

IntelligentBoxにおけるコンピュータアニメーション作成環境について

岡田義広 田中譲

北海道大学工学部

060 札幌市北区北13条西8丁目

電子メール:okada,tanaka@huee.hokudai.ac.jp

著者らは、3次元応用ソフトウェアの構築が画面上での直接操作により対話的に行える対話型3次元ソフトウェア構築システム(IntelligentBox)の研究開発を行っている。3次元形状と固有の機能をもつ可視オブジェクト(Box)を画面上での直接操作により組み合わせ形状と機能を合成することにより、3次元応用ソフトウェアの開発が可能となる。IntelligentBoxシステムの適用分野の一つとしてコンピュータアニメーション作成が挙げられる。フルカラー描画された機能部品であるBoxを画面上で合成することで即座にアニメーション作成が可能となる。本稿では、アニメーション作成のためにIntelligentBoxが提供する機能とBoxについて説明し、コンピュータアニメーション作成におけるIntelligentBoxの有用性を述べる。

Computer Animation Creation Environment of IntelligentBox

Yoshihiro Okada and Yuzuru Tanaka

Faculty of Engineering, Hokkaido University

N.13, W.8, Kita-ku, Sapporo, 060 Japan

e-mail:okada,tanaka@huee.hokudai.ac.jp

The authors have been studying a constructive visual 3D software development system called IntelligentBox. The IntelligentBox provides visible objects called Boxes as its primitives which have a unique function and a 3D shape. It also provides the mechanism that enables users to construct 3D graphic applications by combining individually existing Boxes through direct manipulations on a computer screen. One of application fields of the IntelligentBox is computer animation creation. With the IntelligentBox, it is possible to create computer animations rapidly by combining full-color rendered Boxes. This paper describes the functions of the IntelligentBox system and particular Boxes, and clarifies their availability for computer animation creation.

1. はじめに

我々は、固有の機能をもつ3次元表示されたオブジェクトを組合せ機能合成することにより、3次元表示機能をもつ応用システムを構築可能な3次元ソフトウェア構築システムの研究・開発を行っている。すでに、IntelligentBox[1,2]と呼ぶプロトタイプ・システムを開発している

IntelligentBoxの機能合成の機構は、IntelligentPad[3]の機構を採用している。IntelligentPadは、現在、北海道大学で研究・開発を行っているセンセティックメディア・システムである。IntelligentPadにおける紙のイメージをもつ機能オブジェクトであるパッドを3次元形状を持つ機能オブジェクトへと拡張したシステムがIntelligentBoxであるといえる。

図1に示すのは、IntelligentBoxシステムの画面ハードコピーである。すべての3次元オブジェクトがボックスの形状をもつものではないが、IntelligentBoxでは多面体の形状をもつ3次元オブジェクトを総称してボックスと呼ぶ。各ボックスは3次元形状と固有の機能をもつ。このボックスを画面上で互いに親子関係を与え組み合わせることにより、合成された機能をもつボックスを構築できる。IntelligentBoxシステムでは、この構築過程が3次元応用システムの開発過程となる。3次元応用システムを開発する上で有用となる種々の基本部品（特定の機能をもったボックス）を現在までに開発している[4,5,6]。

IntelligentBoxシステムの応用分野として、コンピュータアニメーション作成が挙げられる。従来のIntelligentBoxシステム(以下、旧版IntelligentBoxという)では、ボックスの3次元表示はワイヤフレームイメージのみであり、フルカラー描画された場合の画面上での直接操作は不可能であった。今回、IntelligentBoxシステム(以下、新版IntelligentBoxという)の実装をグラフィックス性能の優れたワークステーション上で行った。これにより、フルカラー描画された

機能部品であるボックスを画面上での直接操作により組み合わせることで形状設計と機能合成を同時に行うことができ、コンピュータアニメーション作成を即座に行うことが可能となった。

従来のコンピュータアニメーション作成用ソフトウェアのアニメーション作成過程は、1)オブジェクトの形状設計、2)モーション定義、3)各モーションに対応したシーンの描画という過程に分けられる。今回実装したIntelligentBoxシステムでは、1)オブジェクトの形状設計機能と3)描画機能はIntelligentBoxシステム自身の機能として提供する。2)モーション定義はボックスの機能として提供する。本稿では、種々の機能部品を用いたモーション定義機能とソフトオブジェクトを扱うための形状変形機能をもつボックスについて述べる。また、これらのボックスを用いてオブジェクトの形状定義を行うための対話型インタフェースについて述べる。さらに、いくつかのアニメーション作成例を挙げて、これらボックスの有用性を示す。

以下では、2節で、まずIntelligentBoxの概要としてIntelligentBoxの機能合成機構を述べる。3節ではコンピュータアニメーション作成のためにIntelligentBoxが提供する機能と機能部品であるボックスについて説明する。4節ではコンピュータアニメーションの実例を示し、IntelligentBoxシステムのコンピュータアニメーション作成における有用性を示す。最後に5節でまとめと今後の課題を述べる。

2. IntelligentBoxの概要

本節では、IntelligentBoxの基本機構について述べる。IntelligentBoxの基本機構は以下の4点にまとめられる。1)MVモデリング、2)親子関係、3)スロット結合によるデータ授受、4)Model共有によるデータ共有。以下では、これらについて簡単に説明を加える。

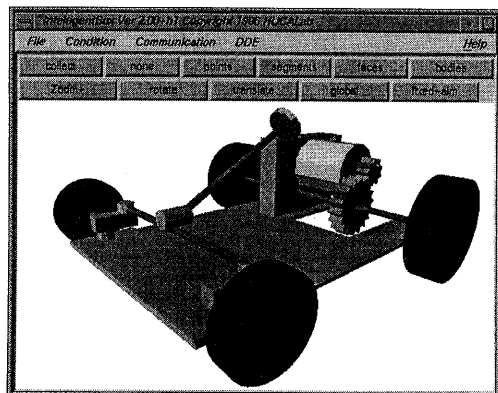
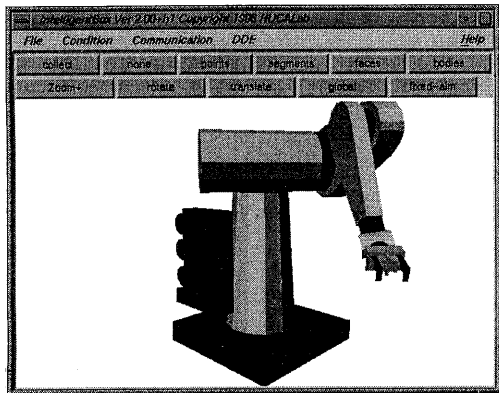


Fig.1 IntelligentBoxシステムの画面ハードコピー

2.1 MVモデリング

旧版IntelligentBoxは、IntelligentPadと同様にMVCモデリング[1,2,3]を採用していた。1つのボックスは、Model、View、Controllerと呼ばれる3つのオブジェクトで構成される。Modelはボックスの内部状態を保持するオブジェクトである。状態情報はスロットと呼ばれるものに保持される。Viewはボックスの画面上での視覚的な形状を表すオブジェクトである。Controllerはパッドに対する使用者からのマウスイベントを受け取り、Viewへ送るためのオブジェクトである。一方、新版IntelligentBoxではMVモデリングを採用している。ボックスに対するユーザ操作はすべてのボックスにおいて共通であるため、複数種のControllerを用意する必要がない。新しいボックスを作成する際のコーディングの冗長性をなくすため、Controllerの機能をViewの機能に含めた。図2に示す方式でメッセージ送信と状態の更新報告を行う。

2.2 親子関係

ボックスを機能合成する際、合成されるボックス間に親子関係が与えられる。IntelligentBoxにおける親子関係は、子ボックスはその親ボックスに從属するという関係である。子ボックスの空間的な位置はその親ボックスの位置に対する

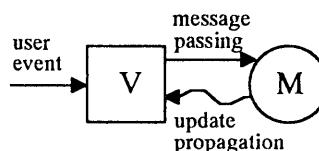


Fig.2 ボックスのMV構造

相対的な座標で表される。親子関係は各ボックスのViewにおける関係として決められる。

2.3 スロット結合によるデータ授受

IntelligentBoxは、IntelligentPadと同様のプロトコルにより、親子関係があるボックス間でデータの授受を行う。

1) メッセージの種類と動作

図3に示すように、set <slotname> <value>、update、gimme <slotname>メッセージがある。set <slotname> <value>は、各ボックスが自分の親ボックスが持つ<slotname>で指定されるスロットに対して値<value>を送るためのメッセージである。gimme <slotname>は、自分の親ボックスが持つ<slotname>で指定されたスロットから値を読み出すためのメッセージである。updateは、各ボックスが自分のすべての子ボックスに対して、自分の内部状態が変化したことを知らせて状態更新を要求するためのメッセージである。

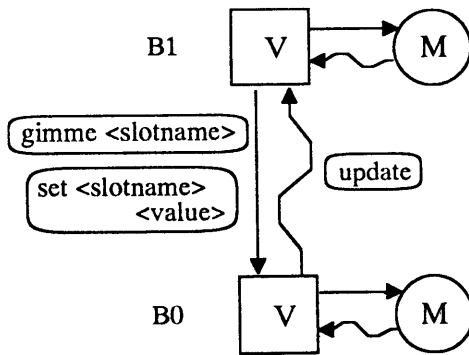


Fig. 3 ボックス間の標準メッセージ

setメッセージにより自分のどのスロットの値を親ボックスのどのスロットに代入するか、およびgimmeメッセージにより自分のどのスロットに親ボックスのどのスロットの値を代入するかは、ユーザのメニュー選択により決められる。これをスロット結合と呼ぶ。

2) メッセージの意味

上記のメッセージの送受信許可は、View部がもつset、gimme、updateフラグによって決められる。setフラグが立っているボックスはその親ボックスに対する入力器として機能する。逆に、gimmeフラグが立っているボックスはその親ボックスに対する出力器として機能する。

2.4 Model共有によるデータ共有

先述したように、それぞれのボックスはModel、Viewの2者で構成される。図4に示すように、ひとつのModel部は複数のViewをもつことができる。この際、複数のViewはModelを共有しているという。これにより、Model部を共通にもつ異なる2つのボックス間でデータの共有が可能となる。

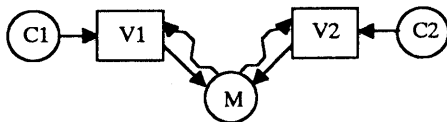


Fig. 4 2つのボックス間のModel共有

3. アニメーション作成機能

アニメーション作成過程は、以下に示すようにモデリング機能、レンダリング機能、モーション定義機能の3つに分けられる。本節では、これら3つの機能を中心にIntelligentBoxが提供するアニメーション作成のための機能について述べる。

3.1 モデリング機能

旧版IntelligentBoxはSymbolics社のLISPマシン上に実装されていた。このマシンにはFlavor言語で書かれたS-Productsというコンピュータアニメーション作成ソフトウェアが装備されている。旧版IntelligentBoxシステムは、このソフトウェアにIntelligentPadがもつ基本機構を加えることにより実現されている。各ボックスの形状設計はこのソフトウェアが提供するモデリング機能を用いることにより可能であった。グラフィックスワークステーション上に今回実装したIntelligentBoxシステムでは、このソフトウェアと同様のモデリング手法を採用している。このソフトウェアが提供する機能の一部を実現している。3次元形状の各構成要素（頂点、稜線、面、立体）を単位として、種々の変形操作が画面上での直接操作により可能である。

3.2 レンダリング機能

すでに述べたように、新版IntelligentBoxシステムは、実時間描画の可能なワークステーション上に実装されている。各ボックスを実時間でフルカラー描画可能である。したがって、画面上での直接操作により各ボックスを組み合わせそれらもつ機能を合成することにより、対応するボックスの動きや形状に制約を与えることができ即座にコンピュータアニメーションが作成できる。

3.3 モーション定義機能

すでに述べたように、オブジェクトの動きに制約を与えるための特定の機能をもつボックス

がいくつか実現されている。以下の図5,6,7に示すのはこれらの例である。

3-3-1 汎用稼動部品の合成による制約

伸縮部品や回転部品等の汎用稼動部品を合成することにより、オブジェクトの動きに制約を与えることができる。図5は、回転部品(RotationBox)と伸縮部品(ExpandBox)を用いて機能合成することにより、シャトルが回転しながら上昇する動きを実現した例である。

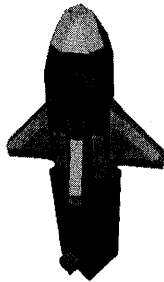


Fig. 5 稼動部品の合成による動きの定義例

3-3-2 軌道部品

特定の軌道上を移動するという動作を実現するために、軌道自身を表すボックス(TrajectoryBox)とその軌道上のみ移動するという機能をもつボックス(TrajectoryMoverBox)を用意している。図6は、与えられた軌道に沿って進んで行くシャトルの例である。

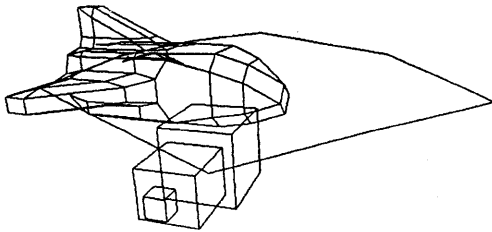


Fig. 6 軌道部品を用いた制約の例

3-3-3 物体表面を移動する制約部品

特定の物体の表面に沿って移動するという動作を実現するために、親ボックスの表面上を移動するという機能をもつボックス(CreepingBox)

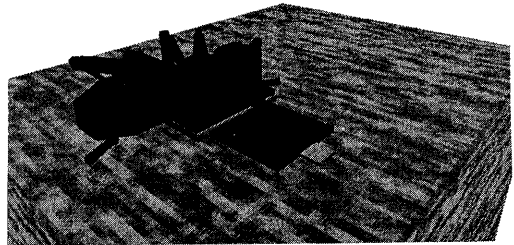


Fig. 7 机の上をアリが這う例

を用意している。図7は、机の上をアリが這っていく例である。

3.4 形状変形機能

すでにFree Form Deformation(FFD)とよばれる立体形状変形の方法[7]が提案されてる。これをFFDControlBoxとよぶボックスの機能として実現している。FFDControlBox自身の形状が制御部品として働くのではない。FFDControlBoxは、別に用意される制御部品の形状情報にしたがって制御対象の形状を変化させる機能をもつ。制御部品は合成されたものでもよく、汎用稼動部品を任意の組み合わせで用いてもよい。これにより種々の動きを定義できる。以下の図8に示すのは、FFDControlBoxを用いてワシの翼の動きを実現した例である。回転部品と直方体の形状をもつ部品を合成することにより、翼の動きを定義している。この合成ボックスをワシの翼の立体形状を制御する部品としてFFDControlBoxと組み合わせることで用いることにより、ワシが翼をうねらせる様子を模倣する。

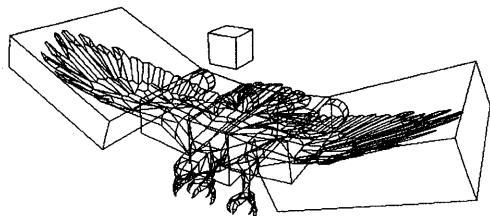


Fig. 8 FFDControlBoxを用いたワシの翼の動きのシミュレーション例

3.5 スケルトンモデル

以下の図9は、3自由度をもつボックス(3DRotationBox)を複数用いて階層的に合成したものである。これは手のスケルトンモデルを表しているといえる。図8の例のように、この合成ボックスを制御部品としてFFDControlBoxと組み合わせて用い、手の立体形状へ適用することで手の動きの模倣が可能となる。

図10は、図9の合成ボックスを手の立体形状に適用した例の画面ハードコピーである。同様の方式で人体のスケルトンモデルを定義し、人の全身を表す立体形状へ適用することで、人体の動きの模倣が可能となる。

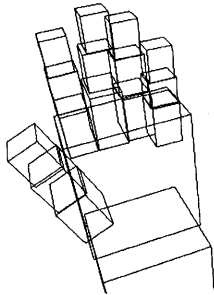


Fig.9 3DRotationBoxを用いた手のスケルトンモデル例

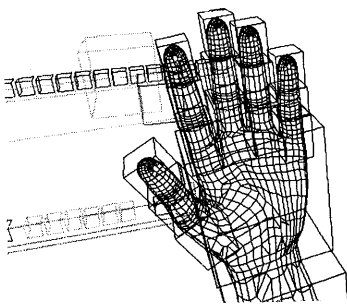


Fig.10 スケルトンモデルを手の立体形状へ適用した例

3.7 リージョンモデル

上述したように、FFDControlBoxを用いた形状変形では、汎用稼動部品の単体あるいはそれらの合成部品を制御部品として用いる。それを

制御対象へ適用することで変形を行う。この方法では、顔の表情に代表される筋肉モデルを基に形状が変化するような例への適用は難しい。物体形状をいくつかの領域に分けて管理し、各領域に対して独立にFFDを適用することで、より精細な変形制御が可能となると思われる。顔の形状を筋肉モデルを基に複数の領域に分け、各領域に対して Rational Free Form Deformation(RFFD)を適用することで顔の筋肉の動きを模倣するという提案[8]がすでにある。そこで、物体形状を領域に分けて管理する機能をもつボックス(MeshBox)と一つの領域に対する変形制御機能をもつボックス(RFFDControlBox)を今回新たに実装した。MeshBoxの領域の定義は、画面上での対話的な操作のみで容易に行える。

次の図11は、MeshBoxとRFFDControlBoxを用いた顔のアニメーションの実例である。

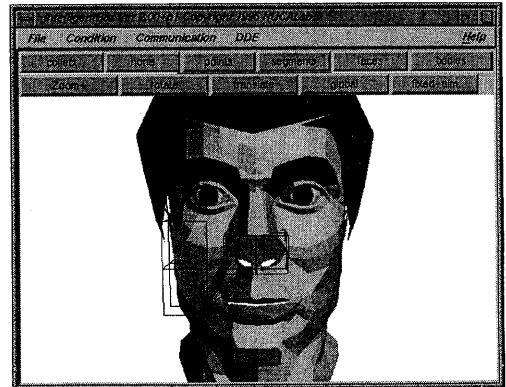


Fig.11 MeshBoxとRFFDControlBoxを用いた顔の表情定義の例

3.8 Shape Editor

文献[8]では、さらに、顔の表情を定義するためのインターフェースとしてExpressionEditorとよぶツールを提案している。複数のパラメータを用意し、これらのパラメータを画面上で操作することにより、対応する領域に対して指定された変形が行われるものである。これを使うこ

とにより顔の表情の定義を対話的な操作で容易に行える。

IntelligentBoxシステムは、ExpressionEditorに対応するものとして、ShapeEditorとよぶツールを用意している。これは複数のパラメータからなるウィンドウを提供する。パラメータを操作することにより、ユーザによって指定された命令がメッセージとして対応するボックスに送られ操作が行われる。このShapeEditorは汎用ツールである。パラメータの数や命令の種類、メッセージが送られるボックスの指定は、定義ファイルを書くことにより行う。したがって、定義ファイルを個々に用意することにより、先の図に示した顔の表情の定義用インタフェースとして、あるいは、手の形状の定義用インタフェースとして用いることができる。次の図12は、手の形状定義用ShapeEditorの画面ハードコピーである。

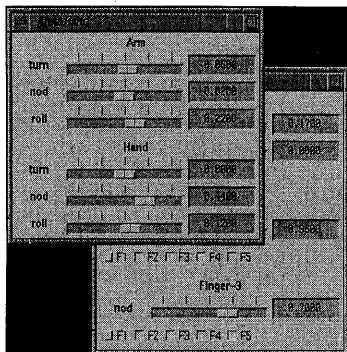


Fig. 12 ShapeEditorの画面ハードコピー

3.9 カメラ機能

アニメーション作成において視線は重要な要素である。IntelligentBoxは視線の管理を行うための機能部品(CameraBox)[6]を用意している。また、このCameraBox自身が他の部品によって制御され移動する際に、特定のオブジェクトに常に視線を向けたい場合がある。このような場合のために、特定の面が指定されたオブジェクトを常に向くという機能をもつ機能部品(AutoFocusBox)を用意している。これらの部品

を他の機能部品と組み合わせて用いることにより、アニメーション作成の際の視線管理が容易に行える。

4. アニメーションの実例

旧版IntelligentBoxシステムでは、次の図13に示すアニメーションがすでに作成されている。これは二人のファイターがお互いに戦っている場面を表したものである。ファイターの動きはすでに開発されているいくつかの機能部品を利用することにより定義している。また、空には鳥が飛んでいる。鳥の進行方向は軌道部品を用いることにより制御している。鳥の翼の動きはFFDControlBoxを用いて定義している。視線の管理をCameraBoxとAutoFocusBoxを用いて行っている。すでに述べたように、旧版IntelligentBoxには実時間描画機能がない。そのために、一シーン毎に数分の時間を費やしてフルカラー描画し、ビデオテープに一フレームづつコマ録りしてアニメーションを作成する必要があった。新版IntelligentBoxでは、実時間でフルカラー描画可能であり、即座にアニメーションを実行することができる。現在、図13に示すアニメーション例を新版IntelligentBoxシステムに移植しているところである。

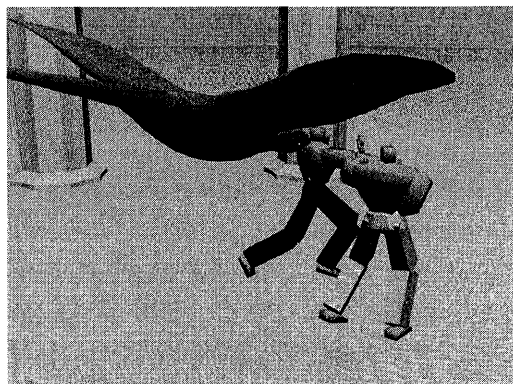


Fig. 13 アニメーションの一例

5. おわりに

3次元表示された機能オブジェクトを画面上での直接操作により組合せ機能合成することにより3次元ソフトウェアの開発が行えるシステムとしてIntelligentBoxの研究開発を行っている。本稿では、グラフィックスワークステーション上に新たに実装したIntelligentBoxシステムの概要とコンピュータアニメーション作成のためにIntelligentBoxが提供する機能と機能オブジェクトであるボックスについて説明した。IntelligentBoxシステムにおいて、すべての機能部品は自由に組み合わせ可能である。アニメーションのための機能をもつ部品を組み合わせることで、ボトムアップにアニメーション作成が行える。いくつかのアニメーションの実例を示すことで、アニメーション作成が容易に行えることを明らかにした。

今後は、種々のアニメーションを実際に作成することにより、IntelligentBoxシステムの有用性を明らかにするとともに問題点を明らかにし、それらの問題点を検討し解決するつもりである。

参考文献

[1] 岡田義広、田中譲：対話型3Dソフトウェア構築システム-IntelligentBox, コンピュータソフトウェア, Vol. 12, No. 4, pp. 84-94, 1995.

[2] Okada, Y. and Tanaka, Y.: A Constructive Visual Software Development System for Interactive 3D Graphic Applications, Proc. of Computer Animation 95, pp. 114-125, Geneva(1995).

[3] 長崎祥、田中譲：シンセティックメディア・システム-IntelligentPad-, ソフトウェア学会論文誌, Vol. 11, No. 1, pp. 36-48, 1993.

[4] 岡田義広、田中譲：機能オブジェクトの3次元化とその機能合成について-IntelligentPadの3次元化-, 情報処理学会研究会資料, Vol. 93, No. 67, pp. 53-60, 1993.

[5] 岡田義広、田中譲：3次元オブジェクトの機能合成システム(IntelligentBox)-幾何情報を扱う部品の提供-, 情報処理学会情報学シンポジウム論文集, pp. 199-208, 1994.

[6] 岡田義広、田中譲：3次元オブジェクト機能合成システムにおける視野管理のための機能部品の導入, 情報処理学会第50回全国大会講演論文集(5), pp. 281-282, 1995.

[7] Sederburg, T. W. : Free-Form Deformation of Solid Geometric Models, ACM SIGGRAPH '86 Proceedings(1986), pp. 151-160.

[8] Kalra, P., Mangili, A., et. al.: Simulation of Facial Muscle Actions Based on Rational Free Form Deformations, EUROGRAPHICS '92, Vol. 11, No. 3, pp. C59-C69, 1992.