では、VTKを使用した医用画像アプリケーションソフトウェアに搭載された医用画像の可視化技術の技術的概要、及び開発事例について紹介する。

(※1) VTKは米国Kitware, Inc.の米国での登録商標である。

Recently, "The Particle Therapy" has attracted much attention because of its high effectivity. We have developed several medical imaging software for the particle therapy under Mitsubishi Electric Corporation, the domestic leading company of particle therapy plants. In order to implement various functions and ensure high performance and high quality image rendering, we have adopted the Visualization Tool Kit (VTK), freely available software for visualization. Here we introduce a technical outline of the medical image visualization by VTK and examples of software development.

1. まえがき

がん治療の有力な手法である放射線治療には様々な治療手法があるが、その中でも、「粒子線がん治療（陽子線、炭素線）」はがんに対する線量集中性の高さが注目され、日本各地及び世界各地で治療専用施設の建設が進みつつある。粒子線治療装置のメーカーとしては国内4社あるが、うち三菱電機は日本国内でのトップメーカー（稼働中 国内8施設）である。当社は、三菱電機の下で、粒子線治療装置関連の各種ソフトウェア開発に従事することで、粒子線治療の発展に貢献している。粒子線治療装置の概略を、図1に示す。

粒子線治療装置関連のソフトウェアのうち、当社が開発を担当している治療計画ソフトウェアを例として、医用画像を取り扱うソフトウェアの概略を以下に述べる。治療計画ソフトウェアは、標的に必要な照射線量を集中させ、かつ、標的以外の正常組織への照射線量ができるだけ小さくなるように、標的に対する粒子ビームの照射方向や、各種照射機器の設定条件を調整、決定する作業を行うソフトウェアである。設定条件全体の内容を「治療計画」と呼び、それをシミュレーションして作成する

ようにする。

ここで、治療計画ソフトウェアでは、治療計画を作成していく過程で、CT画像の表示はもちろんのこと、患者臓器形状、線量分布等の3次元データを様々な方法で画面表示する。表示対象のデータは業界標準のDICOM規格 (Digital Imaging and COmmunication in Medicine) に準拠した医用画像ファイルに格納されており、3次元

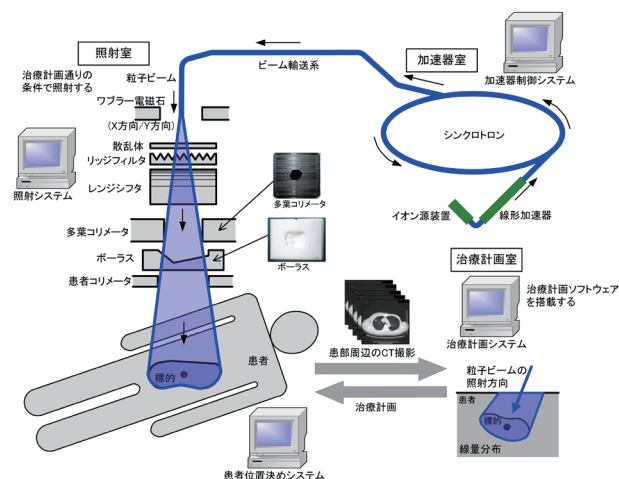


図1 粒子線治療装置の概略

(3D) で高解像度なため情報量は膨大である。この膨大なデータを3Dで表示し、画像のズーム・平行移動・回転、長さを測定するメジャー等の機能を提供することが求められる。

前述した治療計画ソフトウェアを始めとし、粒子線治療装置のような医療の現場で使用されるアプリケーションソフトウェアには、概ねそのユーザインタフェースとして、CT画像、MRI画像などの3次元医用画像を、高精細モニタ上にグラフィカルに表示して操作できることが求められる。ここで、3次元医用画像を診断・治療に生かすためには、診断・治療の視点での有効な表示・操作方法を工夫していくことが必要である。そして、それを実現するための先進的な可視化ソフトウェアの一つとして、VTKが広く知られており、当社での医用画像アプリケーション開発に使用している。これら背景を踏まえて、本報告では、VTKを使用した医用画像アプリケーションの開発技術として、技術的概要、及び開発事例について紹介する。

2. VTKによる可視化アプリケーション構築

2.1 VTKとは

VTKは、2D/3Dデータの可視化を提供するBSDライセンスのOSS (Open Source Software) である。開発言語はC++であるが、他言語 (Java、Python、Tcl) でも使用可能である。VTKが提供するクラスは、データ処理、レンダリング等の役割を持っており、各々の役割を持つオブジェクトを連結して、VTKパイプラインを構築することで可視化を実現する。VTKパイプラインの模式図を図2に示し、各オブジェクトの種類毎に、その役割を説明する^{(1) , (3) , (4) , (5)}。

(1) DataSet

描画対象となる3Dデータ本体。画素値の集合データを格子上に配置するvtkImageDataや、頂点座標列をポリゴンデータとして扱うvtkPolyData等がある。なお、2Dデータは、1ボクセルの厚みを持つ3Dデータとして表現する。

(2) Filter

DataSetに対し、回転・並進移動、データの切断等の加工処理を施して、加工後のDataSetを出力する。複数のFilterを連結することも可能である。

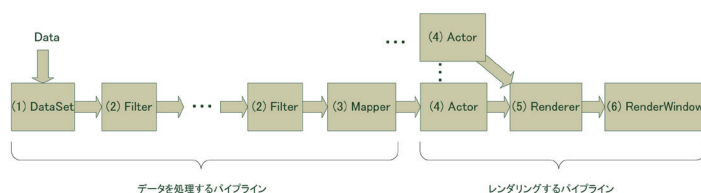


図2 VTKパイプライン模式図

(3) Mapper

DataSetに格納されているデータを、描画時の表現 (色、グレイスケール等) に対応づける。

(4) Actor

描画オブジェクト。描画位置や可視状態等を設定する。条件によっては、Actorで色情報を設定する場合もある。

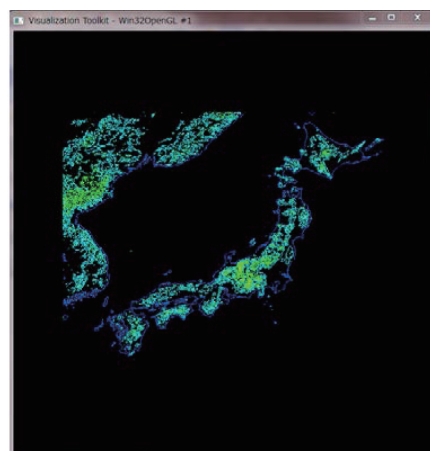
(5) Renderer

描画エンジン。複数のActorをセットし、レンダリングを行う。

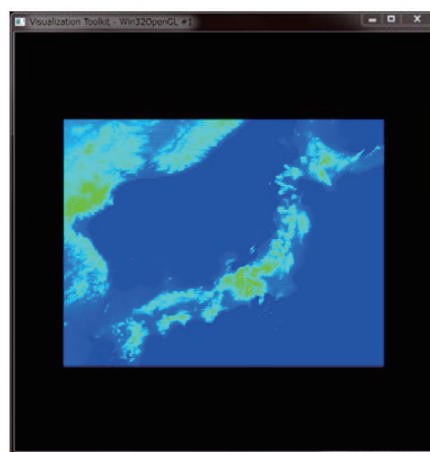
(6) RenderWindow

描画ウィンドウ。Rendererがレンダリングした結果を表示する。

VTKでの描画例を図3に示す。入力データはアメリカ海洋大気庁 (NOAA: National Oceanic and Atmospheric Administration) より取得した日本近海の地形データであり、経度、緯度及び標高からなる⁽⁷⁾。等高線用のFilterやグラデーションを施すためのMapperを用いてパイプラインを構築し、入力データを流すことで図3の表示が可能となる⁽⁶⁾。



(a) 0[m]から3800[m]間に等高線10本設定した場合



(b) -500[m]から3800[m]の範囲でグラデーション設定した場合

図3 VTKでの描画例略

2.2 VTKの特徴

VTKの主な特徴は、以下に説明する3点である⁽³⁾。

(1) 開発の容易性

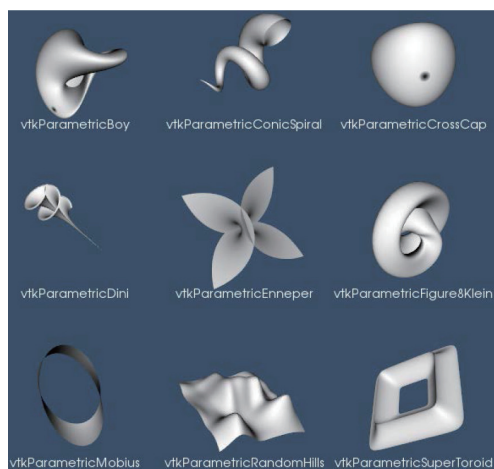
VTKは、3D CGプログラミングに関する高度な専門知識や経験が無くとも容易に使用でき、かつ、多機能である。

(2) 高速な描画

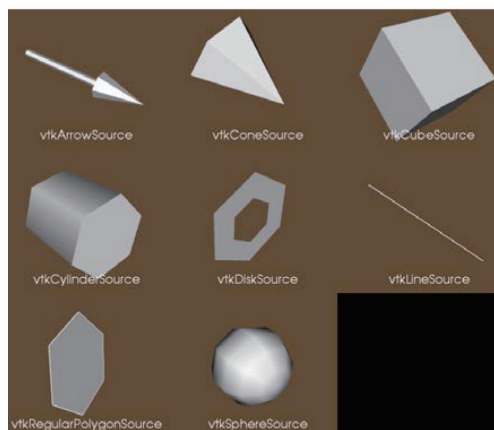
VTKのパイプラインは、要求駆動（demand-driven）で実行される。RenderWindowで出力データが必要になると、入力データの取得を要求する。要求は、VTKパイプライン内をダウストリーム側からアップストリーム側へ遡り、DataSet及びFilterが保持している各データの最終更新時刻を比較する。このことで、VTKはパイプラインのどこから先を再処理すべきかを特定し、特定した起点から終点（RenderWindow）までのパイプラインを「最小限の処理量」で作動させており、これにより、VTKのパイプラインアーキテクチャは、描画の高速性を向上させている。

(3) データ形式の多様性

VTKで扱えるデータ形式は多種多様である。VTKクラスが提供する複雑な3D形状の一部を、図4に示す。



(a) Parametric Object



(b) Geometric Object

図4 VTKが提供する3D形状（一部）⁽²⁾

2.3 VTKの使用例

当社が開発担当している医用画像アプリケーションを題材として、VTKの使用例を説明する。

●対象データ

医用画像アプリケーションでレンダリング対象となる主なデータは、3Dボリュームデータ（CT画像、線量分布）と、2Dベクトルデータ（患者臓器等の輪郭）、の2つに大別される（表1）。これら対象データに対して構築する各VTKパイプラインを図5に、また、構築したVTKパイプラインの実行結果を、各々、図6に示す。

(1) 3Dボリュームデータ（CT画像、線量分布）

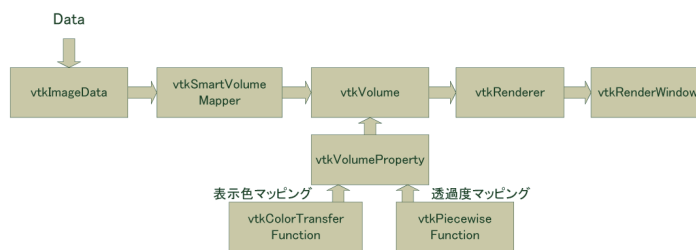
CT画像や線量分布といった3Dボリュームデータは、3次元の格子データであり、vtkImageDataクラスに配置する。表示色及び透過度のマッピングは、vtkImageDataのActorであるvtkVolumeに設定する。

(2) 2Dベクトルデータ（患者臓器等の輪郭）

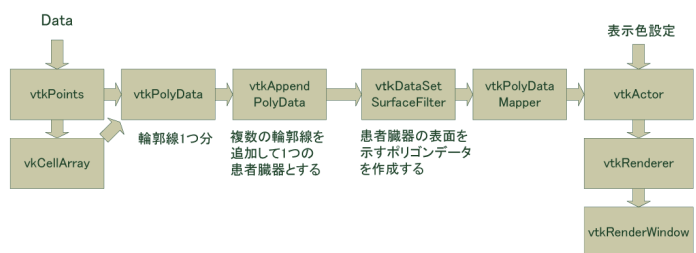
患者臓器等の2Dベクトルデータは頂点座標列（の集合）であり、vtkPoints、vtkCellArrayを経由して

表1 医用画像アプリケーションのレンダリング対象

No.	対象データ	データ例	データの概要	表示色
1	3D ボリュームデータ	CT 画像	スライス枚数分の2D データ	CT 値[HU]のグレースケール表現
		線量分布	線量値枚数分の3D データ	線量値[Gy]のグラデーション（色相）表現
2	2D ベクトルデータ	患者臓器等の輪郭	関心領域を示す頂点座標列データ。3次元の形状を、複数の閉曲線の集合で定義する。	RGB

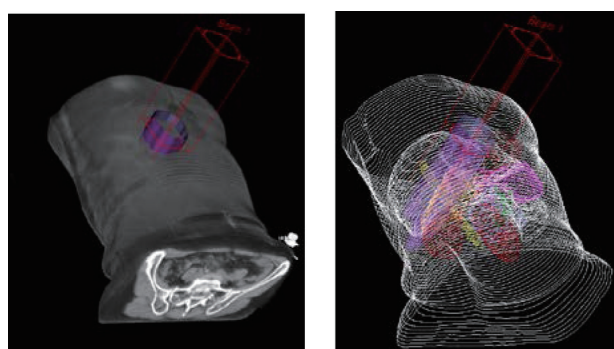


(a) CT 画像



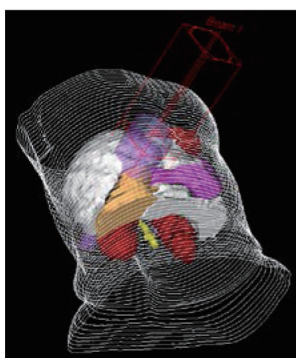
(b) 患者臓器

図5 医用画像アプリケーションのVTKパイプライン



(a) CT 画像と線量分布

(b) 患者臓器と線量分布



(c) 患者臓器の塗りつぶし表示

図6 医用画像アプリケーションの描画

表2 医用画像アプリケーションの主な機能

No.	機能	詳細	使用する VTK クラス
1	MPR 表示	任意の切断面を表示する。	vtkCutter
2	ズーム	マウス操作により拡大・縮小する	vtkInteractorStyleImage
3	平行移動	マウス操作により表示を平行移動する	vtkInteractorStyleImage
4	計測	2 点間を指定し、画面上に 2 点間を結ぶメジャーを表示する。左右の端点はドラッグ & ドロップで自由に移動可能とする。	vtkDistanceWidget
5	患者アイコン	患者がどの方向を向いているかを示すアイコンである。VRML 2.0 で記述したアイコンを表示する。アイコンはウィンドウ上にサブウィンドウ化して表示する。	vtkVRMLImporter vtkOrientationMarkerWidget

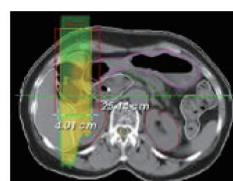
vtkPolyDataクラスに配置する。この時点では、輪郭線データであるが、vtkDataSetSurfaceFilterを利用すると、表面を塗りつぶしたデータに加工することができる。表示色は、vtkPolyDataのActorであるvtkActorに設定する。

●主な機能例

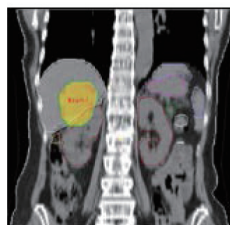
医用画像アプリケーションで実現した主な機能を表2に示す。VTKを使用することで、対象データのレンダリング及び、これら主要な機能を、効率良く実装できる。

(1) MPR表示

MPR (Multi-Planar Reconstruction) 表示は、vtkCutter



(a) Axial 表示



(b) Coronal 表示



(c) Sagittal 表示

図7 MPR表示機能

を利用して実装する。CT画像、患者臓器、線量分布の3データから任意の切断面を得ることができる (図7)。

(2) ズーム、平行移動

ズーム、平行移動等、画像に対するマウス操作の実装には、vtkInteractorStyleImageを利用する。vtkInteractorStyleImageに不要なマウス操作がある場合は、vtkInteractorStyleImageを改変して利用すればよい (BSDライセンスのため自由に改変することができる)。

(3) 計測

計測機能において、左右の端点をマウスで掴むような操作インタフェースは、vtkDistanceWidgetを利用して実装する。

(4) 患者アイコン

患者アイコンは、VRMLというマークアップ言語で作成するが、vtkVRMLImporterを利用することで、VRMLファイルを、VTKのオブジェクトとして読み込むことが可能である。VTKでは他にDEM (Digital Elevation Model)、STL (Standard Triangulated Language) 等の数値形状データにも対応している。さらに、vtkOrientationMarkerWidgetを利用することで、医用画像の表示領域とは独立して、患者アイコンを自由に移動させることができる。

3. 医用画像アプリケーションの開発事例

VTKを用いた開発事例として、医用画像アプリケーションの事例を3件紹介する。

3.1 患者臓器の隠面処理

●目的

患者臓器に個別に透過度を設定して重畳表示する。

●課題

過去の開発では、一般的に使われるDepth Sort と呼ばれるアルゴリズムにより隠面処理をしてきたが、重なり合う患者臓器を見せるために透過度を変更、調整すると、見た目が自然に見えないという課題があった。

●手段

VTKは、2種類の隠面処理アルゴリズムを提供しており、より自然に見える後者のアルゴリズム (Depth Peeling) を使用することができた。前者のアルゴリズム (Depth Sort) は比較説明のために言及しておく。

(1) Depth Sort

Depth Sortは、画面上に表示する全描画対象を、カメラ視点から遠い距離にあるデータから順に描画して、アルファブレンディングしながら重ね合わせていく手法である (図8 (a))。患者臓器同士がサイクリックな前後関係になっていると、若干の不整合が見られ、レンダリング速度が低下する。

(2) Depth Peeling⁽⁸⁾

Depth Peelingは、画面上に表示する全描画対象の、カメラ視点から最も前方の面を取得して (Layer 0)、次にその面よりも後方にある描画対象のうち、最も前方の面を取得する (Layer1) 処理を繰り返すことで、重畳する描画対象を1層ずつ剥がしていく (Peeling) 手法である (図8 (b))。本手法は、vtkRenderer及びvtkRenderWindowでDepth Peelingの設定を有効にすることで実現する。VTKパイプラインは、特別なVTKクラスを使用しないため図示しない。vtkRendererに、Depth Peeling有効化、Depth Peelingの性能パラメータ (ピーリング対象の最大レイヤ数、レンダラ領域における最も

表面側にあるレイヤが占める割合指定) を設定する。vtkRenderWindowに、アルファビット有効化を設定する。Depth Sortで見られたような不整合は見られず、レンダリング速度は低下しない。

●結果

隠面処理アルゴリズムDepth Peelingの適用前後の実行結果を図9に示す。隠面処理適用前には、患者の体輪郭上部が透けて縞模様の乱れが見られる。一方、隠面処理適用後には、患者の体輪郭の全体が指定した透過度で正しく表示できている。すなわち、本事例の課題は、Depth Peelingを使用することで、課題を解決できた。

3.2 複数画面表示における同一データの処理共有化

●目的

異なる線量分布をビーム毎に表示する画面を作成する。CT画像、患者臓器は同一内容とする。

●課題

過去の開発では、画面毎に、CT画像、患者臓器、及び線量分布を表示するためのVTKパイプラインを構築してきたが、以下の問題点が発生した。

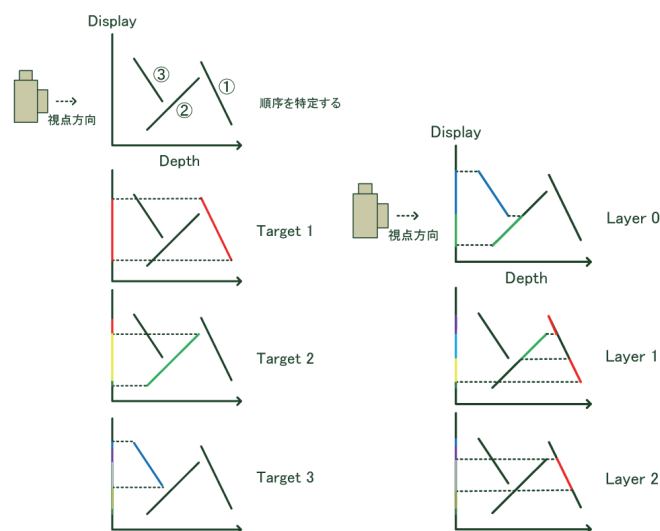
- (1) 画面数分のデータを処理する必要があり、動作速度の遅延に繋がる。
- (2) 画面数分のメモリが必要となる。

●手段

CT画像及び患者臓器のVTKパイプラインを、複数画面で共有することで、動作速度の向上及びメモリ節約を図る。共有するVTKパイプラインはDataSetからActorまでとし、画面毎に作成するRendererに対して、同一のActorを設定することで共有化できる。この機能を実現するVTKパイプライン及びクラス構造を図10、図11に示す。

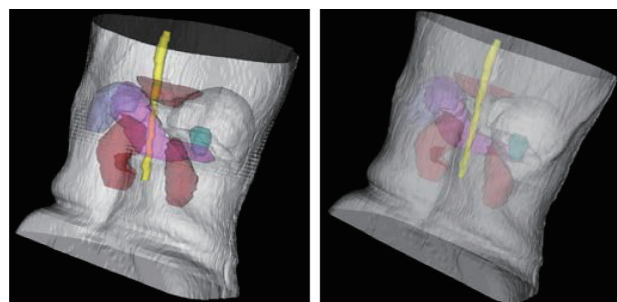
<初期化処理>

- (1) DataSetからActorまでのVTKパイプラインを構築・管理するクラスを、画面とは独立した部品として作成する。図中では以下のクラスに該当する。



(a) Depth Sort (従来手法) (b) Depth Peeling (使用手法)

図8 隠面処理アルゴリズムの模式図



(a) Depth Peeling 適用前

(b) Depth Peeling 適用後

図9 患者臓器の重量表示

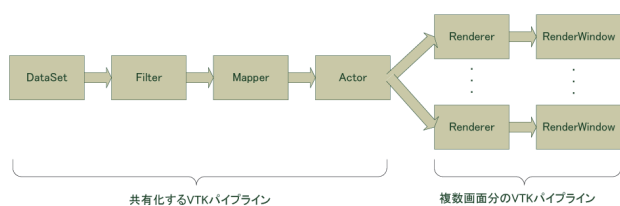


図10 VTKパイプライン共有化

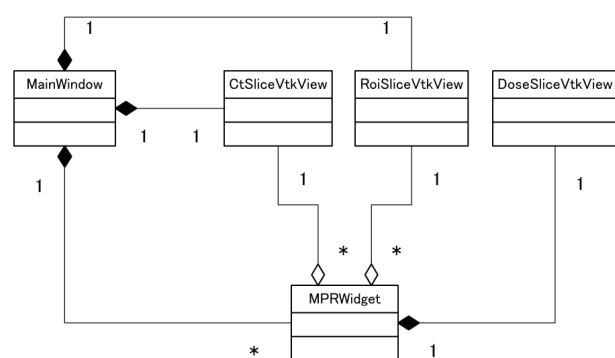


図11 VTKパイプライン共有化におけるクラス構造

- ・CtSliceVtkView…CT画像
- ・RoiSliceVtkView…患者臓器
- ・DoseSliceVtkView…線量分布

- (2) アプリケーション起動時に、上記3クラスにてVTKパイプラインを入力データ無しで構築する。
- (3) 画面を構成するクラスMPRWidgetでvtkRendererを作成して、(2)で構築したActorを設定する。ただし、CT画像及び患者臓器は1つのActorを共有する。

<更新処理>

- (1) VTKパイプライン入口のDataSetに更新後のデータを入力する。CT画像及び患者臓器については、MPRWidgetを統合するMainWindowから、CtSliceVtkView及びRoiSliceVtkViewに更新を通知する。線量分布については、MainWindowから各MPRWidgetを介して、DoseSliceVtkViewに更新を通知する。
- (2) DataSetにデータ入力後、DataSetの最終更新時刻を更新する（Modified関数の実行）。
- (3) 各MPRWidgetでレンダリングを開始する（vtkRenderWindow::Render関数の実行）。

●結果

VTKパイプラインを複数画面で共有することで、画面数に比例して増加していたDataSetからActorまでのデータ処理及び必要メモリを、1つに抑えることに成功して、動作速度及びメモリ使用量が大幅に改善した（図12）。本事例では、同一内容の表示であるためDataSetからActorまでを共有化したが、例えば、データの加工処理は同一であるが、画面毎に表示色と可視状態を変更

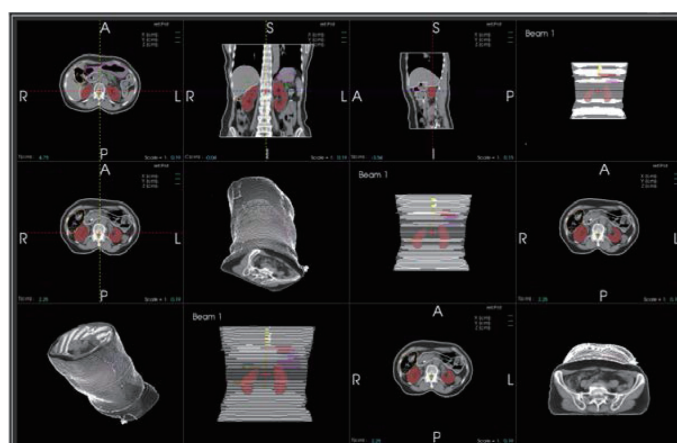


図12 複数画面の医用画像表示（最大12画面）

して表示する場合、DataSetからMapperまでの処理を共有化すれば、同様の動作改善が期待できる。

3.3 患者臓器を構成する輪郭線の編集

●目的

患者臓器の形状を構成する輪郭線の編集機能を作成する。治療計画を立てるためには、ビーム照射の標的となる患者臓器（輪郭線）を何度も編集して最適にする必要があり、ユーザビリティの高い輪郭編集機能が求められる。

●課題

編集操作は、VTK画面上に描画される輪郭線をマウスクリックすることで開始して、再度のクリックで結果確定する。この機能には、以下の課題があった。

- (1) マウスイベント（Press、Move、Release）を受信すること
- (2) 画面上にリアルタイムで輪郭線を描画して表示すること
- (3) 編集後の輪郭線が、閉じた輪郭（閉輪郭線）となること

●手段

- (1) マウスイベント受信
 - (a)コールバックを受信するクラス（以降、CB受信クラスとする）を、vtkAbstractWidgetを継承して作成する。
 - (b)CB受信クラスとVTKの画面（vtkRenderWindow）をvtkRenderWindowInteractorでリンクする。
 - (c)CB受信クラスがメンバで保持しているvtkWidget CallbackMapperクラスを利用し、vtkCommandに定義されているマウスイベント3種類を設定する。
- (2) リアルタイムでの輪郭線描画
 - (a)マウスイベント（Press）のコールバックを受信すると、マウスカーソルの座標から最も近傍の輪郭線上の点を決定して、編集の開始点とする。

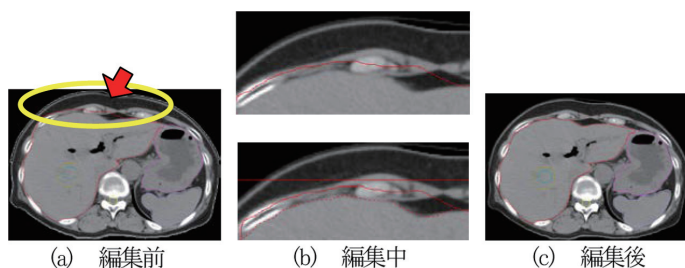


図13 輪郭線の編集機能

- (b)マウスイベント (Move) のコールバックを受信すると、マウスカーソルの座標を記録する。記録した座標列を入力データとして、vtkPoints、vtkPolyLine、vtkCellArrayの3種類のVTKクラスを使用して輪郭線を表示する。
- (3) 閉輪郭線での描画
 - (a)マウスイベント (Release) のコールバックを受信すると、マウスカーソルの座標から最も近傍の輪郭線上の点を決定して、編集の終了点とする。
 - (b)輪郭線を編集後の状態に更新して再描画する。

●結果

輪郭線編集機能の実行結果を図13に示す。図に赤線で示すLiver (肝臓) を編集対象としている。VTKを使用することにより、マウスイベントの受信、画面上の輪郭線描画の仕組みを簡便に実装できて、ライン描画のレンダリングもスムーズに動作した。

4. 今後の展望

当社では、長年に渡り、VTKを用いた医用画像アプリケーションソフトウェアを開発してきた。その経験より、様々な医用画像をレンダリングする技術、医用画像をユーザが望むあらゆる形式に加工する技術、医用画像の表示に補助的な情報を付与する技術等、VTKの技術ノウハウを数多く蓄積してきた。得られた技術ノウハウは、将来の医用画像アプリケーションソフトウェアの発展、機能の拡張に適合可能なものであり、また、医用画像だけでなく、他の3次元データを扱うソフトウェアにも十分に適合できるものであると考え。今後も、VTK及び他ツールを含めた最新技術を取り込むことにより、さらに効果的かつ効率的な開発手法を確立していくことで、ユーザの要求に常に応え続けるソフトウェアを開発する。

5. むすび

本報告では、当社が開発に従事したVTKを使用した医用画像の可視化技術の技術的概要について述べた。当社としては、医用画像アプリケーションの業務実績を重ねるとともに、本可視化技術に限らず今後ともドメインで求められる技術を追求して、将来の粒子線治療の更なる発展に貢献できるように取り組んでいきたい。最後になるが、本報告の作成においては、三菱電機電力システム製作所の皆様には、深く感謝するものである。

参考文献

- (1) VTK : <http://www.vtk.org/>
- (2) VTK Example : <http://www.vtk.org/Wiki/VTK/Examples>
- (3) VTK Architecture : <http://www.aosabook.org/en/vtk.html>
- (4) VTK Pipeline : http://www.training.prace-ri.eu/uploads/tx_pracetmo/vtkPipeline.pdf
- (5) VTK Mailing Lists : <http://www.vtk.org/mailing-lists/>
- (6) http://www.simulation-studies.org/wp-content/uploads/2014/01/vtk_semi_comp.pdf
- (7) NOAA : <http://www.ngdc.noaa.gov/mgg/global/global.html>
- (8) Depth Peeling : http://www.eng.utah.edu/~cs5610/handouts/order_independent_transparency.pdf

執筆者紹介

田中 真幸

2011年入社。関西事業部第五技術部所属。カーマルチメディア開発、医用画像システムのソフトウェア開発に従事。

馬場 明子

2005年入社。関西事業部第五技術部所属。カーマルチメディア開発、医用画像システムのソフトウェア開発に従事。

山本 啓二

1992年入社。関西事業部第五技術部所属。数値シミュレーションや顔認識システム、医用画像システム、医用画像処理などの各種アルゴリズム開発及びソフトウェア開発に従事。