

視覚誘導情報獲得モデルに基づくマルチメディア情報システム — *VideoReality* —

片岡 良治 佐藤 哲司 井上 潮

NTT情報通信網研究所

データベースに蓄積された映像中のオブジェクトに関する情報をユーザがインタラクティブに検索できるマルチメディア情報システム *VideoReality* について述べる。 *VideoReality* は人間が視覚を通して情報を獲得する過程をモデル化しており、ユーザはコンピュータのディスプレイに再生される映像内の空間で自由に視野を移動させて興味のあるオブジェクトを眺めることができる。視野の自由な移動を実現するために、 *VideoReality* は広大な空間を領域的に分割撮影した映像集合から時間的かつ空間的に連続する1つの映像空間を構成し、それを3階層に構造化している。本稿では、映像空間の管理法と現在試作中のアプリケーション「*電視水族館*」について述べる。

***VideoReality* : A Multimedia Information System Based on an Information Retrieval Process Model Induced by the Sense of Sight**

Ryoji Kataoka Tetsuji Satoh Ushio Inoue

NTT Network Information Systems Laboratories

1-2356 Take, Yokosuka-shi, Kanagawa, 238-03 Japan
{kataoka, satoh, inoue}@syrinx.ntt.jp

This paper describes a multimedia information system called *VideoReality*, where users can get information about objects in a video database interactively. *VideoReality* is based on the information retrieval process occurring when human observes objects in the real world with the sense of sight. Users can move their sight in the virtual video space and see their favorite objects freely. *VideoReality* provides a video space with continuous time and space dimensions from a set of stored video data. The video space has three abstraction levels. This paper shows the architecture and management of the video space as well as a prototype application called 'Electronic Aquarium'.

1. はじめに

人間が視覚を通して情報を獲得する過程は、(1) 視線を移動させて外界の様々なオブジェクトを観察し、(2) 観察により得た特定の視覚情報(オブジェクトの形、色、動き、等)に興味を示し、(3) 興味を持ったオブジェクトに注目し、(4) そのオブジェクトに関する詳細な情報を得るための行動(百科事典を引く、専門家に問い合わせる、等)をとる、という流れにモデル化できる。例えば、我々は水族館で魚を観察する際、先ず水槽中の様々な魚に目をやり、次にその中から色や形に興味を持った特定の魚の動きを追い、更に水槽に掲示されている凡例を読むことでその魚に関する詳細な情報を得るといった行動を一般にとる。また、サッカーの試合の観戦でボールの動きに合わせて様々な選手に視線を移動させる一方、特定の選手の動きに興味を持ち、ボールの動きとは無関係に暫くその選手に注目し、その選手の名前が分らなければ掲示板に目をやるといった行動もこのモデルに合致する。我々はこのモデルを視覚誘導情報獲得モデルと呼び、ユーザがこのモデルに基づいてデータベースに蓄積された映像中のオブジェクトに関する情報をインタラクティブに検索できるマルチメディア情報システム *VideoReality* を構築中である。

VideoReality を構築する上では、水族館の水槽やサッカーのグラウンドといった広大な空間を映像化してデータベースに格納し、それをコンピュータのディスプレイという狭小な空間に再生する手法が問題となる。広大な空間全体を一括して撮影した映像を単純に格納・再生したのでは映像中のオブジェクトが小さくなり過ぎてしまい、視覚誘導情報獲得モデルを効果的に実現することができない。そこで我々は広大な空間を領域的に分割して撮影した映像集合をデータベースに格納し、再生時にはユーザの現在の視野に対応する領域の映像のみをディスプレイに表示する手法でこの問題に対処する。*VideoReality* は映像集合を論理的に連続した1つ

の映像空間として管理し、映像空間の論理座標を指定することでその位置に対応する視野の映像をディスプレイに表示する。従って、ユーザの視線移動を論理的な映像空間上での座標移動にマッピングすることで、視線の移動に伴う視野の変化を映像としてコンピュータのディスプレイ上に表示できる。

このように *VideoReality* は、領域的に分割撮影された映像集合から時間的・空間的に連続した1つの映像空間を構成する点で従来の映像データを扱うマルチメディア情報システム^[7]と大きく異なる。本稿では *VideoReality* の概要を映像空間の管理法を中心に述べ、現在試作中のアプリケーション「*テレビ水族館*」について報告する。

2. 技術課題

視覚誘導情報獲得モデルに基づく情報検索過程は、視線を動かして自分の周囲(観察空間)に存在する様々な事物(オブジェクト)を眺める視覚誘導ステップと、興味を持った特定のオブジェクトを眺めながらそのオブジェクトに関連する情報を得る情報獲得ステップの2段階に大別できる。

現実の観察空間は空間全体を眺めるために視線を移動させる必要があるほど広大であり、動きのあるオブジェクトの状態は時々刻々変化している。従って、映像を通してユーザに視覚誘導ステップを行なわせるためには、映像情報の持つ空間的かつ時間的な広がりデータベースにより管理する技術が必要となる。広大な観察空間に比べてコンピュータのディスプレイは狭小である。このため観察空間全体を1台のビデオカメラで一括撮影した映像を単純にデータベースに格納したのでは再生映像中のオブジェクトが小さくなり過ぎてしまい、情報獲得ステップへ移行するきっかけとなるような視覚的效果が得られない。情報獲得ステップへの移行を促すためには現実の観察空間を眺めるユーザの視野に相当する映像をディスプレイに表示す

ることが望ましい。そのためには複数のビデオカメラで観察空間を分割撮影して得た映像集合を論理的に連続した1つの映像空間としてデータベース管理する手法が必要である。また、空間的な広がりのある映像をユーザの任意の視野移動に追従させてリアルタイムに表示する技術も重要となる。近年、映像データベースに関する研究が盛んであるが、これまでの研究は映像情報の時間的な連続性に着目したものが多く[4,5,6,8,9]、空間的な連続性を同時に考慮したものはない。

一方、情報獲得ステップにおける特定のオブジェクトを眺める動作を実現するには、ユーザが指定した映像中のオブジェクトを認識し、そのオブジェクトの追跡映像を映像データベースから生成する技術が必要となる。前述のように、VideoRealityでは映像情報の時間的な連続性に加えて空間的な連続性を映像データベースにより管理している。この機能によれば、表示映像中の任意のポイントを映像データベースが提供する論理的な映像空間の一意の座標にマッピングすることが可能である。従って、オブジェクト認識を実現する上では、認識の対象となるオブジェクトの動きを映像空間の論理座標によりトレースしてデータベース管理する手法が有効と考えられる。トレースデータは特定のオブジェクトの追跡映像を映像データベースから生成する上でも有効に利用可能と考えられる。オブジェクト認識に関しては画像処理技術[2]を応用するアプローチも考えられるが、現状の画像

処理技術では情報検索のインタラクティブ性を劣化させる恐れがあり、VideoRealityへの適用に向かない。

3. システム構成

VideoRealityのシステム構成を図1に示す。ユーザはディスプレイに表示される映像を通して視覚誘導ステップと情報獲得ステップを実行する。視覚誘導ステップにおいてユーザは視野の移動をマウスでVideoRealityサーバに要求する。ディスプレイには観察空間を眺めるユーザの視野に相当する映像が表示されており、ユーザが視野の移動を要求するとディスプレイに映し出される空間が要求方向に移動する。情報獲得ステップはユーザが表示映像中のオブジェクトをマウスにより指定することで実行できる。VideoRealityサーバは指定された時点からオブジェクトの追跡を開始すると共に、そのオブジェクトに関する属性情報をウィンドウに表示する。

VideoRealityサーバは映像システムとシナリオシステムを含む。これらの機能の概要を以下に述べる。

(1) 映像システム

映像システムは映像情報を時間的かつ空間的に連続した論理的な映像空間としてデータベース管理する機能を持つ。映像空間は観察空間を撮影した映像を映像入力部を通して所定の

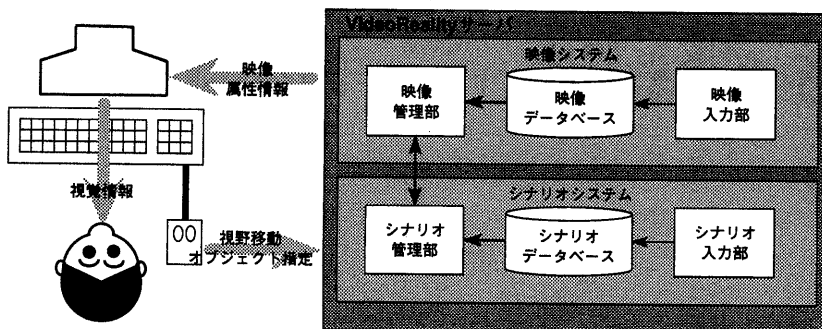


図1 システム構成

フォーマットに変換し、映像データベースに格納することで構築できる。ユーザの視野移動要求は後述するシナリオシステムを通して映像管理部に伝わる。映像管理部はユーザの要求に合わせて映像情報を映像データベースから取り出し、それをディスプレイ上にリアルタイム再生する。また、ユーザが表示映像中のオブジェクトをマウスで指定すると、映像管理部は指定ポイントに対応する映像空間の論理座標を求め、指定されたオブジェクトの追跡映像を再生するための指示をシナリオシステムに要求する。

(2) シナリオシステム

シナリオシステムは映像システムにより再生する映像の筋書き(シナリオ)をデータベース管理する機能を持つ。シナリオデータベースには、トレース情報、追跡シナリオ、属性情報という3種類のデータが格納される。トレース情報とは映像中のオブジェクトの動きと状態を映像空間の論理座標によりトレースした時系列データであり、認識対象となるオブジェクト毎に作成される。追跡シナリオとはユーザが指定したオブジェクトの追跡映像を映像システムにより再生するための手続きを記述したものである。属性情報は映像中のオブジェクトに関する詳細な情報を含む。これらのデータはシナリオ

入力部を用いて作成する。シナリオ管理部は3種類のデータを利用して情報獲得ステップを実現する。情報獲得ステップでは(1)映像システムから受け取った映像空間の論理座標をキーとしてトレース情報を検索してユーザの指定したオブジェクトを認識し、(2)認識したオブジェクトに対する追跡シナリオを用いて映像システムにより追跡映像を再生し、(3)認識したオブジェクトに関する属性情報を求めてディスプレイに表示する。また、シナリオ管理部はユーザの視野移動要求を解釈し、要求された視野移動に対応した映像を再生するための指示を映像システムに与える。

4. 映像空間の管理法

4.1 3階層モデル

映像システムにより提供される映像空間は、従来のデータベースシステムにおける内部スキーマ、概念スキーマ、外部スキーマという3階層の抽象化レベル[1]に対応付けて、蓄積映像空間、論理映像空間、表示映像空間の3階層に構造化されている(図2)。

(1) 蓄積映像空間

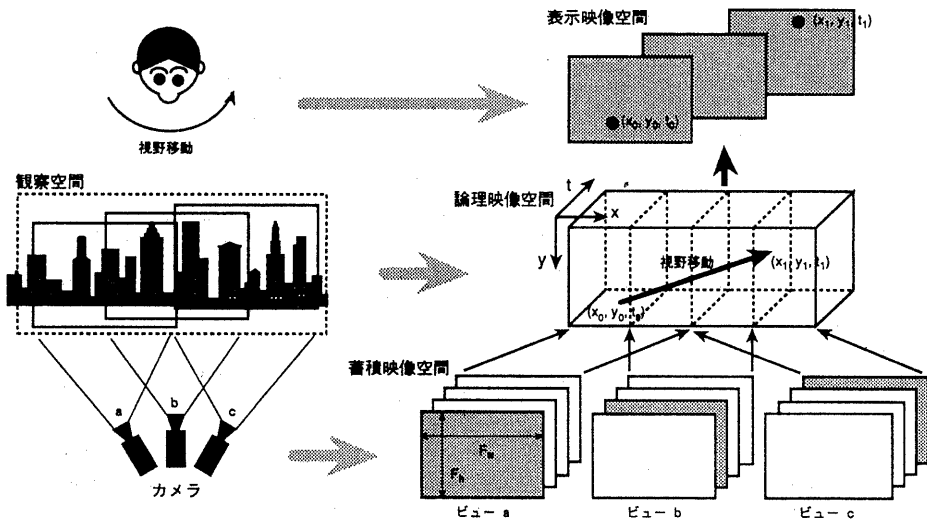


図2 映像空間の階層構造

蓄積映像空間は複数のビデオカメラで観察空間の異なる部分領域を固定した位置から同時に撮影した映像により表現される空間である。各映像は観察空間の特定の部分領域のみをカメラアングルを変化させることなく撮影したフレーム系列であり、ユーザの1つの視野に対応することからビューと呼ばれる。ビューは必ずしもビデオカメラで撮影した映像そのものではなく、ビデオカメラで撮影した複数の映像を合成して1つのビューを構成することもできる。例えば図2ではビュー a の右半分のフレームとビュー c の左半分のフレームを合成してビュー b を作成することもできる。

(2) 論理映像空間

論理映像空間は観察空間と1対1に対応付けられた論理的な空間であり、2次元の空間軸(x, y)と時間軸 t を持つ。1つの論理映像空間座標(x, y, t)は蓄積映像空間に含まれる1つのフレームに一意にマッピングできる。論理映像空間のサイズ(幅×高さ)はビューの構成とそのフレームのサイズに依存して定まる。例えば図2ではビュー a, b 間およびビュー b, c 間に映像領域の重複が1/2ずつあるため、論理映像空間のサイズは $2F_w \times F_h$ となる。

(3) 表示映像空間

表示映像空間はディスプレイ上に再生される映像により表現される空間である。再生映像は視線を移動させて観察空間を眺めるユーザの視野に映る光景を表しており、視野移動を論理映像空間上での座標移動にマッピングすることで再生映像のフレーム系列を決定できる。

4. 2 蓄積映像空間の構造

映像システムは観察空間に対するユーザの視野移動を論理映像空間上での座標移動にマッピングし、座標移動に応じてディスプレイに表示するビューを切り替えることで視野移動を映像化する。例えば図2では論理映像空間座標(x_0, t_0)から(x_1, y_1, t_1)への視野移動を、ディスプレイに表示するビューを a, b, c の順に切り替えることで表現している。視野の移動単位は各ビューに含まれる映像領域から隣接するビューとの重複部分を除いた領域のサイズとなり、図2ではフレームサイズの1/2の幅で左から右へ移動することになる。つまり、スムーズな視野移動を映像により表現するには隣接するビュー間の重複領域サイズを大きくすることが望ましい。

一方、隣接するビュー間の重複領域サイズを大きくすることは蓄積映像空間により提供すべきビュー数を増加させる。そのため、単純に1つのビューに対応付けて1つのフレーム系列を設けたのでは映像データベースの容量が増大してしまう。映像システムでは映像データベースの容量がビュー数に対してリニアに増加することを防止するため、1つのビューを複数のサンプルに分割してデータベースに格納しておき、複数のビュー間で同一のサンプルを共有可能としている。サンプルとはビューを構成するサイズ $F_w \times F_h$ のフレームを $N_x \times N_y$ のメッシュに等分割して得られるサイズ $F_w/N_x \times F_h/N_y$ のフレームの系列であり、 $N_x N_y$ 個のサンプルを合成

一方、隣接するビュー間の重複領域サイズを大きくすることは蓄積映像空間により提供すべきビュー数を増加させる。そのため、単純に1つのビューに対応付けて1つのフレーム系列を設けたのでは映像データベースの容量が増大してしまう。映像システムでは映像データベースの容量がビュー数に対してリニアに増加することを防止するため、1つのビューを複数のサンプルに分割してデータベースに格納しておき、複数のビュー間で同一のサンプルを共有可能としている。サンプルとはビューを構成するサイズ $F_w \times F_h$ のフレームを $N_x \times N_y$ のメッシュに等分割して得られるサイズ $F_w/N_x \times F_h/N_y$ のフレームの系列であり、 $N_x N_y$ 個のサンプルを合成

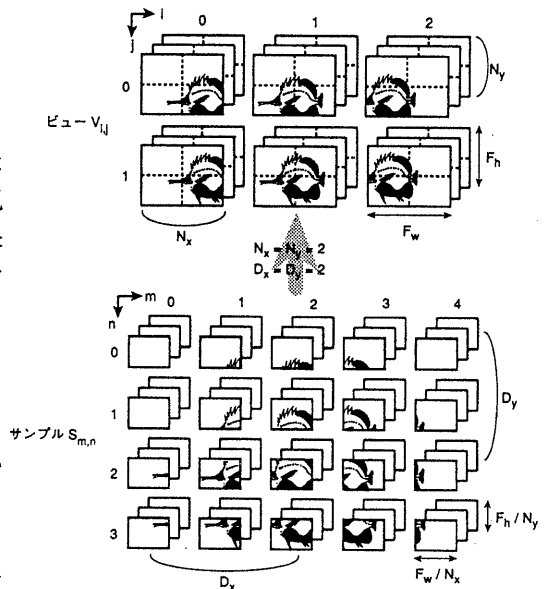


図3 蓄積映像空間の構造

したものが1つのビューとなる。図3に蓄積映像空間の構造を示す。\$D_x\$ と \$D_y\$ はビューの合成に用いるサンプルの間隔を示すパラメータであり、ビュー \$V_{ij}\$ を構成するサンプルの集合は次のように求められる。

$$V_{ij} = \{S_{m,n} | m = i, i + D_x, i + 2D_x, \dots, i + (N_x - 1)D_x, \\ n = j, j + D_y, j + 2D_y, \dots, j + (N_y - 1)D_y\} \quad (1)$$

つまり、\$D_x\$ と \$D_y\$ は隣接するサンプル間の映像領域の重複度を表している。\$N_x\$ と \$N_y\$ の設定値が同一の蓄積映像空間を比較するならば、\$D_x\$ と \$D_y\$ の値が大きいほど隣接するサンプル間の重複領域サイズは大きくなり、よりスムーズな視野移動を表現可能となる。x 軸方向（左右方向）への視野の移動単位 \$M_x\$ と y 軸方向（上下方向）への視野の移動単位 \$M_y\$ はそれぞれ次式により求められる。

$$M_x = F_w / N_x / D_x \quad (2)$$

$$M_y = F_h / N_y / D_y \quad (3)$$

\$M_x\$ と \$M_y\$ の目標値が定められているとき、4つのパラメータ \$N_x\$, \$N_y\$, \$D_x\$, \$D_y\$ は映像システムを実装する環境に合わせて次のように設定される。ディスクを有効に利用する上では、\$D_x\$ と \$D_y\$ をできる限り小さくして隣接するサンプル間の重複領域サイズを小さくし、データベース容量を小さく抑えるのが望ましい。しかし、\$D_x\$ と \$D_y\$ を小さくすると、目標とする \$M_x\$ と \$M_y\$ を実現するための \$N_x\$ と \$N_y\$ が大きくなり、サンプルからビューを合成するオーバーヘッドが大きくなる。このオーバーヘッドは再生映像の品質を劣化させるため、使用するコンピュータの性能に応じて \$N_x\$ と \$N_y\$ の上限値は定まる。つまり、\$N_x\$ と \$N_y\$ が上限値以下となる範囲で最小の \$D_x\$ と \$D_y\$ を求めればよい。

4. 3 論理映像空間のマッピング

映像システムは1つの論理映像空間座標 \$(x, y, t)\$ を1つのビューフレームに次のようにマッピングする。

(1) ビューの選択

映像システムは蓄積映像空間のビュー集合 \$\{V_{ij} | i = 0, \dots, P_x - 1, j = 0, \dots, P_y - 1\}\$ から論理位置 \$(x, y)\$ の映像を最も中心に映し出すビューを選択する。ここで \$P_x\$, \$P_y\$ はそれぞれ x 軸方向、y 軸方向のビューの個数であり、全体で \$P_x P_y\$ 個のビューが存在する。論理映像空間サイズを \$G_w \times G_h\$ とするとき、選択すべきビュー \$V_{ij}\$ は次式により求められる（図4）。

$$i = \begin{cases} 0 & (0 \leq x < F_w/2 - M_x/2) \\ \lfloor (x - F_w/2 + M_x/2) / M_x \rfloor & (F_w/2 - M_x/2 \leq x < G_w - F_w/2 + M_x/2) \\ P_x - 1 & (G_w - F_w/2 + M_x/2 \leq x < G_w) \end{cases} \quad (4)$$

$$j = \begin{cases} 0 & (0 \leq y < F_h/2 - M_y/2) \\ \lfloor (y - F_h/2 + M_y/2) / M_y \rfloor & (F_h/2 - M_y/2 \leq y < G_h - F_h/2 + M_y/2) \\ P_y - 1 & (G_h - F_h/2 + M_y/2 \leq y < G_h) \end{cases} \quad (5)$$

ここで視野の移動単位 \$M_x\$, \$M_y\$ は次式により求められる値である。

$$M_x = (G_w - F_w) / (P_x - 1) \quad (6)$$

$$M_y = (G_h - F_h) / (P_y - 1) \quad (7)$$

(2) フレームの選択

映像システムは選択したビュー \$V_{ij}\$ を構成するフレーム系列 \$\{F_k^{ij} | k = 0, \dots, L - 1\}\$ から論理時刻 \$t\$ に表示すべきフレームを選択する。ここで \$L\$ はフレーム系列に含まれるフレーム数である。ビュー \$V_{ij}\$ にはそのフレーム系列の標準再生速度 \$R_{ij}\$ がフレーム/秒の単位で与えられており、隣接するフレームを \$1/R_{ij}\$ 秒間隔で表示することでそのビューの映像を標準速度で再生できる。従って、選択すべきフレーム \$F_k^{ij}\$ は \$t\$ の

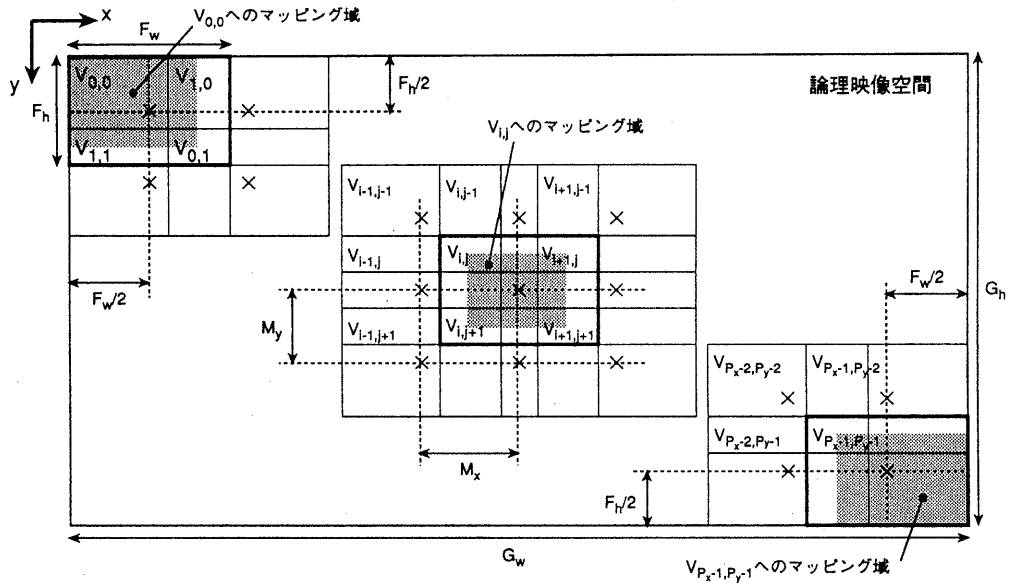


図4 ビューの選択

単位を秒とすると次式により求められる。

$$k = \lfloor t R_{i,j} \rfloor \quad (8)$$

理時刻 t_c は、前回フレーム選択を行なった実時刻 T_b と T_b に対応する論理時刻を t_b を用いて次式により求められる。

$$t_c = t_b + K(T_c - T_b) \quad (9)$$

4.4 表示映像空間の形成

映像システムはシナリオシステムから次の形式で映像再生の要求を受ける。

$$\text{PLAY: } (x_p, y_p, z_p, t_p) \xrightarrow{K} (x_q, y_q, z_q, t_q)$$

これは論理映像空間座標 (x_p, y_p, t_p) から (x_q, y_q, t_q) までの映像を、フレームサイズを z_p 倍から z_q 倍まで一定の割合で拡大あるいは縮小しながら標準速度の K 倍の速さで再生することを意味する。映像の再生方向は $t_p < t_q$ ならば順方向、 $t_p > t_q$ ならば逆方向となり、 $t_p = t_q$ ならばスチルとなる。再生速度は $K = 0$ ならばスチル、 $0 < K < 1$ ならばスロー再生、 $K = 1$ ならば標準再生、 $K > 1$ ならば高速再生となる。

映像システムは表示すべきフレームをリアルタイムに選択しながら再生映像のフレーム系列を形成していく。現在の実時刻 T_c に対応する論

尚、初期状態では $T_b = T_c$ 、 $t_b = 0$ である。従って、 T_c に表示すべき論理位置 (x_c, y_c) とフレーム倍率 z_c は次のように求められる。

$$x_c = x_p + E(x_q - x_p) \quad (10)$$

$$y_c = y_p + E(y_q - y_p) \quad (11)$$

$$z_c = z_p + E(z_q - z_p) \quad (12)$$

ここで、

$$E = (t_c - t_p) / (t_q - t_p) \quad (13)$$

である。実時刻 T_c に表示すべきフレームは前節に示した式 (4) ~ (8) を用いて求められる論理映像空間座標 (x_c, y_c, t_c) に対応するフレームを、 $z_c F_w \times z_c F_h$ のサイズに拡大あるいは縮小したものである。

5. アプリケーション例「電視水族館」

現在、VideoReality のアプリケーションとして「電視水族館」を試作中であり、これまでに映像システム部分が完成した。「電視水族館」のコンセプトを図5に示す。「電視水族館」はユーザが映像中の魚に関する情報を、実際に水族館で魚を観察するときと同様の感覚で検索できるアプリケーションである。ユーザは映像化された水槽上で自由自在に視野を移動させながら魚を観察できる。観察を通して興味を持った魚をマウスでクリックすると、クリックした魚がその時点から自動的に追跡されると共に、その魚の名前や生息地等といった属性情報がウィンドウに表示される。また、凡例パネルに表示された任意の魚をクリックすることで、それに対応する魚の動きを追跡した映像を再生できる。

映像システムの実装環境を図6に示す。サンプルはJPEG圧縮されたフレームの系列としてハードディスクに格納されており、米 Parallax Graphics 社製のビデオボード XVideo^[3]によりリアルタイムに伸長され、SUN SPARC-10のディスプレイ上に映像として表示される。サンプル用の映像は複数のビデオカメラで水槽を分割撮

表1 映像空間仕様

論理映像空間サイズ ($G_x \times G_y$) [ピクセル]	1810 × 918
ビューサイズ ($z_c F_x \times z_c F_y$) [ピクセル]	640 × 480 ~ 960 × 720
ビュー数 (V_x, V_y)	11, 4
視野移動単位 (M_x, M_y) [ピクセル]	117, 146
サンプル数 (P_x, P_y)	11, 4
1ビューあたりのサンプル数 (N_x, N_y)	1, 1
サンプル長 [秒]	120

影する方法ではなく、1台のハイビジョン方式のビデオカメラで水槽全体を撮影し、その映像をNTSC方式の標準サイズ(640×480ピクセル)に分割する方法で作成した。フレームサイズ960×720ピクセル(標準サイズの1.5倍)の映像を毎秒12フレーム程度の品質で再生可能である。

映像空間の仕様を表1に示す。実装環境の性能や機能の制約上、今回は1ビュー1サンプルの構成をとっており、x軸方向に11個、y軸方向に4個のビューを設けるに留まっている。

ユーザインタフェースを図7に示す。現在は映像システム独自のパネルから各種映像再生の指示を与える構成となっている。パネルAにより再生速度の指定が、パネルBにより視野の移動方向の指定が行なえる。パネルCは論理映像空間全体を表しており、現在表示されている映

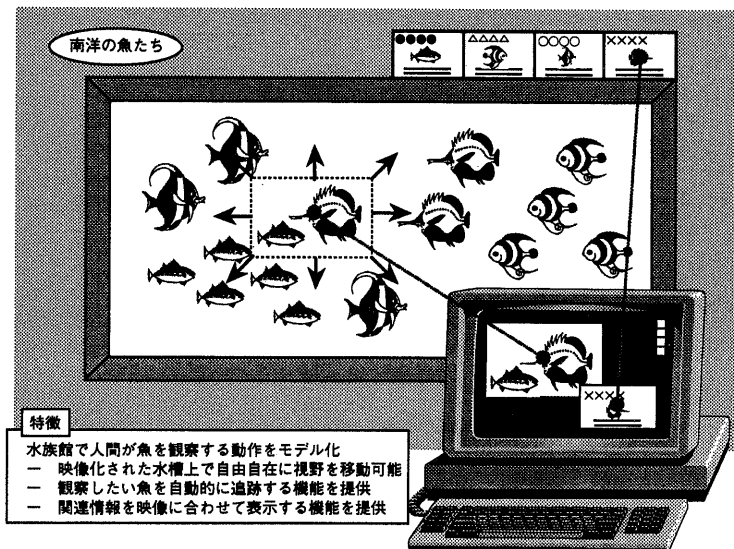


図5 電視水族館のコンセプト

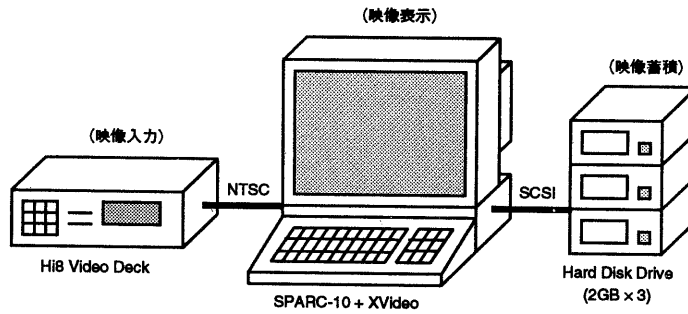


図6 実装環境

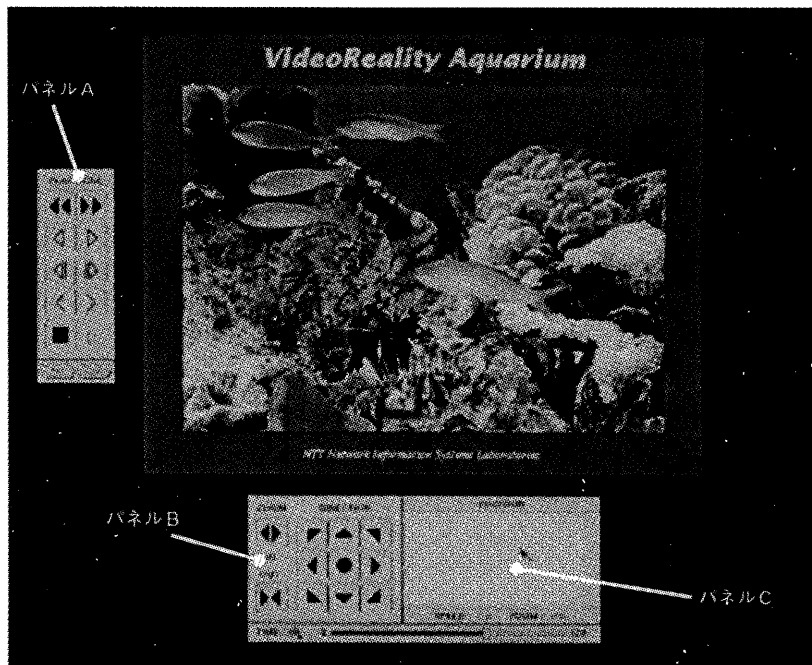


図7 ユーザインタフェース

像の位置が×印により示されている。視野の移動はパネルCの×印をマウスでドラッグして移動させることでも指示できる。

6. おわりに

本稿では、視覚誘導情報獲得モデルに基づいてユーザがデータベースに蓄積された映像中のオブジェクトに関する情報をインタラクティブに検索できるマルチメディア情報システム

VideoReality について、映像空間の管理法を中心に述べた。*VideoReality* では広大な観察空間を領域的に分割撮影した映像集合から時間的かつ空間的に連続した1つの映像空間を構成し、蓄積映像空間、論理映像空間、表示映像空間の3階層に構造化する。映像空間は映像の物理的な格納構造を意識することなくアクセスでき、観察空間と1対1に対応付けられた論理映像空間の座標により表示ポイントを指定できる。この機能は視覚誘導情報獲得モデルを実現する上で必

須となる映像空間上での視野移動を容易に映像化可能としている。

今後は *VideoReality* のもう 1 つの特徴であるシナリオの管理法について検討を進め、「電視水族館」への実装を通して技術確認を行なっていく予定である。

参考文献

- [1] 国井, 大保: データベース・システムの原理, 日本コンピュータ協会 (1985).
- [2] 高木, 下田: 画像解析ハンドブック, 東京大学出版会 (1991).
- [3] Parallax Graphics, Inc.: *XVideo User's Guide* (1991).
- [4] Gibbs, S., Breiteneder, C., Tschritzis, D.: Audio/Video Databases: An Object-Oriented Approach, *Proc. 9th Data Engineering Conf.*, pp.381-390 (1993).
- [5] 田淵, 村岡: 動画像データベース中の系列データを指定する条件の不完全さを許容できる問合せ処理と MeSOD モデル, 電子情報通信学会論文誌 D-I, Vol. J76-D-I, No. 6, pp.288-299 (1993).
- [6] Oomoto, E., Tanaka, K.: OVID: Design and Implementation of a Video-Object Database System, *IEEE Trans. Knowledge and Data Engineering*, Vol. 5, No. 4, pp.629-643 (1993).
- [7] 田中, 下條: 動画像データベース技術の現状と未来—Obase プロジェクトにおけるビデオ動画像の扱い—, アドバンスト・データベースシステム・シンポジウム'93 論文集, pp. 55-62 (1993).
- [8] 増永: マルチメディアデータモデル OMEGA における音データと動画像データのサポート, アドバンスト・データベースシステム・シンポジウム'93 論文集, pp.163-179 (1993).
- [9] 有澤, 由井, 富井: 映像データベースシステムの構成の一方式, アドバンスト・データベースシステム・シンポジウム'93 論文集, pp. 181-190 (1993).