

複数ディスプレイ環境における バーチャル伝統工芸システムの提案

千葉 豪 柴田義孝

岩手県立大学大学院 ソフトウェア情報学研究科

近年、バーチャルリアリティ (VR) の分野では高い没入感を得るために専用グラフィックスエンジンを用いた描画システムや多面体の出力装置を用いた VR 環境が構築されてきた。しかし、これらのハードウェアは非常に高価であり、またネットワークを介した相互運用も困難である。さらにシステム構築が容易でないといった問題があり、これらの装置を利用者が簡単に導入し利用できるものではない。一方で、ブロードバンドネットワークサービスの普及により、IP ネットワーク上で DV や HDV のような高精細映像の配信や、高性能ビデオカードと複数ディスプレイや複数プロジェクタを組み合わせたレンダリングクラスタの構築がソフトウェアレベルで可能となり、高精細で大画面でかつ高臨場感を持つバーチャルリアリティ環境が安価に実現できるようになった。

そこで、本稿では安価な複数のディスプレイを用いたバーチャル伝統工芸システムの提案を行う。本システムでは、出力装置としてタイルディスプレイを用いた 3 次元伝統工芸システムを構築しギガビットネットワークを利用し、複数の利用者によりバーチャル協調作業環境を実現する。

Proposal of Virtual Traditional Crafting System on Multi-displays Environment

Go Chiba Yoshitaka Shibata

Graduate School of Software and Information Science
Iwate Prefectural University

So far, most of virtual reality systems have used dedicated graphics engines and polyhedral display equipment to realize high immersive VR environment. However, since those hardware devices are so expensive, ordinal user cannot easily introduce for his purpose. Those systems also hard to interoperate with multiple user over network.

On the other hand, as advent of broadband network and high performance PC with high definition video card and monitor, more cost effective VR system can be realized by combining multiple number of high performance PCs and display monitors, so called tiled display.

In this paper, we propose a traditional virtual crafting presentation system using the tiled display system. The prototype system is constructed and the simple traditional virtual crafting presentation system is implemented using 4 PCs and LED monitor connected over gigabit-network.

1 はじめに

近年、バーチャルリアリティ (VR) の分野では高い没入感を得るために専用グラフィックスエンジンを用いた描画システムや多面体の出力装置を用いた VR 環境が構築されてきた。しかし、これらのハードウェア

は非常に高価であり、またネットワークを利用して相互運用が困難である。さらにシステム構築が容易でないといった問題があり、これらの装置を利用者が簡単に導入し利用できるものではない。

一方で、ブロードバンドネットワークサービスの普及により IP ネットワーク上で DV や HDV のような

高精細映像の配信や、高性能ビデオカードを搭載した複数の PC と複数ディスプレイやプロジェクタを組み合わせたレンダリングクラスタの構築がソフトウェアレベルで可能となり、高精細で大画面でかつ高臨場感を持つバーチャルリアリティ環境が実現できるようになった [1] [2].

また、筆者らはこれまで先行研究として Windows PC 版および CAVE 版の日本の伝統工芸品を例としたバーチャル伝統工芸システムを構築してきた [4] [5].

PC 版においては感性を考慮したバーチャル伝統工芸システムの構築を行い、利用者にとって容易な感性検索による建具検索や空間検索を実現した。また、利用者が自由に空間構築を行うことができ、閉閉音や環境音を追加した現実感のあるプレゼンテーションシステムとなっている。

CAVE 版では、高臨場感を伴ったバーチャル伝統工芸システムを構築し、さらにリアリティを向上させるためにグラフィックスエンジンとして Onyx3200 を用い、表示装置として CAVE システムを導入し高い没入感を伴った現実感のあるシステムを実現した。しかしながら CAVE は高価格であり、表示用の CAVE ライブラリのライセンスも非常に高価で、伝統工芸品を扱う一般利用者には簡単に導入することは困難である。

そこで、本研究では安価な複数の PC と複数のディスプレイからなるタイルドディスプレイを用いた伝統工芸システムの提案を行う。本システムでは CAVE、Onyx3200 のような専用のハードウェアを利用せず、出力装置としてタイルドディスプレイを用いた 3 次元伝統工芸システムを構築しギガビットネットワークを利用して、複数の利用者によりバーチャル協調作業環境を実現できるシステムの提案を行う。

2 バーチャル伝統工芸システム [4] [5]

バーチャル伝統工芸システムは、VR 空間内に襖、欄間等の伝統工芸オブジェクトを配置し、その中を自由に移動やオブジェクトの操作ができるシステムであり、図 1 のようにクライアント・エージェント、知識エージェント、データベースの三階層からなるアーキテクチャで構成されている。

従来の研究では、VRML2.0 + JAVA によるインターフェースの提供をした PC 版と、OpenGL + CAVE システムを利用した CAVE 版のシステムが構築されている。

クライアントエージェントは利用者からの要求に基づくプレゼンテーションを提供するために、ユーザインターフェース、コントロールマネージャ、メニューマネージャ、オブジェクトマネージャにより構成される。ユーザインターフェースは利用者からのプレゼンテーション要求の受付とプレゼンテーション空間の提供を行う。コントロールマネージャは操作機器の検知

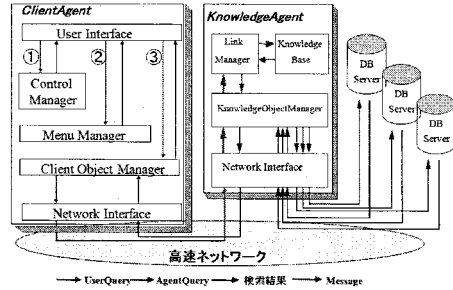


Figure 1: 三層アーキテクチャ

及び制御、ユーザアクションの認識を行う。メニューマネージャは、操作の任意選択、オブジェクトの操作及び選択等を行う。オブジェクトマネージャはプレゼンテーション要求の生成と検索結果の管理を行う。

知識エージェントは感性に基づく検索を提供するために、知識ベース、リンクマネージャ、オブジェクトマネージャにより構成される。知識ベースには感性語による検索要求から特徴量による検索要求にクエリ変換を行うための感性語とプレゼンテーションの関連性が格納される。リンクマネージャはこの知識ベースを利用したクエリ変換を行う。

オブジェクトマネージャは分散するマルチメディアデータベースへの検索要求の発行と、検索して得られた結果の統合に加えて、重複する検索結果のフィルタリングも行う。マルチメディアデータベースにはプレゼンテーション空間内に配置されるオブジェクトと色彩、形状、質感を示す特徴量が格納されている。

2.1 PC 版バーチャル伝統工芸システム



Figure 2: PC 版伝統工芸システム

PC 版は Internet を介した Web ベースのシステムとなっており、実装環境として Windows 上にて Java

+ VRML, Java + Java3D により構築されている。また、利用者にとって容易な感性検索による建具検索や空間検索が可能となっており、利用者が自由に空間を構築することができるインタラクティブなプレゼンテーションを行える。また、閉閉音や環境音を追加した現実感のあるシステムとなっている。図 2 に示すのが PC 版伝統工芸システムの動作画面となっており、ユーザは“落ち着いた”, “重厚な”などの感性語の入力をし、それらの感性語をキーワードとした感性検索と、それによって構築されたプレゼンテーション空間が表示される。また、システム構成は図 3 のようになっており、複数拠点に存在する DB サーバから 3 次元建具データを検索・取得し、個々の利用者のブラウザ上で仮想空間のプレゼンテーションを実現する。

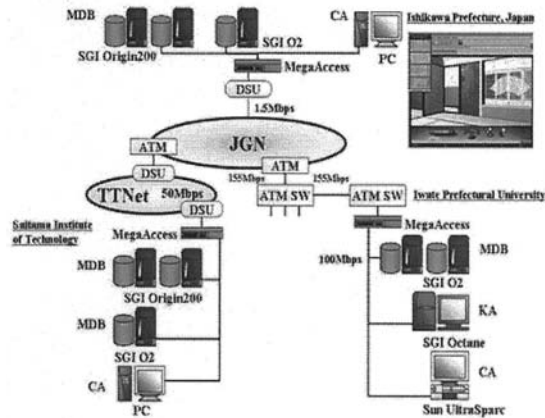


Figure 3: PC 版システム構成

2.2 CAVE 版バーチャル伝統工芸システム



Figure 4: CAVE 版伝統工芸システム

CAVE 版では PC 版をさらにリアリティを向上させるために CAVE システムを導入し、高い没入感を

伴った現実感のあるシステムを構築している。図 4 は CAVE 版伝統工芸システムの動作している様子を示しており、ユーザは入力装置として WAND を利用しており、出力装置である CAVE への描画には OpenGL + Performer を用いている。これらの周辺機器を用いることで更なる操作性や臨場感の向上をさせることができ、部屋空間、庭空間の追加による更なるプレゼンテーション空間の拡張が可能になった。システム構成は図 5 に示すとおりであり、高速ネットワーク上に接続された複数拠点を相互接続し前述した PC 版、CAVE 版の伝統工芸システム間でデータベースや知識エージェントの共有を可能とする環境が提案されている。

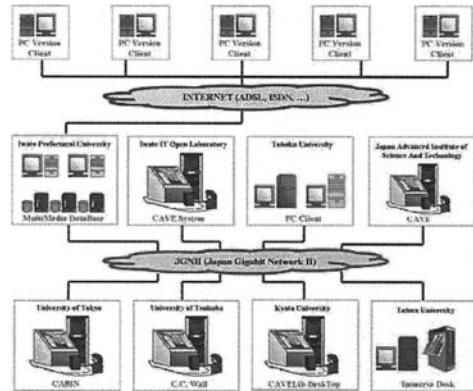


Figure 5: CAVE 版システム構成

3 タイルドディスプレイ版システム

本研究で提案するタイルドディスプレイ型システムの構成は図 6 で示すようにサーバ・クライアント型のシステムになっており、クライアントで動作するアプリケーションで発生するピクセルデータをレンダリングクラスタに向けてストリーム形式で送信する。クライアント-クラスタ間ではデータの送信の有無、状態変化等をメッセージにより制御し、メッセージサーバがそのメッセージの管理を行う。マルチメディア DB には伝統工芸オブジェクトのデータが格納されており、仮想空間内の構築時や伝統工芸品の交換等のインタラクション発生時に DB ヘクエリを発行し、その結果をクライアントに返す。

4 システムアーキテクチャ

本システムのアーキテクチャを図 7 に示す。入力装置としてマウスやキーボード、出力装置としてタイル

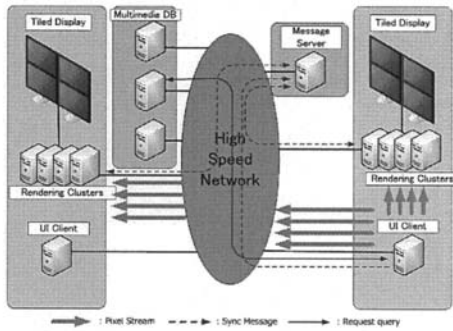


Figure 6: システム構成図

ドディスプレイを想定している。また各レイヤーの機能は下記ようになる。

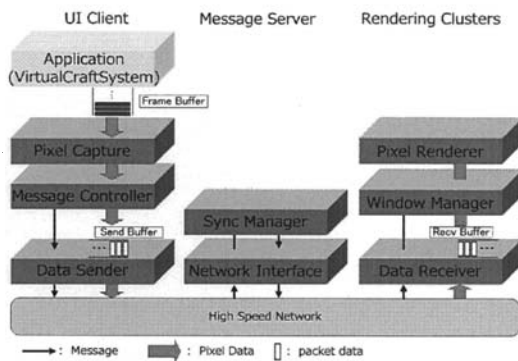


Figure 7: アーキテクチャ

4.1 UI Client

UI Client ではアプリケーションとして利用者へバーチャル伝統工芸空間を提供すると同時に、Pixel Capture モジュールがピクセルデータをグラフィックスカードのフレームバッファから取得、圧縮処理の後に Send Buffer へとデータが格納される。その後 Data Sender が Send Buffer 内のデータをレンダリングクラスタへピクセルストリームとして送信を行う。また、Message Controller モジュールが送受信の同期のためのメッセージを生成し、Data Sender モジュールを介して Message Server へ同期信号を送信する。

4.2 Message Server

Message Server では UI Client, Rendering Clusters から受け取った同期信号をもとにタイルドディスプレイ

への描画処理の同期を計り、その後 Rendering Clusters へ描画用の信号を発信する。これにより各タイル間の同期をとり、スムーズな描画処理が行われる。

4.3 Rendering Cluster

Rendering Clusters では UI Client から送信されたピクセルストリームを受けた後それらのデータを Recv Buffer へと格納する。その後 Window Manager モジュールが Message Server からの同期信号を受け取ると Recv Buffer 内のデータを取得し展開した後イメージを担当するタイルに適合するようリサイズなどの処理を加え、タイルへの描画処理を行う。

4.4 データフロー

システムのデータフローは図 8 のようになっており、UI Client は同期メッセージを Message Server へ送信後各クラスタへピクセルストリームの送信を開始する。各クラスタはそれぞれピクセルデータを受け取ったら描画の準備ができたことを Message Server へと通知する。Message Server では全てのクラスタからの通知を受け取ったら、各クラスタに更新メッセージを送信し、描画処理を開始するよう命じる。また、それと同時に UI Client へ次のデータを送るよう通知をし、UI Client は次のフレームのピクセルデータをストリーミングを開始する。

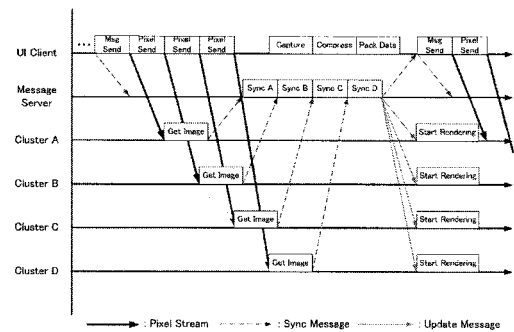


Figure 8: データフロー

5 タイルドディスプレイ

これまで高い没入感を得るために CAVE のような大型の多面体ディスプレイを利用したシステムが多数提案されてきた [5]。しかし、このような大型 CAVE ディスプレイは価格が非常に高くまた CAVE ライブラリも高価であり、一般利用者は導入が困難であり、またシステム構築が容易でないといった問題点があった。

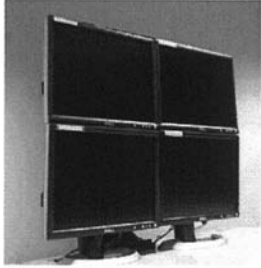


Figure 9: タイルディスプレイ

そこで本研究では出力装置として通常市販されている PC 用ディスプレイを組み合わせたタイルディスプレイを用いる。これは一般的な LCD, DLP 等のディスプレイを複数配置し、仮想的に高解像度ディスプレイ環境を実現するもので、プロトタイプシステムでは 1600x1200 の解像度の LCD を 4 台、9 台組み合わせることにより、3200x2400, 4800x3600 といった高解像度の映像出力を実現できる。各タイルにはレンダリングを行うための複数 PC によるクラスタを構成し、クライアントから受け取ったピクセルデータをタイルに対応した計算機がレンダリング処理を行う。

これにより大型ディスプレイのような専用のハードウェア用いることなく安価な PC によるシステム構成で高臨場感のある表示が可能となり、表示される映像も大画面でかつ、高精細で表示できるので、実物大の伝統工芸空間が再現できるというメリットがあり、高精細でより現実に近いサイズバーチャル伝統工芸空間が実現できる。

5.1 ピクセルデータの取得

UI Client 上でのピクセルデータの取得には図 10 に示すとおり OpenGL の機能である `glReadPixels` 関数によって行い、毎フレームをフロントバッファから RGB にて各ピクセル値を取得する。まず、`glReadBuffer` では読み込む先のフレームバッファの種類を指定する。その後、`glPixelStorei` にてバッファの読み取り方を 1 byte ずつ読み込むように指定し、`glReadPixels` によって読み取る領域の左下の座標 (x,y)、読み取る領域の幅と高さ (width, height)、取得する色情報の形式 (GL_RGB)、読み取ったデータを保存する配列の型 (GL_UNSIGNED_BYTE) を指定してフレームバッファからピクセルデータを取得する。

5.2 データサイズ

ストリームとして扱うピクセルデータは 4800x3600 の解像度の場合

$$4800(\text{width}) * 3600(\text{height}) * 3(\text{RGB}) = 51,840,000 \text{bytes}$$

```
glReadBuffer(GL_FRONT)
glPixelStorei(GL_UNPACK_ALIGNMENT, 1)
pixeldata = glReadPixels(x,
    y,
    width,
    height,
    GL_RGB,
    GL_UNSIGNED_BYTE)
    .
    .
    .
```

Figure 10: バッファ取得関数

となり、図 11 に示すようなピクセルデータを Frame Buffer から取得後、UI Client 上で `zlib[3]` を用い圧縮され UI Client 上の Send Buffer へと格納される。その後、Data Sender モジュールが TCP 経由で各クラスタへとストリーミングを開始する。各クラスタでは Data Receiver モジュールが受け取ったデータを Recv Buffer へと格納し、Window Manager モジュールが展開処理やフレームバッファへの読み込みを行う。その後 Message Server からの更新メッセージを受け取ると Pixel Render へ働きかけ描画処理を行う。

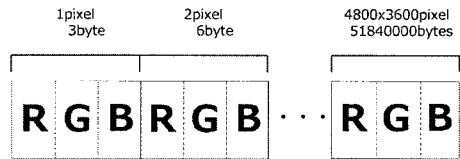


Figure 11: データサイズ

6 プロトタイプ

プロトタイプシステムとして現時点において UI Client, Rendering Clusters の 2 つのコンポーネントを構築し実際に 4 面のタイルディスプレイへの描画を行った。

システムの各マシンの構成は、伝統工芸システムにおいて部屋空間、庭空間等からなるプレゼンテーション空間を提供し、Walkthrough, Flythrough などの基本的な動作を、タイルディスプレイへの出力のためにバッファからのピクセルデータの取得、分割、配信といった機能を実装した。また、各システムの仕様は以下の通りである。

1.) UI Client (DELL Precision470)
 - CPU Intel(R) Xeon 3.4GHz x 2
 - Memory 4GB RAM

- OS Linux kernel2.6.15

2.) Rendering Clusters

- CPU Intel(R) Pentium(R)4 3.73GHz
- Memory 2GB RAM
- Video card nVidia Quadro FX 1400
- OS Linux kernel2.6.19



Figure 12: タイルドディスプレイへの出力結果

また、今後提供する機能として伝統工芸システムにおいては空間の提供、基本動作機能の提供、オブジェクトとのインタラクションといったことを想定しており、空間の提供では大きく”部屋空間”、”庭空間”などの空間が提供される。部屋空間では障子、筆筒などの伝統工芸オブジェクトが部屋空間上に配置され3次元仮想空間として利用者へ提供される。基本動作機能の提供ではユーザが”Track ball”、”Walk through”、”Fly through”、”Flight stick”の4つの移動方法から自由に選択し、選択された動作によって空間内の移動を行うものである。オブジェクトとのインタラクションとしては複数ユーザでの伝統工芸オブジェクトの交換や、配置の変更を可能とした。

分散レンダリングシステムとしてはピクセルデータの取得、データ転送機能、描画処理といった機能がある。ピクセルデータの取得ではグラフィックスカードのフレームバッファから各ピクセルのRGB値を取得し、計算機のメモリへと格納する。データ転送機能では取得されたデータを圧縮した後、画像サイズ、ディスプレイの構成情報等を付加し各クラスターへと送信を行う。描画処理では各レンダリングクラスターが受信したデータを展開し、OpenGLの描画機能を利用してタイルへの描画が可能となった。

7 評価

本システムのプロトタイプ完成後は”フレームレート”、”利用帯域”に関して性能評価を行う。フレームレートはタイルドディスプレイにおいて描画を行うタ

イル数を変化させた際にフレームレートがどのように変化をするか評価をする。また利用帯域ではUI Client, Message Server, Rendering Clusters の3地点間での利用帯域を測定することにより、これら各項目に関して、実際にどのようなボトルネックが存在するかを特定する。そして、今後タイルの数を増加させた場合の表示可能フレームレートや問題点を明らかにする。

8 まとめ

本研究ではタイルドディスプレイ手法を用いた伝統工芸プレゼンテーションシステムを提案した。これは専用のグラフィックスハードウェアを用いず、安価なディスプレイを複数台利用することで容易に高解像度環境を実現し、その高解像度環境において伝統工芸プレゼンテーションシステムを動作させる。これにより、高精細な伝統工芸品オブジェクトの表現が容易にできることが期待される。

今後は更にプロトタイプシステムの構築を進め、描画を行うタイル数の変化させた場合のフレームレートの変化、CAVEシステムを利用した場合とタイルドディスプレイを利用した場合の没入感、臨場感の違いを評価していく。

また、現時点でのプロトタイプがユニキャスト通信を利用している、ピクセルのキャプチャや圧縮をUI Client が全て担当しているためにUI Client の負荷が大きいといった問題が存在している。そこで、マルチキャストを利用することで利用帯域を抑えられることが可能か、UI Client での負荷をどのようにして軽減するかなどに関しても今後検討していきたいと考えている。

References

- [1] Bruno Raffin, Luciano Soares. PC Clusters for Virtual Reality. *IEEE Virtual Reality Conference*, 2006.
- [2] Byungil Jeong, Luc Renambot, Rajvikram Singh, Andrew Johnson, Jason Leigh. High-Performance Scalable Graphics Architecture for High-Resolution Displays. *Tech Paper*, 2005.
- [3] Jean-loup Gailly, Mark Adler. zlib: A Massively Spiffy Yet Delicately Unobtrusive Compression Library. <http://www.zlib.net/>.
- [4] 杉田薫, 宮川明大, 柴田義孝. JGNを利用したVRデジタル伝統工芸システム. *情報処理学会論文誌, Vol.43, No.2, 2002.*
- [5] 石田智行, 宮川明大, 柴田義孝. 超高速ネットワークをベースとした没入型環境システムにおける共有法. *情報処理学会第68回全国大会 5T-11*, 2006.