

マルチメディア構築ツール「HAT」

矢田光治 太田耕三 岩田憲治 池田太
ハイパーメディア, CSK総合研究所

本稿はマルチメディア・オーサリング・ツール「HAT」の概要を示す。この特徴は、ソフトウェア・プロデューサーにプログラミングの具体的知識を要求しないことである。従って、プロデューサーは、高度にインタラクティブなマン・マシン・インターフェース技術を核とするVR（バーチャル・リアリティ）ソフトウェア「VIP」を、そのイメージのままに自在に実現することができる。「VIP」はさらに、音や音声などもこの要素として備える予定であり、これに伴って「HAT」の完成度は一層高度となる。

MULTIMEDIA AUTHORIZING TOOL "HAT"

Koji Yada , Kozo Ohta , Kenji Iwata and Futoshi Ikeda

HYPER MEDIA Corporation
Yuraku Bldg, 1-7-20 Moto-Akasaka,
Minato-ku, Tokyo 107, Japan

CSK Research Institute
2-5-1 Suwa, Tama-shi,
Tokyo 206, Japan

In this paper, the outline of multimedia authoring tool "HAT" is presented. What is characteristic of this tool is that it requires no concrete knowledge of programming of software producers. This enables them, consequently, to represent freely, as they imagine, "VIP", Virtual Reality software which has for its nucleus most advanced interactive man-machine interface technology. "VIP" will be further enhanced by the elements such as sound and voice, which will make "HAT" still more highly-perfected authoring tool.

I. はじめに

マルチメディアの切り口は多様で、その技術の全てがいわゆるハイテクである。第1はコンピュータ技術、第2に人工知能技術、第3に放送・通信技術、第4に家電技術で、現在および将来のマルチメディアのインフラとなる。

マルチメディアの発展方向については、「より便利に」、「より優しく」、「よりリアルに」と言う提示がある〔1〕。第1はハイパーメディアの方向、これは狭義には、ハイパーテキストのマルチメディア・バージョンである。第2は感性メディアの方向、これはいわゆる概念メディアを扱う領域の人工知能技術の方向である。第3はバーチャル・リアリティ（VR）の方向で、ここで紹介するHATはVRソフト「VIP」のオーサリング・ツールである。

VRを実現する道具は、データ・グローブやHMDが知られる。しかしこれらの道具を身につける煩わしさや、仮想世界の追従性の悪さは、逆にリアリティを妨げることもある。

データ・グローブやHMDを使わない比較的軽いハードでVRソフトを実現してみようとする、音と映像の2つだけでユーザーの視覚と聴覚を錯覚に誘い込み、疑似体験をさせることが目標となる。動作速度についても、ユーザーが疑似的なものと意識しないスピード感を与えることが必要となる。本稿では、このような狙いを持つVRソフト「VIP」、そのオーサリング・ツール「HAT」の概要を紹介する。

II. VIPの概要〔2〕

・ハードウェア構成

VIPの基本的なハードウェア構成は、図1のようである。インタラクションの要のハードはビデオカメラで、ソフトはいわゆるパターン認識である。カメラは市販の家庭用を使用し、CPUとグラフィック・エンジンの機能はパーソナル・コンピュータ（FM TOWNS）一台で供給する。鏡を用いるのはリアルタイムで入力画像を左右反転するためである。

・動作原理

VIPは図2に示すように、2枚の画像の比較によって動作する。ここで、1枚目の画像は、対象（人など）のいない初期画面（A）であり、2枚目の画像は、ビデオカメラからの入力によって時々刻々更新される、対象の動く画面（B）である。

・2画面の比較方法

図2の画面（A）に存在しなかった物体を、画面（B）中から発見するためには、（A）と（B）との差分（引算）をとるのが簡単である。実際には、カラー画像を対象とするのでRGBに色分解してから差分をとればよい。しかしその後単純な差分だけでは実用的に不十分な場合があることがわかった。

・単純差分方式の問題点

2枚の画像の差分をとって、移動物体を検出するには、背景は不変であることが前提条件である。すなわち、1枚の画像の上に、アニメーションのセル画のように移動物体が存在していることが必要で、背景となる画像が変化してしまえば、いたるところに移動物体があることになる。

最近売行きが伸びているいわゆるハンディタイプのビデオカメラには、揃ってオートアイリス（自動絞り調節）機能が装備されている。機種によってその感度は異なるが、人間の瞳のように、画面全体が明るすぎる時には、光量を抑え、逆に暗すぎる時には光を集める。

たとえば、背景として黒いスクリーンを用いた場合、図2の画面（A）を記録する際には、画面全体が暗いので、ビデオカメラの絞りは開き気味となる。そこへ、白い服を着た人が入ると、今度は画面全体が明るくなるので、絞り気味へと変化する。これが画面（B）であるとする、たとえ同じ背景（黒いスクリーン）であったとしても、画面（A）と画面（B）との背景の明るさは異なる。これにより、誤動作を起こす場合がある。

・逆アイリス機能

オートアイリス機能の弊害を解除するため、VIPはオートアイリスの逆の機能を備えている。すなわち画面（B）全体に分散配置したセンサにより画面の平均明度を求め、アイリス機能の補正を行なう、逆アイリス機能である。

オートアイリスによってあまり影響を受けないパラメータを用いることも有効である。式（1）で演算されるHは色相を表す値で、オートアイリスの影響をあまり受けないパラメータであることを予備実験で確かめた上これを用いた。

$$\left. \begin{aligned} V &= \max(R, G, B), \quad T = \min(R, G, B) \\ H &= (G-B) * 60 / (V-T) \quad (R_{\max}) \\ H &= (B-R) * 60 / (V-T) + 120 \quad (G_{\max}) \\ H &= (R-G) * 60 / (V-T) + 240 \quad (B_{\max}) \end{aligned} \right\} \dots \dots \dots (1)$$

・色相変化の利用

オートアイリスの問題とは異なるが、人間は肌色に敏感である。顔色などの微妙な肌色の变化を見分ける。また、VIPを使う場合には、手などの肌色の部分で触る場合が多いので、式（1）で演算されるHの値をそのまま用いるのではなく、肌色に敏感となるようにした。即ち、色相の全方向を平等に取り扱わず、肌色方向についてのみ、色相の変化を増幅するよう補正を行なった。

この補正により、図3のような背景色に対して式（1）の結果は同じ量だけ離れている色を持つ物体①と②であっても、肌色領域に属する物体③は検出され易くなった。

これら3つの技術を結びつけたセンサによって、VIPは動作している。実際には、これら全ての演算を統合した演算テーブルを持ち、実行時にはこれらの演算は行なわないため、リアルタイム処理が可能となっている。

・「VIPボール」

VIPのコンセプト商品として開発したシリーズ第1弾アプリケーションが、「VIPボール」である。画面上を転がる仮想ボールをユーザーがさわると、ボールがはねかえる、といった仮想世界が実現されている。ユーザーとボールとの関係により、VR技術のインターフェイスとしての可能性を提案している。

また、この「VIPボール」の中には、他のボールがぶつかるとそのボールを飲み込んでしまうボールも表現しており、これにより仮想世界内部における仮想物体同士の対話の可能性を提示している。

Ⅲ. HAT-Vの概要〔3〕

HAT-Vを用いるアプリケーション作成の大まかな流れを図4に示す。企画を基に、画データ、音データ、シナリオプログラムの各ファイルをそれぞれ作成する。画データファイル（拡張子は.TIF）は、TOWNS標準TIFFファイル（32K色）形式を使い、16 dot×16 dotの画を約900枚分扱える。音データファイルは、効果音データファイル（.SND）、楽器データファイル（.FMB又は.PMB）、楽譜データファイル（.EUP）の3つを持つ。

シナリオファイル（.VIP）は、HAT-V専用言語で記述する。HAT-V言語は、10～20のコマンドを標準で持ち、「VIP」システムを直感的に記述できるプログラミング言語で、汎用コンピュータプログラムを知らない人にも非常に短時間で簡単にソフトを作成できる。この各ファイルをHAT-Vを用いて実行形式のファイル（.EXP）へと変換し、これをCD-ROMに焼けば「VIP」アプリケーション・ソフトとなる。

Ⅳ. HAT-V言語によるシナリオプログラム例

ここではHAT-V言語を使った簡単なシナリオプログラムを紹介する。リスト1はHAT-V言語によるシナリオプログラムの例である。このプログラムの実行イメージは、「画面上に1個のボールが回転しながら動いている。障害物に触れると跳ね返る。」である。以下でこのシナリオプログラムを各ブロックごとに見る。

・シナリオA

シナリオAは、ボールの画データファイルとセンサー定義ファイルを定義している。まず、「motion<オブジェクト名>」でオブジェクト名を「ball」とする。

次に「SENSOR<センサー定義ファイル名>」で、センサーを定義する。センサー定義ファイルは、オブジェクトごとのセンサーの位置を描いた画データで、ここでは「sball」とする。センサー定義ファイルの拡張子は省略して書く。

そして、「FIG<画データファイル名>」で画柄を定義する。なおFIGの前の数字は、画の表示時間を表す。ここでは、FIG命令を縦に8個並べてあるがこれは「ボールが回転している」というイメージを複数枚の画で表現している。つまり、「ball_1」から「ball_8」の8枚の画を使って、ボールが1回転しているように見せる。なお「ball_?」の拡張子は省略する。

・シナリオB～D

シナリオBでは、「defmotion<モーション名>=<オブジェクト名>」の書式に従って、シナリオAでFIG命令で定義した一連の動きを、<モーション名>で総称し定義する。

シナリオCでは、システム組み込み関数名及びユーザー作成関数名を定義する。書式は「extern<関数名>」である。リスト中には、「engine01」と「printf」の2つの関数が定義され、関数「engine01」はシステム組み込み関数で、「画面上でオブジェクトのセンサーが障害物に反射する」動きを記述するときに使用する。

もう一つの関数「printf」はC言語の標準関数である。HAT-V言語からは、C言語の関数も「extern」命令で宣言することにより呼び出すことができ、ユーザーはC言語を使ってより柔軟にシナリオプログラムを作成できる。

シナリオDでは、オブジェクトをクラス定義している。「class<クラス変数名>(X, Y, V, btheta, type)<インスタンス名>」で、クラス名と生成するインスタンスの最大個数を定義する。

「class_var<クラス変数名>, ……」でクラス変数を、そして「inst_var<インスタンス変数>, ……」でインスタンス変数をそれぞれ宣言する。

「new」以下ではオブジェクトの動きを定義している。「MOTION<モーション名>[type]」で、オブジェクトのモーション名を定義する。<モーション名>はシナリオBで既に定義した(ここでは「ball」)。

「beat_state」以下は障害物に当たり、センサーに反応した後の動きを記述している。「engine01」の戻り値「n」には、センサーに反応しているときに0以外の正值(反応が強いほど大きな値になる)が入る。センサーに反応したかどうかは、「n」値で判定できる。クラス定義は「end_class」で閉じる。

・シナリオE

最後にシナリオEでメイン関数を記述する。「main<ソフト名>」でソフトの名前を定義する。ここで「VIPball」である。

次の「NEW」で、シナリオDで定義したボール・オブジェクトを画面上に生成する。カッコ内は、オブジェクト生成時の初期値で、左からクラス名、初期X座標・Y座標、オブジェクトの移動スピード、オブジェクトの運動方向、インスタンス番号である。

「main_close」以下で終了処理を記述し、画面上に「END」の表示で終了する。「end_main」でメイン関数を閉じ、シナリオプログラムが完成する。

V. まとめ

「HAT」はHAT-VとHAT-Gの2機能を持つ。後者はCD-Gカラオケに特に有効であるが、未だインターフェイス部分に改良の余地があり、本稿ではHAT-Vの記述のみにとどめた。

いまのところ、仮想世界は簡単なグラフィックスであるが、メディアがもっと高度のグラフィックスや、音、音声などに拡がるに伴って奥行きと広がりを加えてゆく筈である。これと共に、HATはます

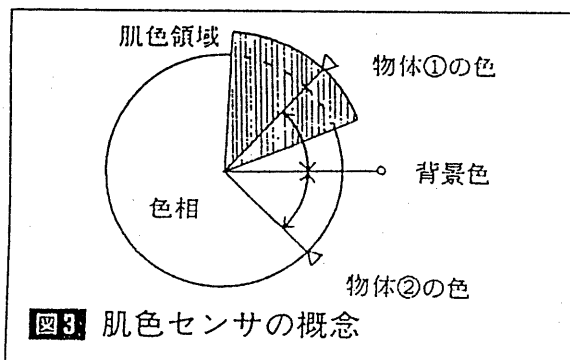
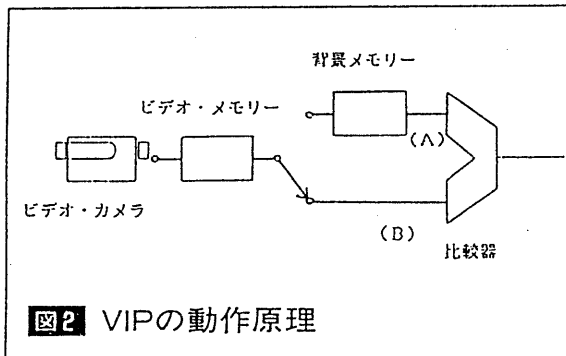
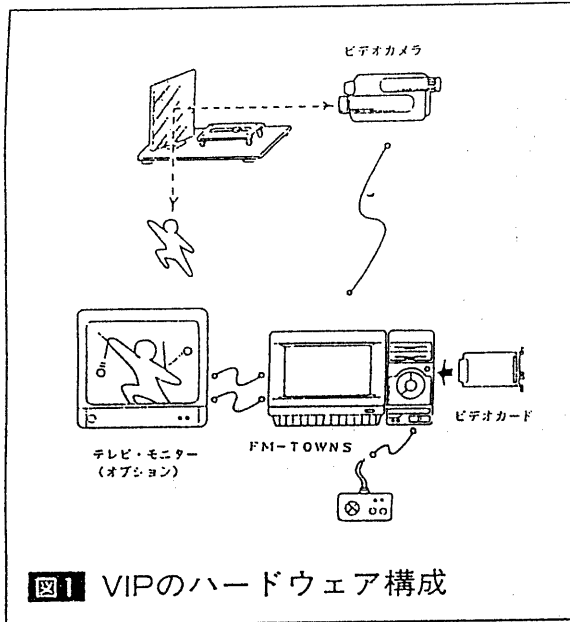
ますますユニークかつ高度なオーサリング・ツールとして完成度を高めて行くと考えている。

プログラマーの力量だけでは、人に感動を与えるほど美しいものや、思い出に残るような味のある演出はなかなかむつかしい。VRのように技術的に高度なものとなる程、芸術家のセンスを疎外する危険性がある。疎外要因を除き、このような能力を活用できるツールを提供することによって、VIPのアプリケーション分野はまだまだ広がる。この目的で開発されたツールがここで紹介した「HAT-V」である。当社では、これにより社外より芸術家を招いて、VIPシリーズの開発に携わって頂いている。目下の所、芸術家が一人歩きして、アプリケーションを作り上げるという所までは至っていない。しかしプログラマーが協力者の形で、芸術家の要求する演出のためのプログラムを提供する形を取り易くなっている。その機器構成の単純さからみて、VIPの応用分野は今後ますます広がるものと考えられ、特にアミューズメントゲーム、教育、プレゼンテーション等の分野に注目している。

本稿は、(株)CSK総合研究所が開発したVRソフト「VIP」、および(株)ハイパーメディアならびに情報処理開発協会（IPA）が共同開発したそのオーサリング・ツール「HAT」の概要を示すものである。また本稿の作製には、(株)ハイパーメディア社長室 田中みさおさん、橋本葉子さんの御協力を得た。付記して感謝の意を表す。

参考文献

- [1] 原島 博, 「ハイパーメディアとバーチャルリアリティ」
『CD-I news』 Vol. 3, 4
- [2] 岩田 憲治, 「バーチャル・リアリティ」の世界
『Expert Systems』 Vol. 5 (株)CSK総合研究所)
- [3] 『HAT-V USER'S MANUAL』 (株)ハイパーメディア)



リスト1

各モーションの定義

```

motion ball
SENSOR sbball
3 FIG ball_1
3 FIG ball_2
3 FIG ball_3
3 FIG ball_4
3 FIG ball_5
3 FIG ball_6
3 FIG ball_7
3 FIG ball_8

```

モーション (配列) の定義

```

definition balls = ball

```

```

extern engine01(:); ..... 反解型エンジン
extern print1(:);

```

```

class abc(x, y, v, btheta, type)10
class __ var v, n, btheta, theta, type, depth
new

```

MOTION balls[type]

```

beat__state n=engine01(&x, &y, &v, &btheta, SENI)
end __class

```

メイン関数定義

```

main vipball
NEW(abc, $100, $100, $1.0, 128, 0)
main__close
main__print("ENDYn)
end __main

```

HAT-V 仕様の流れ

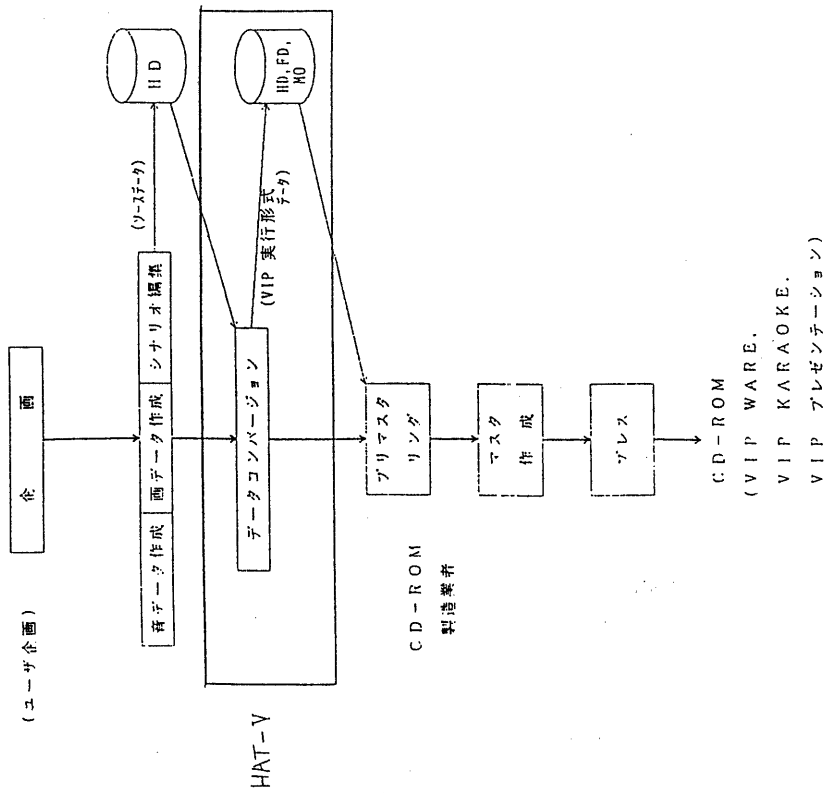


図4