

マルチモーダルプレゼンテーション記述言語 MPML の 3次元 VRML 空間への拡張

岡崎 直観[†] Santi Saeyor[†] 土肥 浩[†] 石塚 満[†]

An Extension of Multimodal Presentation Markup Language MPML
to a 3D VRML Space

Naoaki OKAZAKI[†], Santi SAEYOR[†], Hiroshi DOHI[†], and Mitsuru ISHIZUKA[†]

あらまし WWW と連携した効果的な情報発信を行う新形態マルチモーダルメディアとして、我々は擬人化キャラクタエージェントを用いるマルチモーダルプレゼンテーションについて研究を行っており、そのマークアップ型記述言語として MPML(Multimodal Presentation Markup Language) を開発している。今回、MPML の特徴を受け継ぎながら、3次元 VRML 空間でのマルチモーダルプレゼンテーション記述を可能にする MPML-VR(Multimodal Presentation Markup Language for VRML) の設計と処理系の開発を行い、VRML プラットフォーム上で動作する擬人化エージェントシステム MPML-VR Agent の実装も行った。3次元 VRML 空間を使用することで、仮想展示場や仮想博物館などの環境でのプレゼンテーション、製品などの3次元物体を提示してのプレゼンテーションが可能となり、視聴者は任意の位置に移動して任意の方向からプレゼンテーションコンテンツを視聴できる。本稿は、マークアップ言語 MPML-VR について記すとともに、VRML 向けに新たに開発した MPML-VR Agent、プレゼンテーション観賞システム MPML-VR Viewer について報告する。

キーワード マルチモーダル、プレゼンテーション、キャラクタエージェント、VRML、マークアップ言語

1. ま え が き

新しいメディアやインタフェースとして人間の認知能力に適合するマルチモーダル化が進んでいるが、我々はそのような新形態情報コンテンツとして擬人化キャラクタエージェントを用いるマルチモーダルプレゼンテーションが有望と考え、その研究開発を進めている。マルチモーダルと言う場合、音声は重要な要素であるが、円滑な音声入力となるとまだ困難な点が多い。しかし、プレゼンテーションにおいては音声出力が大部分のモードであり、音声入力はわずかであるので実用性が高いと言える。

最近のプレゼンテーションはプレゼンタがプレゼンテーションツール (PowerPoint など) を用いるのが主要な形式になってきているが、これは音声による説明、動作や表情、テキストや図表による情報伝達が人間の認知能力によく適合していることから、広く普及

したと言える。しかし、プレゼンタである人間がその場所に居なければならないという大きな制約がある。

我々のマルチモーダルプレゼンテーションは、この人間のプレゼンタの役割を擬人化エージェントに代行させ、時、場所を問わず、プレゼンテーションの視聴を可能にすることを狙いとしている。最近では、クリエータ等の専門的作成者が労力をかければ、コンピュータゲームに見られるような魅力あるマルチモーダル情報コンテンツを作成できる段階になってきているが、多くの人々が HTML を使用して Web コンテンツを自ら記述、発信できるようにしたことにより WWW が大発展したのに習い、我々はマルチモーダル・プレゼンテーションコンテンツも一般の人々が容易に記述できるようにすることが重要であると考えている。そのような目的で、これまでに XML 準拠でマークアップ型の擬人化エージェントを用いるマルチモーダルプレゼンテーション記述言語 MPML(Multimodal Presentation Markup Language) を設計、開発してきた [1] ~ [6]。

一方、VRML(Virtual Reality Modeling Lan-

[†] 東京大学情報理工学系研究科，東京都
School of Information Science and Technology, University of
Tokyo, Hongo 7-3-1, Bunkyo-ku, Tokyo, 113-8656 Japan

guage) [7] に見られるように, WWW は仮想 3 次元空間への拡大が図られている. プレゼンテーションもこれまでの平面的な提示でなく VRML 空間で行うようにすると, 奥行きのある仮想展示場や仮想博物館などのプレゼンテーション, 製品などの 3 次元物体を提示してのプレゼンテーションが可能となる. また視聴者は, 任意の位置に移動して任意の方向からのプレゼンテーションコンテンツを視聴できるというメリットもある.

そこで我々は MPML を拡張し, VRML 空間でのプレゼンテーション記述言語 MPML-VR(MPML for VRML) の設計と処理系の開発を行った. Web 上の仮想 3 次元空間でのアバタ (仮身) キャラクタがガイドをするような VRML コンテンツが作成されるようになっているが, このような仮想空間でアバタによる動きのあるコンテンツの作成は容易ではない段階にある. ゆえに, 一般の人々がプレゼンテーション用途を中心としたコンテンツを容易に記述することを可能にする MPML-VR の有効性は高いと考えている.

本論文では, MPML-VR について記すと共に, VRML 空間でプレゼンテーションを行うエージェントとして新たに開発した MPML-VR Agent, MPML-VR の鑑賞ツールとして開発した MPML-VR Viewer について報告する.

2. マルチモーダルプレゼンテーション記述言語 MPML

2.1 MPML 開発の背景

キャラクタエージェントを使用するマルチモーダルプレゼンテーション・システムとして, 文献 [8]~[17] に示されるような研究が行われている. このようなシステムを構築するには, 動作するキャラクタシステム, 音声認識/合成, 図表等のプレゼンテーションマテリアルなど多くの要素技術を必要とし, それらの連携も図らなければならず大変であった.

最近では Microsoft Agent [18] のように汎用的に使用できる音声機能付きキャラクタシステムも登場しており, 構築の環境は改善されつつある.

しかし, この種のキャラクタシステムを使用するには, キャラクタの動作を低レベルで指示するプログラムを書かなければならず, またそれらはキャラクタシステム毎に異なり共通性がなく, 他への転用はできなかった. 今後, キャラクタエージェントによるマルチモーダル情報コンテンツの作成環境を整えるた

めには, 各種のエージェントシステムに共通に適用できる記述言語, そして多くの人々が容易に使用して魅力的なコンテンツを作成でき, 中位レベルで動作を指示する記述言語が必要である. 以上より, 我々はマルチモーダルプレゼンテーション用途に特化した記述言語 MPML(Multimodal Presentation Markup Language) を開発した.

2.2 MPML の特徴

MPML は, 最初に開発された MPML1.0 [1], [5] をもとに, 複数エージェントに対応した MPML2.0a [4] と感情表現機能が追加された MPML2.0e [2], [3] の 2 つのバージョンが利用可能となっている. これらの言語の詳細についてはそれぞれの論文, もしくは MPML のホームページ [5] を参照していただくことにして, ここではすべての MPML に共通する特徴を紹介する.

システム非依存 MPML はブラウザやエージェントシステム等に依存しない中位レベルで動作を記述する記述言語である. プレゼンテーションのキャラクタが存在し, MPML 記述からこの動作制御プログラムへのコンバータが用意されていれば, MPML プレゼンテーション・コンテンツを異なるキャラクタシステムによっても視聴できる.

キャラクタ制御機能 MPML は豊富なキャラクタ制御要素を搭載しており, キャラクタエージェントの発話内容, 動作, 位置, 感情などをコントロールすることができる.

容易な記述性 MPML は Web 上の標準として普及しつつある XML(Extensible Markup Language) の規格に準拠しており, HTML に似たようなタグ付けを行うことによって, 誰でも簡単に MPML コンテンツを記述することができる.

メディア同期 MPML は SMIL(Synchronized Multimedia Integration Language) [19] に準拠したメディア同期のための機能を有している.

メタデータ記述 プレゼンテーションの内容とともに, コンテンツの作者, 発表日時などの著作権情報を添付することができる.

図 1 は MPML2.0e を利用したプレゼンテーションの画面の例を示している.

3. MPML の 3 次元空間への拡張

3.1 仮想 3 次元空間を用いたプレゼンテーション
現在の MPML システムでは図 1 のように, HTML

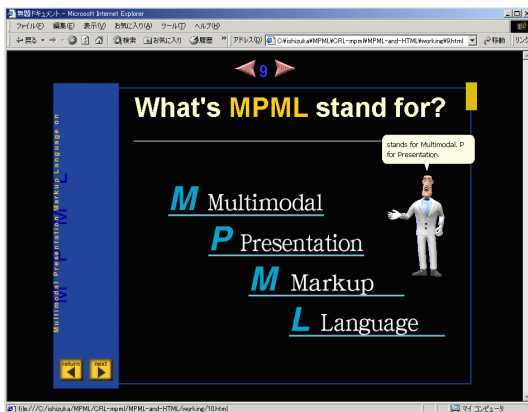


図 1 MPML2.0e の実行画面の例
Fig. 1 Screenshot of an MPML2.0e Presentation

で記述されたプレゼンテーション画面の上に 2 次元の擬人化エージェントを配置している。この「パネル画面（資料）と説明する人」というプレゼンテーションは、学会発表や講義に留まらず一般的に広く行われているものであり、テキストや図等の題材を用いた概念の伝達に適したスタイルであると考えられる。ここでは、このようなプレゼンテーションを 2 次元のプレゼンテーションと呼ぶ。

ところで、不動産屋で写真や図面を使って説明された物件でも実際に現地に行くと思っただけより狭いことや、頭の中で使おうと考えていた家具が内装に合わないことはよくある話である。この例をプレゼンテーションに照らして考えると、不動産屋での図面や写真を用いた説明は住宅という 3 次元物体を 2 次元の媒体へ投影した 2 次元のプレゼンテーションであり、実際に住宅を見に行くのは実物を用いた 3 次元のプレゼンテーションである。この例では 3 次元のプレゼンテーションが聞き手の理解に関して大きなウェイトを占めていたが、これは視点を連続的に変化させたことにより視聴者が空間把握のための情報をより多く獲得し、写真などの 2 次元の物体では伝わりづらかった距離感を体感したことによるものであると解釈できる。

このような 3 次元のプレゼンテーションが威力を発揮すると思われる事例は多岐にわたり、新製品のカメラや自動車の説明、街の道案内、立体構造を説明する教材など適用範囲は広いと思われる。

一方で、3 次元プレゼンテーションはさほど普及していないというのが現状である。その原因の一つとして、3 次元コンテンツ作成の難しさが挙げられる。ま

して、キャラクターエージェントを用いたマルチモーダルなプレゼンテーションを 3 次元環境で実現するとなれば、専門のクリエイターでないと作成できないくらいハードルの高いものとなってしまう、一般のユーザーが気軽に作成できるとは言い難い。

3.2 MPML-VR

そこで、今までの 2 次元のプレゼンテーションと相補的な役割を果たす 3 次元のプレゼンテーションにも MPML の容易な記述性を生かし、プレゼンテーション空間に VRML2.0 (Virtual Reality Modeling Language Version 2.0) [7] を採用し、MPML の適用範囲を 3 次元空間でのプレゼンテーションへと拡張した記述言語 **MPML-VR** (Multimodal Presentation Markup Language for VRML) の設計・開発を行った。VRML2.0 を採用したのは、Web コンテンツとしての適合性に注目したからであるが、1996 年のリリースから日数も経過しているため、オーサリングツール、コンテンツともある程度実績を積んでいることも見逃せない。

MPML-VR では説明に用いる背景 VRML が静的であっても、動作する 3 次元キャラクターエージェントを含む MPML-VR スクリプトを記述することで、動きのある 3 次元プレゼンテーションを作成できる。さらに、MPML-VR は背景 VRML とは別途に作成した VRML ファイルをプレゼンテーション空間に配置する機能も有している。このようにプレゼンテーションによく用いる背景世界や題材などをテンプレートとすることで、素材となる VRML コンテンツの再利用性を高めるとともに、背景世界の作成を支援している。

ところで、3 次元空間とキャラクターエージェントの組み合わせ方としては、背景世界だけ 3 次元化してキャラクターエージェントに Microsoft Agent のような 2 次元のキャラクターを用いる方法と、背景世界とキャラクターエージェントの両方を 3 次元で作成する方法の 2 通りが考えられる。前者の場合は背景世界とキャラクターエージェントの座標系が独立になってしまうため、視聴者が背景世界の視点を動かしてしまうと、キャラクターエージェントが背景世界で表示されている物体を指差したり、指定された場所へ移動するというような背景世界とのインタラクションが困難となり、仮想世界を用いたプレゼンテーションの魅力が半減する。よって、本研究では後者の方法を採用して、キャラクターエージェントをプレゼンテーションの仮想環境内に埋め込むことにした。

3.3 MPML-VR の設計方針

MPML はプレゼンテーションの処理系に依存しない記述言語であり、このスタンスは MPML-VR においても引き継がれるべきものがある。本来なら MPML の仕様を拡張することなく 3 次元空間を適用できるのが望ましいが、3 次元空間での MPML の実証は初めての試みであり、現在の MPML の仕様では 2 次元プレゼンテーションに特化した点^(注1)がいくつか残されている。

そのような経緯で MPML-VR を新たに開発したが、MPML を 3 次元空間へ拡張するにはこれまでの MPML の言語仕様や特徴をできるだけ継承することに重点を置き、過去に MPML コンテンツを作成したことがある人であれば、無理なく MPML-VR も利用できるように配慮した。また、言語設計に際しては MPML-VR も 2 次元プレゼンテーションと相反することの無いように十分注意し、別途研究が続けられている 2 次元版 MPML と近い将来統合されることを視野に入れている。また、MPML-VR はプラットフォームとして VRML を選択したが、同じような理由で VRML に依存した言語仕様にならないように注意した。このことは将来普及すると思われるポスト VRML への移植性の確保にも貢献する。

具体的に心がけた点として、MPML-VR の仕様では XML 要素・属性名に 3 次元や VRML 特有の用語・概念を使わず、すべてのプレゼンテーション環境において共通に使用することのできる抽象的な概念・呼称を採用する^(注2)、エージェントシステムや背景世界の処理系に依存しそうな記述箇所はエージェントプロフィール(後述)やオブジェクトプロフィール(後述)として MPML-VR コンテンツの外に記述することにする^(注3) ことなどである。

3.4 MPML-VR 言語仕様の概要

MPML-VR の XML 要素の構造木は図 2 のとおりである。網掛けされたボックスが MPML-VR で新たに追加された XML 要素(タグ)を示している。詳しく

(注1): 例えば MPML では位置指定に 2 次元 (x,y) 座標を用いている

(注2): 例えばエージェントの位置指定には座標を用いず場所名を用いると定義する。その上で VRML 処理系における場所名はノード名と解釈するが、HTML 処理系における場所名は HTML オブジェクトの属性 ID と解釈させることもできる。

(注3): MPML-VR コンテンツとエージェントプロフィール、オブジェクトプロフィールとの間のプロトコルは統一し、MPML-VR の処理系とプロフィールとの間のプロトコルは MPML-VR の処理系に任せることにした。これは OS のシステムコールのようなメカニズムを MPML-VR に導入したと考えると分かりやすい。

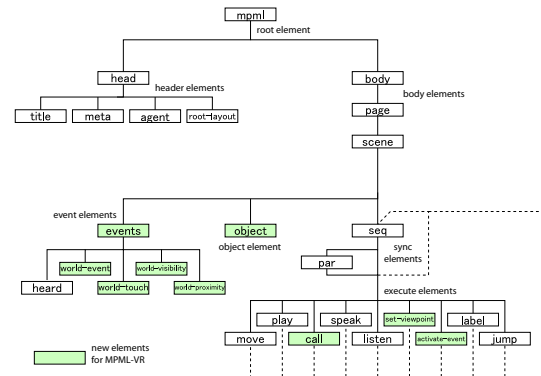


図 2 MPML-VR の XML 要素ツリー
Fig.2 Element Tree of MPML-VR

い MPML-VR の言語仕様については MPML-VR のホームページ [20] を参照していただくことにする。

今までの MPML では、プレゼンテーション画面上で擬人化エージェントによる説明が行われており、背景となるパネル画面が切り替わることによってプレゼンテーションが進行していた。しかし、3 次元プレゼンテーションにおいて背景の仮想空間を頻りに切り替えるということは、いわゆる「カット」を多用することを意味し、リアリティを損ねたり、視聴者の理解を寸断してしまう恐れがある。前にも述べたように、3 次元プレゼンテーションでは背景の仮想空間をできるだけ切り替えずに、空間的位置や時間を変化させた方が効果的なプレゼンテーションとなる。

以上のことから MPML-VR では、MPML2.0 において追加された scene 要素に「状態」という意味合いを含ませ、シーン間の状態遷移によってもプレゼンテーションの実行制御を記述できるようにした。

イベント要素は、音声入力とか背景空間へのクリックなど、シーン内において発生する事象に対してどこへ状態遷移すればよいかをあらかじめ記述しておくものである。以下の例は「video」という音声を認識したときに"scnVideo"という ID 値を持つシーンへジャンプし、"panel"という場所 (VRML ノード) を視聴者がマウスでクリックしたときに"scnRun"というシーンへジャンプするという遷移を記述している。

```
<events>
  <heard id="evVideo"
    agent="Andy" voice="video" jump-to="scnVideo"
  />
  <world-touch id="evTouch"
    type="Click" location="panel" jump-to="scnRun"
  />
</events>
```

他にも、背景世界からのイベントを検出する world-

event 要素, 背景世界の物体の可視性を検出する world-visibility 要素, 視聴者の視点がある領域に入ったことを検出する world-proximity 要素などが用意されている.

実行スクリプトの側からは, それぞれのイベントを受け付けるかどうかを以下の例のように設定できるので, 特定の期間のみ音声入力やクリックを受け付けるというような記述も行える.

```
<seq>
  <speak agent="Andy">Please touch the panel.</speak>
  <activate-event event="evTouch" flag="True" />
</seq>
```

object 要素はプレゼンテーション仮想空間の中に, 外部オブジェクトと呼ばれる物体を配置するための要素である. 2次元版の MPML では, ボタンやテキストボックスなどのオブジェクトを背景画面内に配置する機能があるが, MPML-VR でも同様の機能を用意した. さらに, MPML-VR では外部オブジェクトのメソッドを呼び出すことにより, 配置した外部オブジェクトの状態を実行時に変更することができる. このような処理の記述を行おうとする場合, 背景世界を制御する処理系に依存してしまうので, MPML-VR ではオブジェクトプロフィールという概念を導入して処理系に固有の記述と MPML-VR の記述を分離した. オブジェクトプロフィールは取り込むオブジェクトの情報を XML 形式で以下のように記述するもので, 例えば VRML では, 取り込みを行う物体の外観を記述した VRML ファイルの URL, オブジェクトが保持するメソッドの名前や型情報, バインドさせるフィールド名などの固有情報を記述することになる.

```
<object type="vtml"
  url="television.wrl" node="Television">
  <call>
    <method id="url" type="MFString-url"
      node="Movie1" field="url"
    />
    <method id="start" type="SFTime"
      node="Script_Television" field="startAt0"
      sync-node="Script_Television"
      sync-field="triggerStop"
    />
  </call>
</object>
```

上のように定義された外部オブジェクト (テレビパネル: television.xml) をプレゼンテーション空間の中に取り込み, call 要素を使用してテレビパネルオブジェクトのメソッドを呼び出す. 指定した URL の動画を読み込み・再生を行うには MPML-VR で以下のような記述を行うだけである.

```
<scene id="scnMovie">
  <object
    id="TV" url="television.xml" location="posTable"
```

```
</scene>
```

このように, MPML-VR では処理系に依存するような記述を排除し, できるだけプレゼンテーションにおいて抽象的な記述を採用しているため, MPML-VR を他の処理系へ移植した場合でも仕様を修正する必要がない.

set-viewpoint 要素は仮想空間における視聴者のカメラ位置を変更するもので, カメラ位置を移動させた場所 (VRML ノード名) を以下のように指定する.

```
<set-viewpoint location="camCenter" transition="ON"/>
```

move 要素は, エージェントの位置を変更するものであるが, 座標値による指定から場所名による移動先の指定に変更されるとともに, 以下に例示するように, 移動手段 (歩く, 走る等) の設定ができるようになってきている.

```
<move agent="Andy" how="run" location="posCenter" />
```

play 要素の動作アニメーション名や発話時のピッチなど, 特定のエージェントシステムや音声入出力システムに依存しそうなパラメータは, 後ほど説明するエージェントプロフィールを介することにより, 異なる実装システム間の互換性を確保している.

付録には, MPML-VR スクリプト記述のサンプルを示す.

3.5 関連研究

DFKI (独) では, 3次元自動車モデルを複数の Microsoft Agents が説明するシステムを作成している [10]. このシステムの説明対象物は3次元であるが, プレゼンテーションエージェントは2次元の画面上であり, ユーザも3次元空間を移動することはできない.

仮想環境のみを用いている研究の中でも Jacob [17] は VRML を用いており, Humanoid Animation で記述されたキャラクターエージェントが仮想環境内で自然言語等のインタラクションにより, ハノイのデモなど特定の仕事をユーザーに教えてくれる.

このように仮想環境のみでプレゼンテーションを行うシステムは, 対象とするプレゼンテーションの範囲を限定しているものが多い. また, いずれのシステムも記述言語として設計されていないため, システムを構築した人以外がコンテンツを記述することは想定されていないが, Java などの言語によるプログラミング

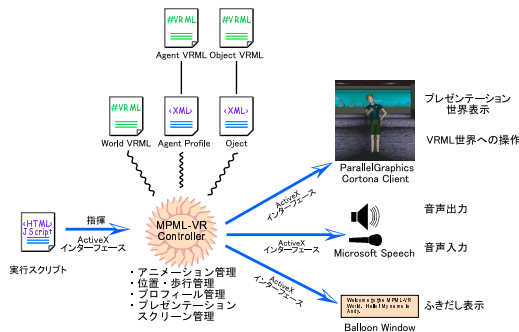


図 3 MPML-VR Agent のブロック図
Fig. 3 The Diagram of MPML-VR Agent

が必要となる。

Virtual Human Presenter [12] は Tex ライクなプレゼンテーションスクリプトを書くことにより, PaT-Nets と呼ばれる制御モデルを導入したエージェントによるプレゼンテーションを実現しているが, プレゼンテーションに用いることのできるジェスチャの数は少なく, ユーザーインタラクションの機能も有していない。

4. VRML によるキャラクタエージェント

4.1 MPML-VR Agent

MPML-VR でプレゼンテーション空間に VRML を適用するに当たり, MPML-VR プレゼンテーション向けのキャラクタエージェントシステムが必要となり, MPML-VR Agent を開発した。この構成について以下に記す。

VRML を用いてシステムを構成する場合, 音声認識・合成など VRML だけでは記述できない処理を, 別の言語で記述されたモジュールで補う必要がある。VRML には他のモジュールとインタラクションを行うために, EAI や Java3D などの手段が用意されているが, ここでは ActiveX を用いて Java によらない Web アプリケーションの作成を行った。

MPML-VR Agent は図 3 のように, 音声入出力・VRML ブラウザのライブラリと, それらを統括してキャラクタエージェントの管理を行う MPML-VR Controller という構成になっている。

まず下層側(図 3 右側)から説明すると, エージェントの音声関連の処理については, Microsoft 社の提供している Microsoft Speech SDK 4.0 [21] を利用している。キャラクタエージェントやプレゼンテーション世界の表示を担当する VRML ブラウザの制御には,

ParallelGraphics 社製 Cortona SDK [22] を用いた。この 2 つの SDK(Software Development Kit) はいずれも, ActiveX と呼ばれるコンポーネントオブジェクトモデルを採用しており, 他のソフトウェアからの統一的な制御が可能になっている。

これらのコンポーネントの総合的な管理を行うために, MPML-VR Controller というライブラリを用意した。MPML-VR Controller の実装には, Microsoft Visual Basic 6.0, Microsoft Visual C++ 6.0 を用いた。以下に, MPML-VR Controller の構成と機能について箇条書きでまとめる。

ActiveX コンポーネント MPML-VR Controller

は, MPML-VR プレゼンテーションの表示・制御を行うための高水準なインタフェースを提供する ActiveX コンポーネントである。ActiveX コンポーネントは, HTML の中に埋め込んで JavaScript や VBScript 等のスクリプト言語から制御することができるので, Web ブラウザのプラグインという形態で MPML-VR Controller を利用することができる。

エージェントのふきだし表示 MPML-VR ではキャラクタエージェントの発話による情報伝達が可能であるが, 合成音声出力は聞き取りにくい場合もある。このような点も考慮して, 音声出力と同期してふきだしを表示させるようにした。

エージェントの動作アニメーション管理 キャラクタエージェントによるプレゼンテーションでは, 主な情報伝達を音声チャンネルが担うことになるが, 身振り・手振りといったノンバーバルコミュニケーションもバックグラウンド・チャンネルとして視聴者の理解を助ける。MPML-VR Agent では, キャラクタエージェントのプレゼンテーション用の動作を VRML データ中にあらかじめ組み込んでおき, それらの動作アニメーションを制御する VRML フィールドにアクセスすることによって, エージェントの動作の開始・終了要求, 動作完了通知を行うようにした(図 4)。これは, これまでの MPML によるエージェントの動作制御の考え方を踏襲するものであり, コンテンツ作成者による中位レベルの動作記述を可能にする。

エージェントの位置・歩行管理 前もって準備されるエージェントの動作アニメーションに対し, エージェントの移動では簡単な歩行エンジンを搭載し

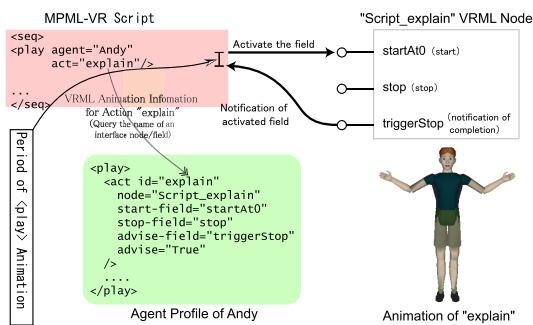


図 4 play 要素によるアニメーション制御の概念図
Fig. 4 How to Animate with 'play' Element

```

<?xml version="1.0" encoding="Shift_JIS"?>
<agent-setting type="vrml">
  <vrml
    url="file://c:/users/okazaki/sotsuron/vr-mpml/samples/agent/andy_low.wrl"
    hair-model="hair_andy_low"
    walkmodel-url="file://c:/users/okazaki/sotsuron/vr-mpml/walkTest/walkModel.wrl"
    normal-anim="restpose"
    on-load-anim="eyes"
  />
  <speech
    engine="ModeName=Mike" pitch="100" speed="200"
  />
  <play>
    <aplaud
      node="script_aplaud" start-field="startAt0" stop-field="stop"
      advise-field="triggerstop" advise="True" sound="applaud.wav"
    />
    <breeze
      node="script_breeze" start-field="startAt0" stop-field="stop"
      advise-field="triggerstop" advise="True"
    />
    ...
    <restpose
      node="script_restpose" start-field="startAt0" stop-field="stop"
      advise-field="triggerstop" advise="false"
    />
  />
  <move>
    <walk
      node="script_walk" start-field="startAt0"
      turntime="0.5" movetime-scale="1.0" time-per-step="1.0"
    />
    <run
      node="script_run" start-field="startAt0"
      turntime="0.5" movetime-scale="0.3" time-per-step="0.5"
    />
  />
</agent-setting>
  
```

図 5 エージェントプロフィール XML の例
Fig. 5 A Sample of an Agent Profile XML

ている。エージェントの現在位置と移動先情報から、歩行時の VRML キーフレーミングアニメーションに必要な Interpolator ノードを動的に生成することで、任意の地点からの移動を可能にしている。

エージェントプロフィール管理 エージェントプロフィールは、エージェントの外観を記述する VRML の URL、声の性別やピッチ、歩行時の諸定数と同時実行する動作アニメーション名、利用できる動作名などの各エージェント固有の設定を、図 5 のようにひとつの XML ファイルにまとめたものである。これはキャラクターエージェントそのものを表すデータベースと捉えることができる。MPML-VR でプレゼンテーションに使うエージェントを指示するにはこのファイル名を指定し、MPML-VR Agent がプロフィールの記述内容を解釈して、エージェントに関するパラメータとして参照するという仕組みである。

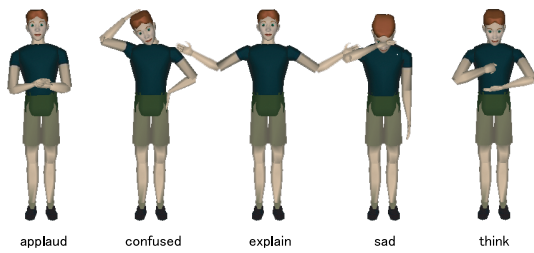


図 6 Andy のアニメーション例
Fig. 6 Sample Animations of Andy-agent

外部オブジェクト取り込み MPML-VR では、プレゼンテーション空間中に出現するエージェント以外の部分は、背景 VRML に記述することになっているが、プレゼンテーションにおいて 3 次元空間の背景は重要でなく 1 つの物体を説明したいだけの場合、このようなメカニズムだけでは具合が悪い。そこで MPML-VR では、背景 VRML のほかに、外部オブジェクトと呼ばれる VRML で記述されたオブジェクトをプレゼンテーション空間内に取り込むためのインターフェースを用意した。外部オブジェクトの外観 VRML ファイルの URL や公開メソッドに関する情報は、エージェントプロフィールと同様に、オブジェクトプロフィールとして XML ファイル形式で記述しておく。テレビ(動画を提示する)や黒板(2次元プレゼンテーション題材を提示する)など、プレゼンテーションでよく用いる物体をテンプレートとしてあらかじめ準備しておくことにより、背景の VRML を効率よく製作できる。

4.2 MPML-VR Agent の例: Andy

今回、MPML-VR Agent 用のキャラクターエージェントの作成に当り、SeamlessSolutions 社のフリーサンプル版 Andy [23] を利用した。Andy は、Humanoid Animation [24] に準拠した VRML によるアバタであり、アニメーションは含まれていなかった。そこで、Humanoid Animation に対応した VRML オーサリングツールを用いて、プレゼンテーションに必要と思われる 23 種の動作を作成した(今後、この種類は充実させることを予定している)。作成したアニメーションの例を図 6 に示す。

5. MPML-VR 鑑賞システム

5.1 MPML-VR Viewer

MPML-VR による 3 次元マルチモーダルプレゼン

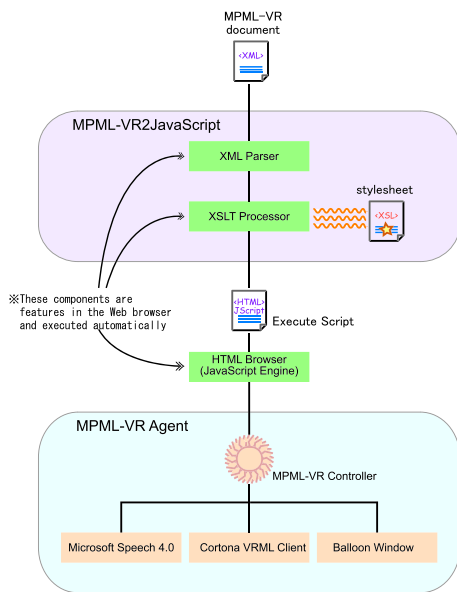


図 7 MPML-VR Viewer の全体構成
Fig. 7 Software Layers in MPML-VR Viewer

テーションを実用化するためには、オーサリングツールやプレイヤーをはじめとした環境整備も必要である。今回は MPML-VR に対応したプレゼンテーション鑑賞ソフトウェア MPML-VR Viewer も併せて開発し、MPML-VR による 3 次元プレゼンテーションの実証を行った。

MPML-VR Viewer は、Web ブラウザのプラグイン形式で動作する MPML-VR プレゼンテーション鑑賞ソフトウェアである。プラグイン形式にしたため、利用者の負担はプレイヤーのインストール等の必要最低限に留まり、HTML コンテンツの閲覧と同等な操作でプレゼンテーションを鑑賞することができる。

MPML-VR Viewer は図 7 に示すように、MPML-VR2JavaScript と MPML-VR Agent の 2 つのレイヤから構成されている。以下では各レイヤについて説明する。

5.2 MPML-VR2JavaScript

XML に対応したブラウザで MPML-VR のファイルを開くと、ブラウザは XML 文書として認識しパース (parse) 処理を行う。XML 文書としての正当性の検証を行い、MPML-VR ドキュメントはスタイルシート適用のため XSLT(eXtensible Stylesheet Language Transform) プロセッサへ送られる。XSLT プロセッサはスタイルシートに記述された変換ルールに従い、

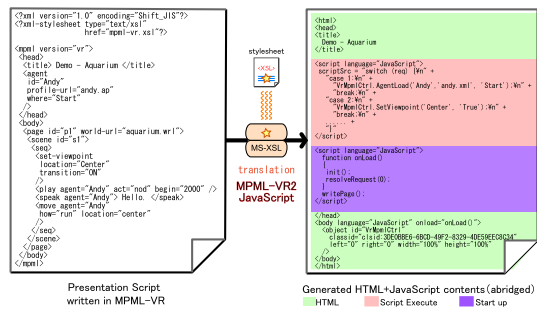


図 8 MPML-VR2JavaScript の動作例
Fig. 8 Work of MPML-VR2JavaScript

MPML-VR コンテンツを Web ブラウザで実行可能な HTML と JavaScript によるコンテンツに変換する (図 8)。

以上の動作をまとめると、MPML-VR ドキュメントを Web ブラウザで直接開くことのできる HTML+JavaScript のコンテンツに翻訳することであり、MPML-VR Viewer 中におけるこのレイヤを MPML-VR2JavaScript と名付けた。XML に対応したブラウザの機能により MPML-VR2JavaScript の開発はすなわち、スタイルシートを XSL を用いて記述することに置き換わる。

5.3 MPML-VR Agent

MPML-VR2JavaScript による MPML-VR プレゼンテーションの翻訳作業が完了すると、ブラウザにより HTML+JavaScript コンテンツの実行が開始される。このコンテンツの実行が MPML-VR プレゼンテーションとなる。

JavaScript は元々、動きのないホームページにインタラクティブ性を持たせるために開発されたスクリプト言語であり、いろいろと制約が多い。また、プレゼンテーションを実行する JavaScript は XSLT で動的に生成させなければいけないため、「JavaScript プログラムを生成するための XSL スタイルシートを作る」という特殊な開発スタイルをとらなければならない。このようなことも、MPML-VR プレゼンテーションをコントロールするための高水準なインタフェースを提供する ActiveX コンポーネント MPML-VR Controller を別途作成した理由である。

JavaScript プログラムから MPML-VR Controller を呼び出すことにより、MPML-VR2JavaScript の生成する JavaScript はシンプルなものとなった。

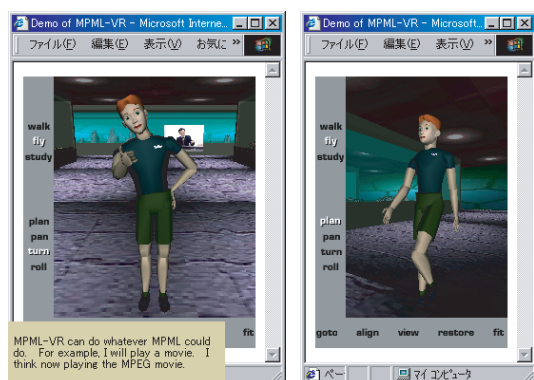


図 9 MPML-VR プレゼンテーション例

Fig. 9 Screenshots of an MPML-VR Presentation

5.4 MPML-VR のプレゼンテーション例

図 9 は、サンプルとして作成した MPML-VR コンテンツを Internet Explorer で開き、MPML-VR Viewer を通してプレゼンテーションを視聴しているところである。図 9 左は Andy が “good” のアニメーションをしているところであるが、その背後で外部オブジェクトであるテレビパネルを取り込んで、MPEG 動画を再生しているところにも注目してほしい。図 9 右は Andy が歩いているところである。

6. 考 察

定量的評価は比較対象のシステムが少ないことから、定性的な考察を中心に記しておく。

MPML-VR コンテンツを MPML-VR2JavaScript によって HTML と JavaScript に変換した際のファイルサイズの比較を図 10 に示す。ファイルサイズの小さい分だけ記述しやすくネットワークへの負荷も小さいと言えるのに加えて、エージェントシステム依存の部分の処理を極力排除し、seq, par 要素によるメディア同期関連の煩雑な記述、イベント処理の記述などもすべて自動で行われるなど、プレゼンテーションスクリプトの作者の負担を大幅に軽減している点も重要である。しかしながら、MPML-VR2JavaScript は XSLT によって実装されているため、MPML-VR のコンテンツ記述に文法的な誤りがあった場合に受け取るエラーメッセージは XSLT プロセッサが吐き出す難解なものになってしまう。さらに、XSLT 規格は要素属性の型チェック機能が貧弱であるため、型エラーであっても変換後の JavaScript を実行するまで捕捉できない。以上の問題点を克服するには、残念ながら XSLT

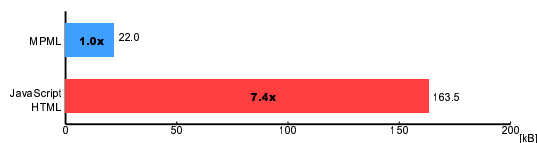


図 10 MPML-VR と MPML-VR2JavaScript で生成されるコンテンツのサイズ比較

Fig. 10 Size Difference between MPML-VR and the low-level description converted by MPML-VR2JavaScript from MPML-VR

プロセッサの使用やめるしか方法がない。幸い、XML には DOM や SAX といったような標準的な処理方法が用意されているので、これらを利用してコンバータを自前で記述し、エラーチェックを強化するのも今後の一つの方法である。

MPML-VR Agent には簡単な歩行エンジンが搭載されているため、エージェントはプレゼンテーション空間内の任意の地点へ移動可能である。VRML オーサリングツールを用いなくても、擬人化エージェントの任意の地点への移動アニメーションを表現できるのは大きな特徴である。しかし、エージェントの動作（身振り）については、現状では play 要素でのみ記述するため、「 をせよ」といった程度の表現力しか持っていない。今後「 を指す」とか「 の方へ顔を向ける」といったように、より動作指示をし易い記述をサポートする必要があると考えている。

3次元キャラクタエージェントの作成にはキャラクタエージェントの外観の作成、動作の追加という2つの工程を踏まなければならない。外観の作成には既存の VRML ツールを利用でき、最近では VRML アバタを着せ替え人形のように作成できるツールも存在する。動作の追加にも既存の VRML ツールを用いることができるが、こちらは、逆運動学やモーションキャプチャを用いて作成する方が、より自然な動作を作成できると思われる。今回は VRML のオーサリングツールを用い、キーフレームを編集することでキャラクタエージェントの動作を作成したが、MPML-VR Agent 用のキャラクタエージェントの作成環境を整備し、既存の研究 [25] などのさまざまな手法を転用できると好ましい。

エージェントの動作に関する記述は、今のところ play 要素で行われるが、複数の動作が par 要素を用いて並列実行された場合は、現在は保証外となっている。通常はニュートラルポジションに戻ってから次の動作へ移るので問題は生じないが、ある動作の実行中

に別の動作を割り込ませた場合、そのトランジションを生成する機能が無いため、アニメーションに不連続点が生じてしまう。このような問題に対処するには、Humanoid Animation の骨格情報を生かし、人体の関節の位置情報を直接制御してアニメーションを作成する方法が効果的であると思われる。ただし、VRML2.0 の仕様ではキーフレーミングによるアニメーションしかサポートされないため、VRML ブラウザを外部から制御し、アニメーションデータを逐次生成するか、キーフレーミング以外のアニメーションの枠組みを VRML ブラウザに実装しておく必要がある。

7. ま と め

マルチモーダルプレゼンテーション記述言語 MPML を 3次元 VRML 空間上へ拡張した MPML-VR を設計し、その処理系を開発・実装した。これによりキャラクターエージェントによる 3次元空間でのマルチモーダルプレゼンテーションを、多くの人々が比較的容易に記述、作成できることを可能にした。

また、VRML 空間上でのキャラクターエージェントシステム MPML-VR Agent も併せて開発し、MPML-VR からプレゼンテーション用途に使用できるようにした。今後改善すべき点も残されているが、WWW で流通する 3次元仮想空間での有望な新形態マルチモーダル情報コンテンツの基盤技術になるものと考えている。

直近の機能向上としては以下の項目を予定している。

- ・ 豊かな顔表情や任意の地点への指示など、プレゼンテーションエージェントとしての機能の向上
- ・ エージェントのアニメーションの動的生成による自然な動きの作成
- ・ テンプレート等を使ったワールド VRML の簡単な作成方法

謝辞 本研究は日本学術振興会未来開拓研究「マルチモーダル擬人化インタフェースとその感性基盤機能」によるものである。

文 献

- [1] 筒井貴之, 石塚満, “キャラクターエージェント制御機能を有するマルチモーダルプレゼンテーション記述言語 MPML,” 情報処理学会論文誌, vol.41, no.4, pp.1124-1133, 2000.
- [2] Y.Zong, H.Dohi, H.Prendinger and M.Ishizuka, “Emotion Expression Function in Multimodal Presentation,” *Advances in Multimodal Interface - ICM2000* (Proc. 3rd Int'l Conf. on Multimodal Interface) (Y.Tan, Y.Shi and W.Gao (eds.)), Lecture Notes in Computer Science 1948, pp.57-64, Springer, 2000.
- [3] Y.Zong, H.Dohi, and M.Ishizuka, “Multimodal Presentation Markup Language MPML with Emotion Expression Function Attached,” Proc. 2000 Int'l Symp. on Multimedia Software Eng. (IEEE Computer Sec.), pp.359-365, Taipei, 2000.
- [4] 何斌達, 宗元, 土肥浩, 石塚満, “キャラクターエージェントを用いたプレゼンテーション記述言語 MPML の拡張,” 第 60 回情報処大全予稿, No.5ZE-8, Mar. 2000.
- [5] MPML Homepage, <http://www.miv.t.u-tokyo.ac.jp/MPML/mpml.html>
- [6] OASIS-MPML Web page, <http://www.oasis-open.org/cover/mpml.html>
- [7] VRML Homepage, <http://www.web3d.org/vrml/vrml.html>
- [8] J.Cassell, S.Prevoost, J.Sullivan and E.Churchill, *Embodied Conversational Agents*, The MIT Press, 2000.
- [9] E.Andre, T.Rist and J.Muller, “Web Persona: A Lifelike Presentation Agent for the World-Wide Web,” *Knowledge-Based Systems*, vol.11, no.1, pp.26-36, 1998.
- [10] E.Andre and T.Rist, “Presenting through Performing: On the Use of Multiple Lifelike Characters in Knowledge-based Presentation Systems,” *Knowledge-Based Systems*, vol.14, no.1-2, pp.3-13, 2001.
- [11] B.Hayes-Roth and P.Doyle, “Animate Characters,” *Autonomous Agent and Multi-Agent Systems*, vol.1, no.2, pp.195-230, 1998.
- [12] T.Noma, L.W.Zhao and N.I.Badler, “Design of a Visual Human Presenter,” *IEEE Computer Graphics & Applications Magazine*, vol.20, no.4, pp.79-85, 2000.
- [13] J.Rickel and W.L.Johnson, “Animated Agents for Procedural Training in Virtual Reality, Perception, Cognition, and Motor Control,” *Applied Artificial Intelligence*, vol.13, pp.343-382, 1999.
- [14] J.C.Lester et. al., “Deictic Blievability, Coordinating Gesture, Locomotions and Speech in Lifelike Pedagogical Agents,” *Applied Artificial Intelligence*, vol.13, no.4-5, pp.383-414, 1999.
- [15] W.L.Johnson, J.W.Rickel and J.C.Lester, “Animated Pedagogical Agents, Face-to-Face Interaction in Interactive Learning Environments,” *Int'l Jour. of Artificial Intelligence in Education*, vol.11, pp.47-78, 2000.
- [16] J.Cassell, “More than Just Another Pretty Face, Embodied Conversational Interface Agents,” *Commun. ACM*, vol.43, no.4, pp.70-78, 2000.
- [17] M.Evers, A.Nijholt, “Jacob - An Animated Instruction Agent in Virtual Reality,” In *Advances in Multimodal Interfaces - ICM2000*, Proc. Third International Conference on Multimodal Interfaces, Beijing, China, T.Tan, Y.shi and W.Gao(Eds.), Springer-Verlag, Berlin, pp.526-533, 2000.
- [18] Microsoft Agent Web Site,

- [19] <http://msdn.microsoft.com/workshop/imedia/agent/>
SMIL Homepage,
<http://www.w3c.org/TR/REC-smil/>
- [20] MPML-VR Homepage,
<http://www.miv.t.u-tokyo.ac.jp/mpmlvr/>
- [21] Microsoft Speech 2000 Web Site,
<http://www.microsoft.com/speech/>
- [22] ParallelGraphics Cortona SDK Homepage,
<http://www.parallelgraphics.com/products/sdk>
- [23] SeamlessSolutions Humanoid Animation Homepage,
http://www.seamless-solutions.com/html/animation/humanoid_animation.htm
- [24] Humanoid Animation Homepage,
<http://www.h-anim.org/>
- [25] 遠藤守, 安田孝美, 横井茂樹, “VRML に基づく人体動作アニメーション作成システム”, 信学技報, vol.99, no.16, MVE99-12, pp.37-44, Apr. 1999.

```

</seq>
</scene>
</page>
</body>
</mpml>
    
```

(平成 x 年 xx 月 xx 日受付)

付 録

1. MPML-VR スクリプトのサンプル

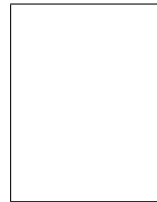
```

<?xml version="1.0" encoding="Shift_JIS"?>
<?xml-stylesheet type="text/xsl" href="mpml-vr.xsl"?>
<mpml version="vr">
  <head>
    <title> Sample Script of MPML-VR </title>
    <agent
      id="Andy" profile-url="agent/andy.xml" location="dad_AgntPosStart"
    />
    <root-layout left="0" right="0" width="100%" height="100%" />
  </head>
  <body>
    <page id="p1" world-url="world/sample01.wrl">
      <!-- Initialization -->
      <scene id="toInit">
        <object
          id="TV" url="parts/television.xml" location="dad_ObjectPosPanel"
        />
        <seq>
          <jump to="toMenu" />
        </seq>
      </scene>

      <!-- VoiceCommandMenu -->
      <scene id="toMenu">
        <events>
          <heard id="evVideo" agent="Andy" voice="video" jump-to="toVideo" />
          <heard id="evMove" agent="Andy" voice="move" jump-to="toMove" />
        </events>
        <seq>
          <speak agent="Andy"> I am waiting for your command. </speak>
          <listen agent="Andy" />
        </seq>
      </scene>

      <!-- Video -->
      <scene id="toVideo">
        <seq>
          <par>
            <play agent="Andy" act="pleased" />
            <speak agent="Andy">I will play a movie. </speak>
          </par>
          <par>
            <play agent="Andy" act="acknowledge" />
            <seq>
              <speak agent="Andy">I think now playing the MPEG movie. </speak>
              <set-viewpoint location="CameraPanel" transition="ON" />
            </seq>
            <seq>
              <call object="TV" method="url" value="movie/ishizuka.mpg" />
              <play agent="Andy" act="good" begin="2000"/>
              <call object="TV" method="start" value="1" begin="2000" />
            </seq>
          </par>
          <set-viewpoint location="CameraStart" transition="ON" />
          <jump to="toMenu" begin="1000" />
        </seq>
      </scene>

      <!-- Move -->
      <scene id="toMove">
        <seq>
          <speak> I am going to walk to the AgentPosCenter node. </speak>
          <set-viewpoint location="CameraWalk" transition="ON" />
          <move agent="Andy" how="walk" location="dad_AgntPosCenter" />
          <set-viewpoint location="CameraCenter" transition="ON" />
          <speak begin="1000"> I can also run, you know. </speak>
          <set-viewpoint location="CameraRun" transition="ON" />
          <move agent="Andy" how="run" location="dad_AgntPosStart" />
          <set-viewpoint location="CameraStart" transition="ON" />
          <jump to="toMenu" />
        </seq>
      </scene>
    </page>
  </body>
</mpml>
    
```



岡崎 直観 (学生員)

2001 年東京大学工学部電子情報工学科卒業。現在同大学大学院情報理工学系研究科修士課程在学中。2002 年より科学技術振興事業団リサーチスタッフ。趣味は音楽の作成から演奏、果ては圧縮まで。現在は文章自動要約の研究を行っている。



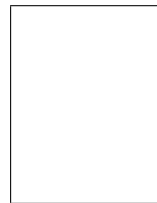
Santi Saeyor

1993 年 King Mongkut's Inst. of Tech. (タイ)・電子卒。1999 年 9 月東京大学工学系研究科電子情報工学博士課程修了。博士(工学)。1999 年より未来開拓研究学振りサーチ・アソシエイト。Web 情報システムの研究を行い、現在は擬人化エージェントによるマルチモーダルシステムの研究を行っている。



土肥 浩

1985 年慶應義塾大学理工学部電気卒業。1987 年同大学院修士課程修了。同年東京大学生産技術研究所勤務。1993 年工学系研究科電子情報工学専攻助手。1999 年より新領域創成科学研究科基盤情報学専攻。研究分野は知的擬人化エージェントインタフェース、画像メディア技術、ネットワーク化知的情報環境。ACM, 情報処理学会各会員。



石塚 満 (正員)

1971 年東京大学工学部電子卒業。1976 年同大学院博士課程修了。工学博士。同年 NTT 入社、横須賀研究所。1978 年東京大学生産技術研究所助教授。1992 年工学部電子情報工学科教授。2001 年より情報理工学研究科電子情報学専攻。研究分野は人工知能、知識処理、マルチモーダル擬人化エージェント、ネットワーク化知的情報環境。IEEE, AAAI, 情報処理学会, 人工知能学会, 映像情報メディア学会, 画像電子学会, 顔学会等の会員。