

A Visualization Cloud Service for X-CT DICOM Images Applied to Deep-sea Drilled Core Database

Yuichiro Sakamoto*, Shin Sasaki*, Takaya Okamoto*, Shinichi Yamagiwa†, Toshio Hisamitsu‡ and Koichi Wada†

*Department of Computer Science, Graduate School of Systems and Information Science, University of Tsukuba

1-1-1 Tennodai, Tsukuba, Ibaraki, 305-6573, JAPAN

Email: { sakamoto, sasaki, okamoto } @iplab.cs.tsukuba.ac.jp

†Faculty of Engineering, Information and Systems, University of Tsukuba / JST PRESTO, JAPAN

1-1-1 Tennodai, Tsukuba, Ibaraki, 305-6573, JAPAN

Email: { yamagiwa, wada } @cs.tsukuba.ac.jp

‡Sciences Services Group, Kochi Institute for Core Sample Research (JAMSTEC)

Nankoku City, Kochi, 783-8502, JAPAN

Email: hisa@jamstec.go.jp

Abstract—The deep-sea drilling vessel named as “CHIKYU” produced by JAMSTEC has riser coring system of which ability enables to keep drilling into thousands meters below deep seafloor safely and continuously. The drilled sample consists of sediment and rock obtained from the seafloor is generally called as a “core sample”. One of the special equipments onboard is an X-ray CT scanner to investigate internal structures of the core sample visually. The scanned data including 3D information saved into the DICOM format and stored as digital archive of core samples. The digital archive gives many opportunities to researchers exploring micro-structures, vein minerals, sedimentary structures and study of mega-thrust at the tectonic-accretionary prism. All people can find and download original DICOM file of the core sample via web site of the Kochi Institute for Core Sample Research, JAMSTEC. A file size of the DICOM file for 150 cm of a core section, which is the general length of IODP core section, becomes more than 400 MByte that have been bringing difficulties to users who deal with many core section images. To resolve the drawback for dealing with the heavy DICOM files of the core samples, we need to develop a novel system for 3D rendering of DICOM file that can be accessed via the internet. This paper introduces design and implementation of the system to solve the visualization difficulties of the core samples in the geology research filed. We targeted to implement a cloud service for visualizing the core samples using simple mobile terminals such as the ones with a standard web browser or an Android application. We have developed a remote rendering service that constructs 2D images from the slice image data set of DICOM processed in a GPU cluster environment, and transfers the 2D image saved in a common image format to the client side. The service also provides various visualization functionalities to browse the core samples such as cutting, rotating, coloring etc. remotely processing the DICOM image data.

I. はじめに

JAMSTEC 所有の地球深部探査船「ちきゅう」は、人類史上初めてマントルや巨大地震発生域への大深度掘削を可能にする世界初のライザー式科学掘削船である。統合国際深海掘削計画 (IODP) の主力船として、今日も科学研究航海を行っている [1]。海底を掘削することにより、地球環境変動、地球内部構造及び地殻内生物圏の解明を目的としている。海底より掘削された柱状の地層サンプルは掘削コア試料と呼ばれ、「ちきゅう」船上にはこの掘削コア試料を分析

するための様々な設備が搭載されている。他の掘削船にない「ちきゅう」の特徴的な設備として、X線CTスキャナがある。掘削コア試料をこのX線CTスキャナを用いてデジタルデータ化することにより、コア試料の物性や構造が変化してしまう前に、その内部を可視化することができる。この掘削コア試料のX-CTスキャンデータ(以下、コアデータ)を用いて予めコア内部を調査をしておくことで、実際のコア試料を用いた分析の効率や効果を高めることができる。そのため、掘削コア試料を用いた研究には、コアデータの活用が非常に重要となる。掘削された実物のコア及びコアデータは高知コアセンターにおいて保管・管理されるが、コアデータは同センターのVirtual Core Library [2]に公開されており、自由にダウンロードし、利用することが可能である。

しかしながら、コアデータを用いた分析にはユーザの負担が大きく、主に2つの課題がある。1つ目はそのデータサイズである。コアデータは1つのコアセクション(1.5m)あたり400メガバイトあり、これをPCにダウンロードし、3次元描画するためには高速なプロセッサと大容量のメモリ、そして通信回線に高いバンド幅を必要とする。もう1つは、掘削コア試料を閲覧するための専用ソフトウェアが存在しないことである。コアデータは医用画像の標準フォーマットであるDICOMであるため、DICOMファイルを読み取れる専用ソフトウェア(OsiriX等)を用いれば3次元化が可能である。しかしDICOMビューワは医用であり、骨や筋肉、靭帯などの生体情報にマップされたCT値の閲覧は容易であるが、コアを分析する際にはユーザがコアの構造を表現するCT値に対して適切な色付けを行い、可視化しなければならない。

本研究は、これらの課題を解決することが目的である。前述の1つ目の課題を解決するために、3次元描画処理サーバの構築を行った。コアデータを軽量な2次元画像としてユーザへ返すことにより、端末性能に依存せず、かつ、低速な回線でも利用可能なシステムとなる。2つ目の課題を解決するために、掘削コア試料分析のためのクライアントアプリケーションを開発をした。これは、CT値に自由な色付けやコア試料の切断面の閲覧等のコアに対する操作や、何千とあるコア試料を選択するためのインタフェースを有する。

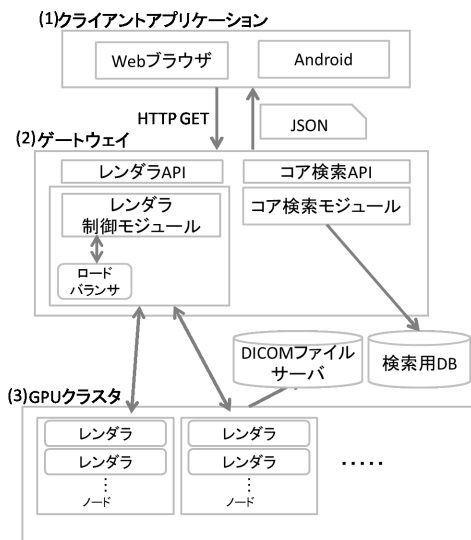


Fig. 1. システム構成

本研究では、携帯端末からの利用を想定した Android アプリケーションと、PC からの利用を想定した Web アプリケーションの 2 つを開発した。

II. システム概要

A. システム構成

本サービスを実現するために、(1) クライアントアプリケーション、(2) ゲートウェイ、(3) GPU クラスタから成るシステムを設計した。その構成を Fig. 1 に示す。

1) クライアントアプリケーション: コアを閲覧するためのユーザインタフェースを提供する。ユーザからの要求に応じて、レンダラ API 及び検索 API を利用してコアの情報や画像を取得する。

2) ゲートウェイ: クライアントアプリケーションからの要求を初めに受け付けるサーバである。その中に、コアの情報取得するためのコア検索モジュールと、コアの画像取得するためのレンダラ制御モジュールを有する。コア検索モジュールは、クライアントからのクエリに一致するコア情報を検索用 DB から選択する。レンダラ制御モジュールはクライアントから、選択されたコア番号と、コア画像に対する操作の情報を受け取り、GPU クラスタにあるレンダラに処理を依頼する。ただし、レンダラに対する依頼はロードバランスを介して行われる。ロードバランスは、負荷の少ないノードを選択し、レンダラにコア画像生成処理を依頼する。

3) GPU クラスタ: 画像の生成を行うレンダラが稼働するノード群である。DICOM ファイルサーバに格納されているコアデータを読み込み、レンダラ制御モジュールから受け取ったパラメータをもとに画像を生成する。

B. ハードウェア構成

システムのハードウェア構成を以下に記述する。ゲートウェイは CPU(Core i7 930 2.8GHZ)、メモリ 12GB で構成され、クラスタは CPU(Intel Xeon E5645 2.40GHZ (6 cores)) ×

2、GPU(Tesla M2050 1.55GHZ) × 2、メモリ 12GB で構成されたノードが計 16 ノードで構成されている。ゲートウェイ、GPU クラスタ間は Infiniband QDR で接続されている。

III. クライアントアプリケーション

クライアントアプリケーションとして、Android 版と Web ブラウザ版の 2 種類を作成した。本研究の目指すシステムに必要な機能は、大きく分けて以下の 4 つである。

1) 検索機能: 数千セクション存在するコア試料の中から、航海番号や掘削サイト番号を選択し、閲覧したいコアを特定する。

2) 回転、拡大・縮小: 3 次元化されたコアを閲覧するための基本的な操作である。コアの全体像や細部表示、任意の視点からの閲覧を可能とする。

3) 縦方向の切断: 実際の掘削コア試料を用いた分析では、コアの切断面を観察することによりその地質学的な特徴を把握する。これを仮想的に行うインタフェースを提供する。

4) CT 値への色付け、透過設定: 従来の DICOM ビューアは医用画像(人体)を閲覧するために特化している。本アプリケーションでは、掘削コア試料の分析に適した CT 値に対する色付け・透過設定をできるインタフェースを装備する。

A. Android 版ビューア

前述の機能を実現するため、START 画面、INFO 画面、WHOLE 画面、CUT 画面、COLOR 画面の 5 画面を設計した。START 画面を除く 4 画面は画面上部のタブにより画面の切り替えを行うことができる。5 つの画面それぞれについて詳細を後述する。Fig. 2 は Android タブレットの REGZA tablet AT570 (7.7 インチ: 解像度 1280 × 800) で 5 画面それぞれを表示させた様子である。

1) コア画像に対する操作: COLOR 画面、CUT 画面、INFO 画面の左側と WHOLE 画面において、コア画像が表示されている領域でドラッグ操作やピンチ操作を行うことで、回転、拡大・縮小が可能となる。コアの回転は、ドラッグ操作により実行される。上下方向のドラッグ操作が前後方向(手前、奥行)の回転、左右方向のドラッグが左右方向の回転に対応している。ドラッグの開始点と終了点の長さで回転角度を決定する。コアの拡大・縮小は、ピンチイン動作で縮小、ピンチアウト動作で拡大が実行される。ピンチ動作の大きさによって、拡大率、縮小率を定めている。拡大・縮小の最大値、最小値については制限を設けている。また、表示されている画像の実際の縮尺を、画像の右下に表示している。COLOR 画面、CUT 画面、INFO 画面の左側に表示されるコア画像は、WHOLE 画面に表示されるものと同じ画像である。色付けを変更する、縦切断を行う等の操作を行った場合すべての画面に表示されているコア画像が同期して変化するようにした。

2) コアの絞り込み検索: 航海番号、掘削サイト番号等のキーを基にしてコアの検索を行う。検索 API を使用し、航海番号、掘削サイト番号、ホール番号、コア番号、ビットタイプ、セクション番号の順に絞り込み検索を行い、表示するコア試料を指定することが可能である。

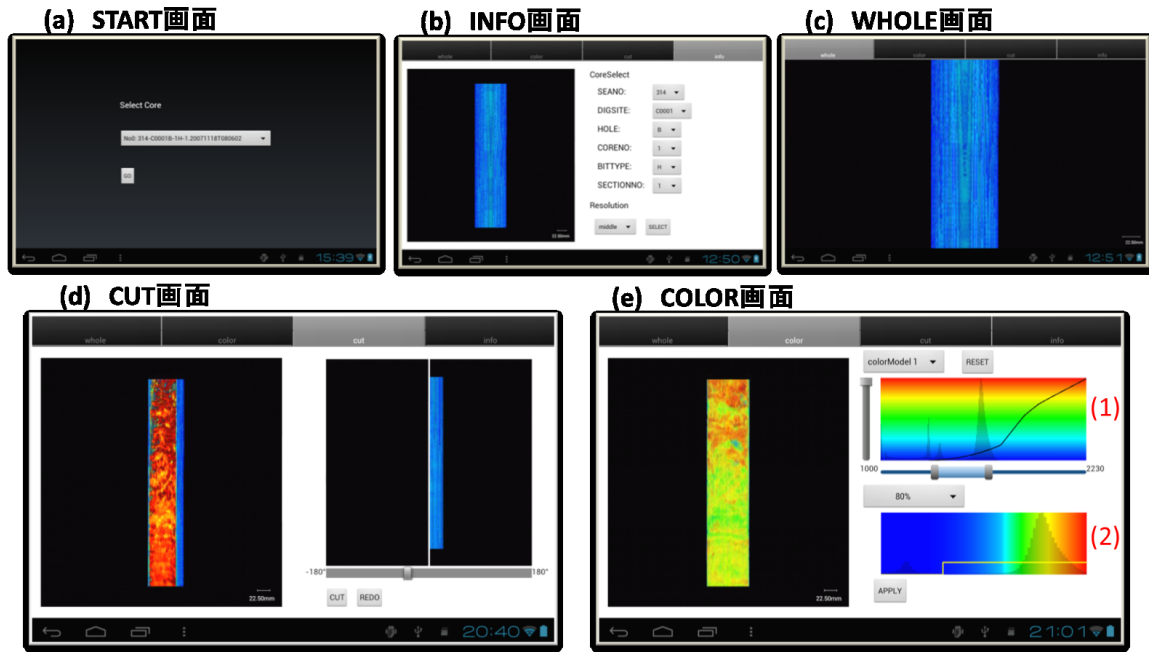


Fig. 2. Android 版ビューアの画面

3) コアの切断: コアの縦方向の切断を行う。Fig. 2(d)の右は切断対象のコア画像であり、中央の白線に沿って切断することを示している。切断面の方位角調整はスライドバーで行う。具体的には画像における切断面の水平方向の回転角度を指定する。角度の指定後、CUT ボタンを押すと Fig. 2(d)の右の画面中央の線より左側を切断したコア画像を表示する。UNDO ボタンを押すことにより、操作の取り消しができる。Fig. 2(d)の左は縦切断を行った CUT 画面の様子である。

4) コアの CT 値に対する色付け・透過: Fig. 2(e) はコアの色付け、透過度を変更する COLOR 画面である。CT 値に対する色付けは、対象値の範囲の指定と、CT 値と色の関数を指定することにより適用される。ユーザは、Fig. 2(e) のヒストグラムを参考に調整できる。ヒストグラムは、横軸に CT 値、縦軸にデータのボクセル分布数を示す。Fig. 2(e)(1)には画像の CT 値が 0~4095 の範囲で分布しているヒストグラムを重ねて表示し、Fig. 2(e)(2) に色付け調整範囲を指定されたヒストグラム表示している。これらの計算はレンダラで処理している。色値には、一般的な RGB ヒートマップ関数を用いている。CT 値と色付けの関数には 3 次スプライン曲線を用い、Fig. 2(e)(2) 上にマッピングしている。カラーマップのヒストグラム領域内でタッチした点を 3 次スプライン曲線の補間点とし、最大 8 箇所指定できる。補間点を一点も選択していない場合は直線的なマッピングを行う。Fig. 2(e)(1) は、1000~2230 間の CT 値を 2 点の補間点を用いた 3 次スプライン曲線でマッピングしている具体例である。スライドバーにより色付けを行う CT 値の最小値、最大値を選択することができる。両つまみの間をタッチし、スライドさせることで、CT 値の幅を固定したまま色付け対象範囲を移動することができる。

透過度を設定するには、Fig. 2(e)(2) の透過マップを操作

する。透過度の指定方法は 2 種類ある。1 つは、11 段階 (0~10) のリストから任意の透過度を選択する方法である。2 つ目はコア画像の透過度を区間別に指定する方法である。透過マップをタッチで操作することにより一定区間ごとに透過度を設定することができる。透過マップは縦軸を透過度、横軸を CT 値とし、カラーマップによって対応付けられた CT 値にマッピングしている。現在は個別に設定できる透過度の区間を 10 区間としている。Fig. 2(e)(2) の例は青の部分の透過度を 100% に、その他の部分の透過度を 80% に設定した様子である。

B. Web ブラウザ版ビューア

Web ブラウザ版の画面を Fig. 3 に示す。基本的な機能は Android 版と同じであるが、いくつか異なる箇所がある。マウス操作を想定しているため、回転や拡大・縮小操作の実行をボタンに割り当てている。また、検索画面と操作画面を 1 画面に表示している。

IV. ゲートウェイ

ゲートウェイは、クライアントアプリケーションからのコア情報やコア画像の要求を受け付けるためのコア検索モジュールとレンダラ制御モジュールおよびロードバランサから成る。レンダラ制御モジュールは、GPU クラスタでコア画像生成をするレンダラとの間でコア画像生成に関するリクエストおよび、画像情報を交換する。ロードバランサは、レンダラの起動や管理および通信を制御する。

A. コア検索モジュール

コアデータに含まれる航海番号、採掘サイト番号、ホール番号、コア番号、ビットタイプ、セクション番号の 6 つのパラメータがあれば、コアを一意に特定することができる。これらの値を GET リクエストによりパラメータをクラ

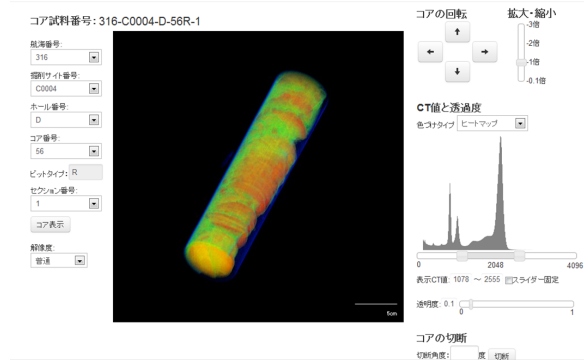


Fig. 3. Web ブラウザ版ビューア

TABLE I
コア検索 API のパラメータと取りうる値

パラメータ	意味と値
seano	航海番号 . 3 桁の正の整数
digsite	探掘サイト番号 . アルファベット 1 文字 + 数字 4 桁
hole	ホール番号 . アルファベット 1 文字
coreno	コア番号 . 正の整数
bittype	ビットタイプ . アルファベット 1 文字
sectionno	セクション番号 . 正の整数が文字列「CC」

クライアントアプリケーションが送信し、JSON フォーマットのデータをクライアントアプリケーションに返す。API に送るパラメータを 1 つずつ追加することにより、絞り込み検索ができる。TABLE I に、API に送信できる 6 つのパラメータの詳細を示す。TABLE II は、API に送信するパラメータ数に応じて取得される返り値の一覧である。返り値を随時 API のリクエストに追加していくことで、コア試料を一意に特定できる。

B. レンダラ制御モジュール

コア検索モジュールと同様に、GET リクエストによりパラメータをクライアントアプリケーションが送信し、生成した画像情報を含む JSON フォーマットのデータをクライアントアプリケーションに返す。実際にレンダラへの画像生成の依頼を行う際は、後述のロードバランサを介して行われる。リクエストのパラメータを TABLE III に、返り値を TABLE IV に示す。

C. ロードバランサ

ゲートウェイに接続されている GPU クラスタには計 16 個のノードが存在する。各ノードで起動しているレンダラ

TABLE II
コア検索 API の返り値

送信するパラメータ	返り値
seano	digsite
seano, digsite	hole
seano, digsite, hole	coreno
seano, digsite, hole, coreno	bittype
seano, digsite, hole, coreno, bittype	sectionno
seano, digsite, hole, coreno, bittype, sectionno	coreid

TABLE III
レンダラ API のパラメータと取りうる値

パラメータ	意味と値
corenum	検索 API で調べたコア番号 . 0 以上
rotx, rotz	前後回転, 縦軸回転 . 0-360
zoom	拡大率 . 0.0-
depth	コアの上下移動 . 符号付整数
resolution	解像度 . 5 段階
cf	色関数 . CT 値と色値の組の配列
af	透過関数 . CT 値と透過値の組の配列

TABLE IV
レンダラ API の返り値

送信するパラメータ	返り値
img_url	生成された画像の URL
histgram	CT 値のヒストグラムの配列
scale	1 ピクセルあたりの長さ

によって、コア試料の 2 次元画像が生成されるが、画像生成の処理にかかる負荷は高く、1 つのノードにリクエストが集中するとサービスの悪化を招く。そのため、ロードバランサによるロードバランサを実装した。利用可能なノードを順番に選択し、処理を割り振ることで 1 つのノードに処理が集中することを防ぐ。Fig. 4 のように、各ノードを順に 1 つずつ選択し、コア試料の 2 次元画像を生成するためのレンダラを起動する。16 番目のノードでレンダラを起動後、再び 1 番目のノードから順にレンダラを起動する。SSH 経由でリモートにレンダラを起動する。コア試料の 2 次元画像を生成するリクエストを受け取った時にロードバランサがレンダラを起動する。1 種類のコア試料につき、レンダラが 1 つ起動される。そのため、3 種類の画像が要求された場合、それぞれ計 3 つのレンダラが異なるノードに起動される。ロードバランサとレンダラの間にはソケット通信を用いる。ソケット通信を行う際に用いるレンダラのポート番号は、レンダラを起動する際に 50000 ~ 65535 番の間で順次割り振る。クライアントアプリケーションからのリクエストを受け取ったタイミングで、そのリクエストに指定されたコア試料のレンダラの起動状況を確認する。起動していない場合、新たにレンダラを起動する。既に起動している

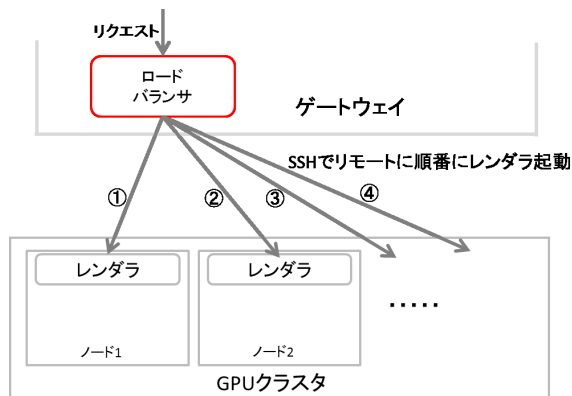


Fig. 4. ラウンドロビンによるレンダラの起動

場合は、そのレンダーラに対してソケット通信で画像生成リクエストを送る。これらの手順は、レンダーラの起動状態を記録した「起動ログ」を参考にし、その内容はコア試料番号、GPU クラスターのノード番号、ポート番号、レンダーラへのアクセス日時をコマ区切りで記述している。ロードバランサがこのログを読み込むことでレンダーラの起動状況を確認し、ある一定のタイムアウト条件を満たすレンダーラを終了処理させる。これはレンダーラによる不要なメモリ占有を避けるためであり、30 分間アクセスがない場合はレンダーラに対して終了メッセージを送る。

V. GPU クラスター

GPU クラスターでは、レンダーラがノードで実行される。ロードバランサによりレンダーラが起動されると、まず初期化処理としてコアデータの読み出しが行われる。次に、描画パラメータの受信待ち状態に入る。描画パラメータを受信すると、描画処理を開始する。画像を生成し終わると、描画結果を送信し、再び受信待ち状態に遷移する。

A. レンダーラの描画処理

レンダーラは、起動時に引数として与えられたディレクトリのパスに保存されている複数の DICOM スライス画像を GDCM (Grassroots DICOM library) [3] を用いて読み出す。DICOM に格納されているデータには 2 種類あり、スライス画像データとその画像に関するメタデータである。初期化処理では、この画像データとメタデータを用いて、ボリュームデータを作成し、3 次元モデルを描画する。

描画処理では、(1) 視点の決定、(2) ボリュームデータの圧縮、(3) CT 値に対する色の設定、(4) CT 値に対するアルファ値の設定、(5) 縦方向切断の各処理を適用し、2 次元画像を生成する。初期化時に読み込んだボリュームデータを 3D テクスチャデータに変換し、複数枚重ねた長方形ポリゴンに対してテクスチャマッピングをすることにより、ボリュームレンダリングを実現する。DICOM のメタデータから長方形ポリゴンの縦横及び間隔を計算し、実物のオブジェクトの縦・横・高さの比を保っている。オブジェクトは、その重心がオブジェクト座標の原点と一致するように配置する。

1) 視点の決定: コアは円柱状であるため、側面と上面・底面の閲覧を考えれば良い。側面は、コアの垂直軸に沿った回転により側面を閲覧することが可能となり、水平軸の回転により上面と底面の閲覧が可能となる。今回の実装では、デフォルトの視点を y 軸上の+から原点方向とし、回転を z 軸回転 (垂直軸回転)、x 軸 (水平軸回転) の順序で適用することとした。

2) ボリュームデータの圧縮: 1 つのコアのボリュームデータは最大で約 $512 \times 512 \times 2000 \times 16\text{bit} = 1\text{GB}$ である。これをそのままテクスチャデータとして用いる場合、描画が終わるまでに長い時間を要する。描画処理の時間短縮のため、処理するボリュームデータの圧縮を行う。今回の実装では、ボリュームデータの解像度を 5 段階 (データ量 1/1, 1/8, 1/64, 1/216, 1/512) 用意し、ユーザの求める品質、応答速度に応じて選択できるようにした。圧縮には、平均画素法を用いた。例えば、 $3 \times 3 \times 3$ ボクセルが $1 \times 1 \times 1$ ボクセルに圧縮されるとき、元のボクセル値の平均値を新たな値として用いる。この手法を用いることにより、単純にボクセルを間引いた圧縮手法に比べて、画像の粗さを抑えることがで

きる。また、ズームの度合いに合わせて読み込むスライス数を変更している。例えば、ズームアップした場合には描画範囲は少ないにも関わらず 2000 枚分の全データを扱うことを避けるため、ズーム倍率の逆数 $\times 512$ スライス分の領域を取り扱う。

3) CT 値に対する色の設定: 本システムで利用する CT 値の範囲は 0 ~ 4095 である。従って、色深度に 12bit を割り当てることとし、RGB (Red, Green, Blue) 値にそれぞれ 4bit を配分した。

4) CT 値に対するアルファ値の設定: CT 値に対するアルファ値は 15 段階 (4bit) で設定できる。CT 値とアルファ値の組列を指定し、点列を線形補間することによりアルファ値関数を生成する。

5) 縦方向切断: 縦方向の切断は、OpenGL のクリッピング関数を用いて実現する。クリッピング処理により、オブジェクト座標における $Ax + By + Cz + D < 0$ を満たす領域は描画しない。すなわち、切断平面の上面を表示しないので、コアの内部構造が露出し、切断されているように見える。この不等式は、切断角度 θ を用いて 1 式で表される。

$$\text{sgn}(\cos \theta)((\tan \theta)x + (-1)y + (0)z + (0.5 - \tan \theta)) < 0 \quad (1)$$

B. パラメータと結果の送受信

レンダーラは、視点移動、画像の品質、画像保存先に関する情報を元に描画を開始する。それまでは、起動時に指定されたポート番号をソケットに割り当ててレンダーラ制御モジュールからの接続を待ち受ける。また、結果の送信時にはレンダーラ起動時に指定された送信ポートに基づいてレンダーラ制御モジュールとソケット通信を行う。送信内容は、現在の視点位置ベクトル、視点の上向き方向ベクトル、CT 値ヒストグラム、1 ピクセルあたりの長さ (mm) と、描画について処理が正常に終了したか否かを示す OK 或いは NG の文字列である。ヒストグラムはボリュームデータの圧縮時に計算され、1 ピクセルあたりの長さは視点の決定時に計算される。

VI. 関連研究

DICOM ファイルをサポートし、高品質なボリュームレンダリングを提供する一般アプリケーションとして Osirix が知られている [4]。Osirix はローカルのデスクトップ環境上で動作するアプリケーションである。大容量の DICOM 形式のファイルをローカル端末上の Osirix で操作する際には、ストレージの十分な空き容量と、ボリュームレンダリングを可能とする高いスペックのハードウェアが要求される。端末のスペックが不足している場合、Osirix による DICOM ファイルの処理は大変遅くなる。こうした問題を解消するために、リモートサーバを用いた DICOM ファイルの提供・閲覧に関する研究が行われている。大容量の DICOM ファイルをリモートサーバに保存する方法として、ネットワークに接続されたストレージを利用する研究がある [5]。しかし、このサービスはあくまで DICOM ファイルを提供するだけのサービスであり、ファイルを直接操作する環境 (ソフトウェアや表示用機材) は持たない。

リモートサーバでボリュームレンダリングを行う方法として、リモートのレンダリングサーバを利用したボリュームデータの閲覧に関する研究がある [6]。この研究ではリモートサーバで視覚化した 3 次元モデルを、MPEG ビデオスト

リーミングを通してクライアントのモバイル端末に表示している。通信に必要となるデータが大きいため、ネットワークの帯域やレイテンシ等のネットワークの品質に大きく左右されてしまう欠点がある。また、ストリーミングはネットワークを多く利用するため、大量の同時アクセスがあった際に安定したサービス提供が難しくなる可能性がある。

本研究では、リモートサーバに DICOM ファイルを保存し、DICOM ファイルの 3 次元モデルをレンダリングした 2 次元画像としてクライアントに送信する。2 次元画像データはストリーミングや 3 次元モデルに比べて軽量であるため、ネットワークの品質に左右することなく閲覧を行うことができる。更にクライアントは、レンダリングされた 2 次元画像を閲覧するために高スペックの端末を利用する必要はない。

VII. おわりに

掘削コア試料の X-CT スキャンデータを用いた分析においてユーザの負担を軽減するために、3 次元処理サーバの構築とクライアントアプリケーションの開発を行った。本サービスにより、高性能な端末や高速な通信回線を必要とすることがなくなるうえ、掘削コア試料の閲覧に適したインタフェースにより、インタラクティブなコア画像の閲覧が Web ブラウザやタブレット端末といった手軽なツールによって、インターネットを介して行うことが可能になる。

今後は、研究者を含む様々なユーザに利用を促し、ユー

ザビリティに関するコメントのフィードバックを収集する。そのうえで、更なるインタフェースの改良を行う。また、本システムは DICOM フォーマットのデータを扱っているため、医用画像にも適用可能であるので、そのインタフェースも新規に検討していきたい。

ACKNOWLEDGMENT

本研究の一部は電気通信普及財団の研究助成により支援されています。This research project is partially supported by The Telecommunications Advancement Foundation.

REFERENCES

- [1] Drilling ship Chikyu returns deepest seabed core samples yet: Nature News Blog, <http://blogs.nature.com/news/2012/09/drilling-ship-chikyu-returns-deepest-seabed-core-samples-yet.html>
- [2] Virtual Core Library, <http://www.kochi-core.jp/VCL/>
- [3] Gdcm Home Page, <http://www.creatis.insa-lyon.fr/software/public/Gdcm/>
- [4] A. Rosset, L. Spadola, and O. Ratib, "Osirix: an open-source software for navigating in multidimensional dicom images.," *Journal of Digital Imaging*, vol. 17, no. 3, pp. 205-216, 2004.
- [5] M. Yakami, K. Ishizu, T. Kubo, T. Okada, and K. Togashi, "Development and Evaluation of a Low-Cost and High-Capacity DICOM Image Data Storage System for Research," *Journal of Digital Imaging*, Springer, vol. 24, no.2, pp. 190-195, April 2011.
- [6] F. Lamberti and A. Sanna. A streaming-based solution for remote visualization of 3D graphics on mobile devices. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):247.260, 2007.