

動画ハイパーメディアシステムの構築方式 とその応用に関する研究

平成16年度

阿倍 博信

要旨

本論文では動画ハイパーメディアシステムの構築方式とその応用に関する研究について述べる。

動画ハイパーメディアシステムとは、動画中に登場する人や物などのオブジェクトを含む領域に対して関連情報をリンクとして設定しておき、再生時にその領域をマウス等で直接指定することによって、関連情報を直観的に検索できるシステムであるが、動画ハイパーメディアシステムの効率的な構築手法が確立されていないため、ほとんど普及していないのが現状である。

本論文では、動画ハイパーメディアシステムの構築方式に関する構成要素として、バックエンド技術、オーサリング技術、配信技術、応用に分類し、構成要素ごとに技術課題を設定し、その課題を解決することを目的とする。

バックエンド技術については動画データの効率的な管理方式の確立を目的として、ビデオオブジェクトモデルとその管理システムを提案する。オーサリング技術については、アンカーの効率的な設定方式の確立を目的として、自動追跡と手動修正を組み合わせたアンカー設定方式を提案する。配信技術については、映像と付加情報の同期配信技術の確立を目的として、拡張イベントモデルを提案する。応用については、初等教育への応用を目的とした動画ハイパーメディアシステムについて提案する。

本論文の構成を以下に示す。第2章では、関連研究について述べる。

第3章では、動画データの効率的な管理方式の確立を目的として、ビデオオブジェクトモデルとその管理システムについて提案する。さらに動画応用システムにおける効率的な動画データ管理を目的としてビデオオブジェクト管理システムを設計、構築し、評価システムとして動画ハイパーメディアシステムを構築した結果について述べる。

第4章では、アンカーの効率的な設定方式の確立を目的として動画像解析に基づく自動追跡と手動修正を組み合わせた方式を提案する。提案した方式に基づきアンカー設定ソフトウェアを構築し、スポーツ映像を対象としてアンカー設定方式の評価実験を行った結果について述べる。

第5章では、映像と付加情報の同期配信方式の確立を目的として拡張イベントモデルを提案する。提案した拡張イベントモデルを Windows Media 及び MPEG-2 TS を対象としてエンコード/デコードライブラリとして実装し、実装したライブラリを使用して評価システムを構築し、評価システムの実時間同期性能について評価実験を行った結果について述べる。

第6章では、初等教育への応用を目的とした動画ハイパーメディアシステムの設計、構築及び有効性の評価について述べる。システムの設計にあたり、システム要件について設定を行い、設定した要件に基づき、システムの設計及び実装を行った。実装したシステムを初等教育用のマルチメディア教育コンテンツに適用した結果について述べる。

第7章で本研究のまとめを行った結果、動画ハイパーメディアシステムを構成するバックエンド技術、オーサリング技術、配信技術、応用についてそれぞれ設定した目標を達成することができ、提案した構築方式について、有効性を確認することができた。

Abstract

This paper describes a study on the constructing method for video hypermedia system and its applications.

A video hypermedia system is a system, which makes hyperlinks between the area including the objects in movie and related information and be able to navigate hyperlinks intuitively by selecting the area in movie directly with mouse, but it doesn't spread as affairs stand because an effective constructing method for video hypermedia system is not established.

This paper classifies the technical components of the constructing method of a video hypermedia system into backend technology, authoring technology, delivery technology and applications, and sets technical issues to each component, and aims to solve these issues.

This paper proposes a video object model and management system for establishment of efficient management method of video data about backend technology, anchor setting method by combination of automatic tracking and manual editing for establishment of efficient anchor setting method for video data about authoring technology, an extended event model for establishment of synchronized delivery method of video data and attached information about delivery technology, and a video hypermedia system for elementary education about applications.

This paper is composed as follows. Chapter 2 describes related work.

Chapter 3 proposes a video object model and management system for establishment of efficient video management method, and describes design and implementation of video object management system for efficient management of video data at video application system, and implementation of a video hypermedia system as an evaluation system.

Chapter 4 proposes an anchor setting method by combination of automatic tracking based on motion picture analysis and manual editing for establishment of efficient anchor setting method, and describes implementation of anchor setting software based on proposed method and experiments for evaluation of sport video.

Chapter 5 proposes an extended event model for establishment of synchronized delivery method of video data and attached information, and describes implementation of proposed model as encode/decode libraries for Windows Media and MPEG-2 TS, and development of evaluation system using libraries, and experiments for evaluation of real time synchronization performance.

Chapter 6 describes design, development and evaluation of video hypermedia system for elementary education. At designing phase of system, we defined system requirements, and designed and developed system on these requirements. After that we created multimedia content for elementary education using this system and made experiments.

Chapter 7 concluded these studies, and accomplished the goal about backend, authoring, delivery and applications constructing video hypermedia system, and confirmed the effectiveness of proposed constructing method.

目次

第 1 章	序論	1
1.1	本研究の背景	1
1.2	本研究の目的	3
1.2.1	バックエンド技術	4
1.2.2	オーサリング技術	4
1.2.3	配信技術	5
1.2.4	応用	6
1.2.5	本研究の目的	6
1.3	本論文の構成	7
第 2 章	関連研究と本研究の位置付け	8
2.1	バックエンド技術	8
2.1.1	OVID	8
2.2	オーサリング技術	9
2.2.1	手動による設定方式	11
2.2.2	動画処理を用いた自動設定方式	13
2.2.3	自動設定結果を手動で修正する方式	16
2.3	配信技術	17
2.3.1	Harmony	17
2.3.2	SMIL	19
2.3.3	RealMedia	20
2.3.4	QuickTime	20
2.3.5	Name-it	21
2.4	応用	21
2.4.1	Cmew	21
2.5	本研究の位置付け	22
2.5.1	バックエンド技術	23
2.5.2	オーサリング技術	23
2.5.3	配信技術	23
2.5.4	応用	23
第 3 章	ビデオオブジェクトモデル	25
3.1	はじめに	25
3.2	システム要件と関連研究	26
3.2.1	システム要件	26

3.2.2	関連研究	26
3.3	ビデオオブジェクトモデルの提案	26
3.3.1	Video Structure	28
3.3.2	Display Attributes	28
3.3.3	Retrieval Attributes	29
3.3.4	ビデオオブジェクトのメソッド	29
3.4	ビデオオブジェクト管理システムの設計と構築	31
3.4.1	システム構成	31
3.4.2	対象とする動画データの形式	31
3.4.3	Video Object Manager	33
3.5	動画ハイパーメディアシステムの構築	36
3.6	おわりに	39
第 4 章	自動追跡と手動修正を組み合わせたアンカー設定方式	41
4.1	はじめに	41
4.2	関連研究	42
4.3	アンカーモデルの提案	42
4.4	アンカー設定方式の提案	44
4.4.1	アンカー設定方式	44
4.4.2	自動追跡アルゴリズム	46
4.4.3	中間点削除アルゴリズム	47
4.5	アンカー設定ソフトウェアの構築	48
4.6	評価実験	49
4.6.1	評価実験の内容	49
4.6.2	評価方法	50
4.7	評価	52
4.7.1	追跡アルゴリズムの評価	52
4.7.2	中間点削除処理の評価	54
4.8	考察	56
4.9	おわりに	56
第 5 章	拡張イベントモデルによる映像と付加情報の同期配信方式	57
5.1	はじめに	57
5.2	ハイパーリンク機能の定義とシステム要件	58
5.3	関連研究	59
5.4	拡張イベントモデルの設計	60
5.4.1	設計指針	60
5.4.2	データモデルの設計	60
5.4.3	同期配信方式の設計	62
5.5	拡張イベントモデルの実装	64
5.5.1	Windows Media 版の実装	64
5.5.2	MPEG-2 TS 版の実装	65
5.6	評価システムの構築	65

5.6.1	評価システムの構成	65
5.6.2	配信環境	66
5.6.3	再生環境	68
5.7	評価実験	69
5.7.1	評価方針と評価用データ	69
5.7.2	同期配信方式の評価	69
5.8	考察	71
5.9	おわりに	72
第 6 章	初等教育への応用を目的とした動画ハイパーメディアシステム	73
6.1	はじめに	73
6.2	システム要件及び関連研究	74
6.2.1	システム要件	74
6.2.2	関連研究	74
6.3	VisualSHOCK MOVIE の設計及び構築	75
6.3.1	設計指針	75
6.3.2	インタラクティブ動画オブジェクトモデル	75
6.3.3	インタラクティブ動画オブジェクトの出力形式	77
6.3.4	インタラクティブ動画オブジェクトの編集	79
6.3.5	インタラクティブ動画オブジェクトの再生	81
6.4	初等教育への応用及び評価実験	86
6.4.1	コンテンツ利用に関する評価実験	87
6.4.2	コンテンツ制作に関する評価実験	90
6.5	おわりに	92
第 7 章	結論	94

目次

1.1	動画ハイパーメディアシステムの基本コンセプト	2
1.2	動画ハイパーメディアシステムの利用イメージ	2
1.3	本研究における動画ハイパーメディアシステムの構成要素と技術課題	3
2.1	OVID におけるビデオオブジェクトモデル	9
2.2	OVID における Merge 操作	10
2.3	OVID における Overlap 操作	10
2.4	雅におけるアンカーモデル	11
2.5	IntelligentPad におけるアンカーの線形補間の例	13
2.6	CMML の記述例	14
2.7	アファイン変換モデルを用いたオブジェクト追跡法の概要	15
2.8	動画輪郭法 (Snakes) の概要	16
2.9	Cmew におけるアンカー設定方式の概要	17
2.10	Harmony におけるハイパーオブジェクトモデル	18
2.11	SMIL におけるハイパーリンクの記述例	19
2.12	RealMedia のサポートしているイメージマップの例	20
2.13	Cmew におけるリンク情報の例	21
2.14	本研究の位置付け	22
3.1	ビデオオブジェクトモデルの概要	27
3.2	ビデオオブジェクトの持つ <i>compose/decompose</i> メソッド	29
3.3	ビデオオブジェクト管理システムのシステム構成	32
3.4	Video Object Editor の画面イメージ	34
3.5	Video Object Searcher の画面イメージ	35
3.6	動画ハイパーメディアシステムのシステム構成	36
3.7	動画ハイパーメディアシステムにおけるアンカー	37
3.8	動画ハイパーメディアソフトウェアの画面イメージ	38
4.1	アンカーモデル	43
4.2	アンカーの出力例	44
4.3	モデル領域の設定	46
4.4	アンカー設定ソフトウェアの画面イメージ	48
4.5	アンカーの編集	49
4.6	適合度の定義	52
4.7	「テニス 7」の逆方向追跡結果	55
4.8	中間点削除処理の実験結果	55

5.1	付加情報/関連情報データモデル	61
5.2	拡張イベントモデルにおける同期配信の概要	62
5.3	映像と付加情報の同期配信の流れ	63
5.4	評価システムの構成	66
5.5	オーサリングソフトウェアの画面イメージ	67
5.6	再生ソフトウェアの画面イメージ	68
5.7	基本性能評価結果	70
5.8	スポーツ映像に対する評価結果	71
6.1	VisualSHOCK MOVIE のインタラクティブ動画オブジェクトモデル	76
6.2	VisualSHOCK MOVIE のインタラクティブ動画オブジェクトの出力形式例	78
6.3	Movie MapEditor の画面イメージ	80
6.4	VisualSHOCK MOVIE Player の利用イメージ	83
6.5	VisualSHOCK MOVIE Player ActiveX コントロール (a) プロパティ一覧	84
6.6	VisualSHOCK MOVIE Player ActiveX コントロール (b) メソッド一覧	84
6.7	VisualSHOCK MOVIE Player ActiveX コントロール (c) イベント一覧	85
6.8	アンカークリック時のアクション設定	85
6.9	VisualSHOCK MOVIE における RealPlayer の利用イメージ	86
6.10	コンテンツ利用に関する評価実験のシステム概要 [130]	87
6.11	学習コンテンツの画面イメージ	88
6.12	実験 A の結果	88
6.13	実験 B の結果	89
6.14	動画ウォークスルーコンテンツの画面イメージ	90
6.15	実験 C の結果	91
6.16	実験 D の結果	91

表目次

4.1	評価用データ	51
4.2	キャプチャパラメータ	51
4.3	順方向追跡の適合度の低かったビデオクリップ	53
4.4	自動追跡の実験結果	53
4.5	逆方向追跡結果の適合度が順方向追跡結果を下まわったビデオクリップ	54
5.1	Windows Media のイベント情報	64
5.2	MPEG-2 TS におけるイベント情報	65
5.3	評価用データの形式	69
5.4	評価用データの概要	69
5.5	評価用の圧縮パラメータ	70
6.1	VisualSHOCK MOVIE が対応する動画形式	77
6.2	インタラクティブ情報の設定方式	77
6.3	インタラクティブ動画オブジェクトの再生方式	82
6.4	コンテンツ制作実験の結果	92
7.1	本研究における開発ソフトウェアの対応動画形式	96

第1章 序論

1.1 本研究の背景

近年の急激な計算機のハードウェアの性能向上やオペレーティングシステム等のソフトウェア技術の進歩により，計算機上で音声や映像等のマルチメディア情報を扱うことが現実的となってきた．

学校やオフィス等の構内ネットワーク (LAN) でもネットワーク機器の性能向上により，転送速度が 100Mbps ~ 1Gbps のネットワークを安価に構築することが可能となり，マルチメディア情報は従来中心だったローカルのハードディスクや CD-ROM だけでなく，学校やオフィスでの LAN などのネットワーク上での利用も可能となった．

さらに広域ネットワークに目をむけると，FTTH (Fiber To The Home)，ADSL，CATV などの常時接続で高速なブロードバンドインターネット環境の爆発的な普及や，MPEG-2 を利用した BS/110 度 CS/地上波デジタル放送等の普及により，いつでもどこでもマルチメディア情報を利用できる環境が実現されつつある [1]．

このように映像データを中心としたマルチメディア情報は，ローカルのハードディスクや CD-ROM から，LAN，ブロードバンドインターネットやデジタル放送などの様々な環境で使うことが可能となった．

一方，1990 年頃から計算機上でテキスト，画像，動画，音声などのマルチメディア情報を扱うためのフレームワークとしてハイパーメディア技術が注目されている．ハイパーメディアシステムは，テキスト，音声などの情報の単位をノードとして，ノード間の関連をリンクとして表現し，リンクを介してノード間を辿っていくことにより，必要な情報にアクセスすることができる [2-6]．

例えば，普及したハイパーメディアシステムの例である WWW (World Wide Web) の場合，HTML (Hyper Text Markup Language)[7] で記述されたハイパーテキスト中に埋め込まれたハイパーリンクをたどることにより，起点の内容に関連する新たなテキスト，画像，動画，音声などにアクセスすることを可能としている．このハイパーリンクはネットワークを介して自由に記述することが可能であり，インターネットを介して世界中のサーバ上で公開されている情報のリンクによる相互に接続が実現している．

このような背景のもと，本研究では，図 1.1 に示す動画ハイパーメディアシステムの構築技術を対象とする．

動画ハイパーメディアシステムとは，動画中に登場する人や物などのオブジェクトを含む領域に対して関連情報をリンクとして設定しておき，動画の再生中にその領域をマウス等のポインティングデバイスで直接指定することにより，関連情報を直観的に検索することができるシステムである．

図 1.2 に本研究で目的とする動画ハイパーメディアシステムの利用イメージについて示す．

図 1.2 は，WWW 上に動画ハイパーメディアシステムを構築した例であり，Web ブラウザ上の左側のフレームに表示されたスノーモービルのレース映像に登場する選手を含んだ領域に対して設定されたアンカーを表示し，コンテンツ利用者がそのアンカーをクリックすると，右側のフレームに指定した選手のプロフィールを検索し，関連情報として表示する．

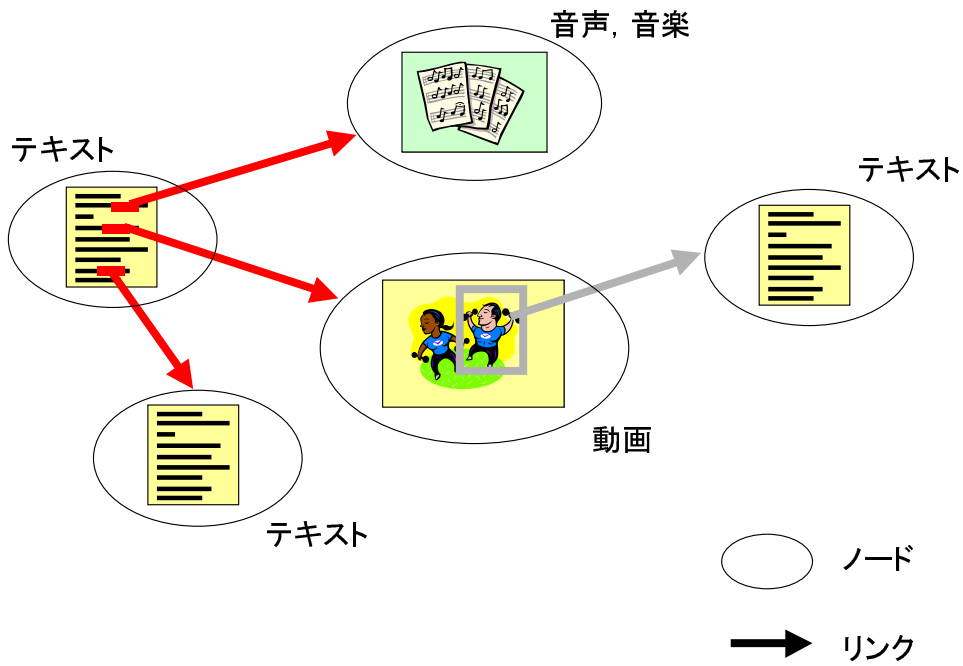


図 1.1: 動画ハイパーメディアシステムの基本コンセプト



図 1.2: 動画ハイパーメディアシステムの利用イメージ

動画ハイパーメディアシステムはインタラクティブな映像メディアを実現するためには比較的単純なシステムであるが、その構築方式が確立していないため、あまり普及していないのが現状である。

1.2 節に、動画ハイパーメディアシステムの構築方式に関する技術課題について整理し、本研究の目的について述べる。

1.2 本研究の目的

本研究の目的は、図 1.1 に示した動画ハイパーメディアシステムの構築方式の確立とその応用についてである。

図 1.3 に本研究における動画ハイパーメディアシステムにおける構成要素とその技術課題について示す。

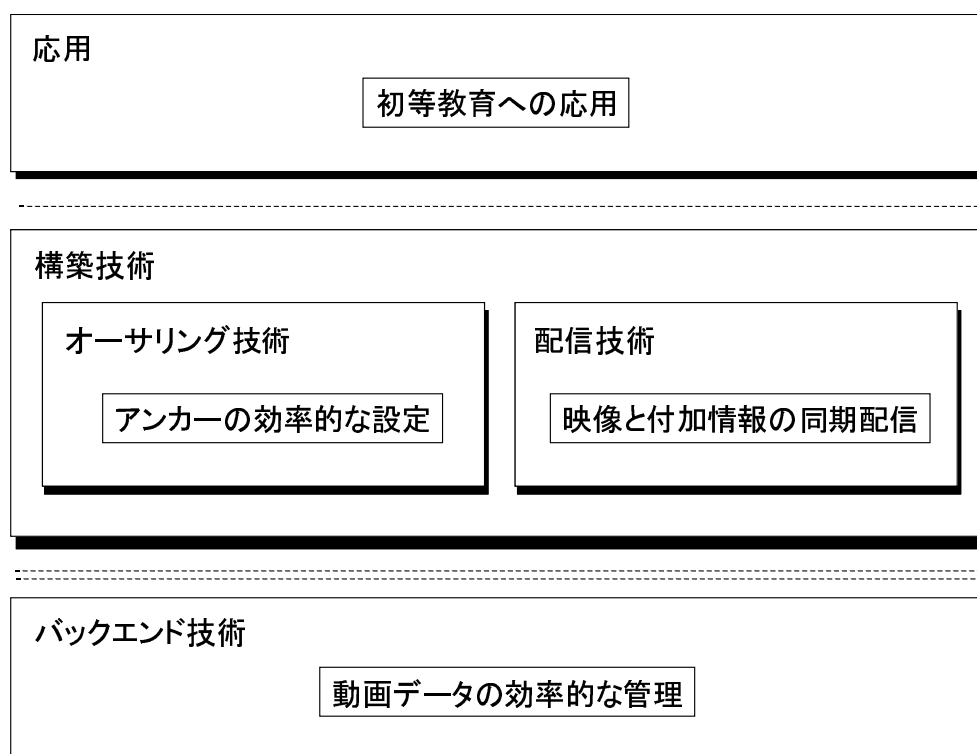


図 1.3: 本研究における動画ハイパーメディアシステムの構成要素と技術課題

1. バックエンド技術:
動画ハイパーメディアシステムのバックエンド技術の技術課題として、動画データを効率的に管理する方式が確立していない、をあげる。
2. オーサリング技術:
動画ハイパーメディアシステムのオーサリング技術の技術課題として、動画データに対してリンクの起点となるアンカーを効率的に設定する方式が確立していない、をあげる。
3. 配信技術:

動画ハイパーメディアシステムの配信技術の技術課題として、映像データとアンカーなどの付加情報を同期して配信する方式が確立していない、をあげる。

4. 応用:

動画ハイパーメディアシステムの応用の技術課題として、初等教育への応用について有効性が確認できていない、をあげる。

1.2.1 以降に、動画ハイパーメディアシステムを構成する構成要素ごとに技術課題について述べ、本研究の目的についてまとめる。

1.2.1 バックエンド技術

図 1.2 に示したバックエンド技術は、動画ハイパーメディアシステムをはじめとする動画応用システムを構築する際の基盤となる技術である。本研究では、動画ハイパーメディアシステムのバックエンド技術における技術課題として、動画データの効率的な管理をあげる。

動画ハイパーメディアシステムなどの動画応用システムを構築する場合、大量の動画、静止画、音声などのマルチメディア情報を効率的に管理する必要があるが、現状では、動画データベースの構築及び検索技術はまだ確立していない [8]。

通常、動画応用システムを構築する場合、様々なソースから動画素材を収集し、それらをデータベースに蓄積し、個々の動画クリップをデータベースから検索することは骨の折れる作業である。その理由としては次の通りである。

1. 動画データの効率的な管理:

インデックス化されていない大量の動画素材から必要なビデオクリップを効率的に管理することは難しい

2. 複数システム間での動画素材の共有:

動画応用システム毎に動画データに対して異なるビューを必要とするため、複数のシステム間で同一の動画素材を共有することは難しい

そこで、本研究では、バックエンド技術において、動画データの効率的な管理方式の確立を目的として、ビデオオブジェクトモデルを提案する。さらに動画応用システムにおける効率的な動画データ管理を目的としてビデオオブジェクト管理システムを設計、構築し、評価システムとして動画ハイパーメディアシステムを構築した。

1.2.2 オーサリング技術

図 1.2 に示したオーサリング技術は、動画ハイパーメディアシステムの構築技術を構成する技術である。本研究では、動画ハイパーメディアシステムのオーサリング技術における技術課題として、アンカーの効率的な設定をあげる。

動画ハイパーメディアシステムのオーサリング時において、動画中に登場するオブジェクトの動きに追従して、リンクの起点となるアンカーの設定を行う必要があるが、現状では効率的なアンカー設定方式が確立していないため、単純に手作業でフレーム毎にオブジェクトの動きにあわせてアンカー設定を行った場合、非常に手間のかかる作業になってしまう。

一方、現状の計算機性能と動画解析技術では、コンテンツ制作者の希望通りにオブジェクトの自動追跡を行うことは技術的に不可能であり、必ず誤りが発生する。そのためシステムとしてアンカーの設定作業の有用性を向上させるには、自動追跡により発生した誤りに手作業で修正する方式を如何に組み合わせるかが重要となる。

そこで、本研究では、オーサリング技術において、アンカーの効率的な設定方式の確立を目的として、自動追跡と手動修正を組み合わせたアンカー設定方式を提案する。提案した方式は動画解析によって登場するオブジェクトの動きを自動的に追跡するとともに、追跡結果を必要に応じて手作業で修正を行う方式であり、次の特徴を持つ。

1. 双方向追跡の適用による追跡効率の向上
2. 中間点削除処理による手作業でのアンカー修正作業の効率化

1.2.3 配信技術

図 1.2 に示した配信技術は、動画ハイパーメディアシステムの構築技術を構成する技術である。本研究では、動画ハイパーメディアシステムの配信技術における技術課題として、映像と付加情報の同期配信をあげる。

前述した通り、映像データを中心としたマルチメディア情報は、従来は PC に接続されたハードディスクや CD/DVD などのローカルでの利用が中心であったが、ネットワーク技術の進展により、LAN、ブロードバンドインターネットなどのネットワークでの利用が可能となった。

特に、ストリーミング技術に基づくブロードバンドインターネットを利用した映像配信サービスは、ネットワークの広帯域化にあわせて、今後急速に普及することが予想されており、その中でも、特にスポーツ映像などのエンターテインメント分野のコンテンツがキラーコンテンツとして着目されている。

例えば、ブロードバンドインターネットでの映像配信サービスに動画ハイパーメディア機構を適用することにより、従来の映像配信サービスでは実現できなかった映像に登場する選手を直接指定して選手のプロフィールや成績などの関連情報を直感的に検索するサービスなどの実現することが可能となる。

映像データのネットワークでの配信に着目した場合、その配信方式として、ダウンロード型配信、オンデマンド型配信、放送型配信など複数の方式が存在するため、アンカーとなる付加情報を映像の形式や配信方式に依存しない形で同期配信する方式を実現する必要がある。

従来の動画ハイパーメディアシステムの研究では、映像の配信方式に依存しない同期方式に着目した研究はほとんどされていない。また、従来のメディア間同期方式に関する研究では、メディア間の同期を行うための同期情報を設定し、その同期情報を用いてメディア間の同期処理を行う研究が多くされているが、本研究で対象とする動画ハイパーメディアシステムを実現するための映像と付加情報の同期配信に適用可能な研究はされていない。

そこで、本研究では、配信技術において、映像と付加情報の同期配信方式の確立を目的として、拡張イベントモデルを提案する。本モデルでは、映像中に対応する付加情報との時間的關係を定義した同期情報を周期的にイベントとして多重化し、その同期情報に基づいて、映像と付加情報の同期配信を実現する方式であり、次の特徴を持つ。

1. 蓄積映像、ライブ映像などの映像の形式やダウンロード型配信、オンデマンド型配信、放送型配信などの配信方式に依存しない同期方式である

2. 開始点同期，実時間同期などの付加情報の使用目的にあわせた同期精度が設定可能である

1.2.4 応用

動画ハイパーメディアシステムの応用例としては，次のようなアプリケーションが考えられる．

1. 教育分野:
動画データを利用した初等中等教育，高等教育，企業内教育用アプリケーション．
2. ビジネス分野:
プレゼンテーションでの利用や，動画マニュアル，製品デモなど．
3. 情報提供分野:
科学館や博物館などに設置された KIOSK 端末上で動作する利用者むけ情報提供用アプリケーション．
4. エンターテインメント分野:
スポーツ映像や映画の予告映像などのエンターテインメント映像用アプリケーション．

このうち，本研究では，動画ハイパーメディアシステムの応用における技術課題として，動画ハイパーメディアシステムの初等教育への応用をあげる．

小中学校等の教育現場では，文部科学省，総務省をはじめとした中央省庁が，全国の小中学校にインターネットへの接続に関する積極的な政策を実施しており，文部科学省の調査では，2004年3月31日現在で，全国の22,913校ある小学校のうち99.7%の22,837校が，10,292校ある中学校のうち99.9%の10,282校が，インターネットに接続されており，WWWなどのインターネット技術を活用した教育に利用されている [9,10] ．

現在の教育の現場では，テレビやビデオをはじめとした映像教材がよく使用されているが，集合教育となりどうしても全員の進捗度をあわせる必要があり，教育効果が上がらないという問題がある．一方，WWWは生徒の自主性や個人の進捗度にあわせて学習が行える利点があり，2002年度より全国の小中学校にて本格的に実施されている総合的な学習の時間でも，積極的に活用されている．

教育現場での利用を目的としたWWWへの動画データの適用について着目した場合，現状のWWWでは，リンク先の情報として動画データを指定し，コンテンツ利用者がそのリンクを呼び出すことによって，リンクされた動画データを再生することは当たり前になってきている．しかし，動画データをリンク元にして，動画データに対するホットスポットをクリックしてリンクされた情報を検索する動画ハイパーメディアシステムはほとんど普及していない．

そこで，本研究では，動画ハイパーメディアシステムの初等教育への応用を目的として，初等教育への応用を目的とした動画ハイパーメディアシステムの設計と構築を行う．構築した動画ハイパーメディアシステムは，初等教育への応用を目的として，次のシステム要件を満たしている．

1. インタラクティブ動画による教育効果向上
2. 編集作業の簡易化
3. Webブラウザ上での利用
4. ストリーミング配信への対応

1.2.5 本研究の目的

本研究の目的について次に示す．

1. 動画データの効率的な管理方式の確立
2. アンカーの効率的な設定方式の確立
3. 映像と付加情報の同期配信方式の確立
4. 動画ハイパーメディアシステムの初等教育への応用

1.3 本論文の構成

本論文では、第2章にて関連研究及び本研究の位置付けについて述べる。

第3章では、動画データの効率的な管理方式の確立を目的として、ビデオオブジェクトモデルとその管理システムについて提案する。さらに動画応用システムにおける効率的な動画データ管理を目的としてビデオオブジェクト管理システムを設計、構築し、評価システムとして動画ハイパーメディアシステムを構築した結果について述べる。

第4章では、アンカーの効率的な設定方式の確立を目的として動画像解析に基づく自動追跡と手動修正を組み合わせた方式を提案する。提案した方式に基づきアンカー設定ソフトウェアを構築し、スポーツ映像を対象としてアンカー設定方式の評価実験を行った結果について述べる。

第5章では、映像と付加情報の同期配信方式の確立を目的として拡張イベントモデルを提案する。提案した拡張イベントモデルを Windows Media 及び MPEG-2 TS を対象としてエンコード/デコードライブラリとして実装し、実装したライブラリを使用して評価システムを構築し、評価システムの実時間同期性能について評価実験を行った結果について述べる。

第6章では、初等教育への応用を目的とした動画ハイパーメディアシステムの設計、構築及び有効性の評価について述べる。システムの設計にあたり、システム要件について設定を行い、設定した要件に基づき、システムの設計及び実装を行い、実装したシステムを初等教育用のマルチメディア教育コンテンツに適用した結果について述べる。

最後に、第7章にてまとめを行う。

第2章 関連研究と本研究の位置付け

2.1 バックエンド技術

本研究では、バックエンド技術の技術課題として、動画データの効率的な管理をあげた。動画データの効率的な管理を実現する動画像データベースの課題としては、動画データの連続性への対応、メディア間の同期、動画データの内容の記述情報の組織化などがある [11]。

バックエンド技術の関連研究として、これらの課題に対応した OVID をあげる。

2.1.1 OVID

神戸大の大本，田中らが開発したビデオデータベースシステム OVID[12-17] では，オブジェクト指向モデルの観点から，ビデオデータのモデル化を行い，Machintosh 上の HyperCard を用いてシステムが構築されている。

次に，OVID のデータモデルの特徴について示す [11]。

- 連続オブジェクトの概念:
意味のあるビデオフレームの部分系列を 1 つの連続的なビデオオブジェクトとしてインクルメンタルに定義できる。各オブジェクトはオーバーラップしてもよい。各オブジェクトはそれぞれ異なる属性構造を有しても構わず，また，各オブジェクトは属性の内容表示やビデオ再生などのメソッドを持つ。
- 複合オブジェクトのサポート:
属性値として単なる文字列だけでなく，ビデオオブジェクト自身も属性値となることができる。すなわち，オブジェクト指向データベースにおける複合オブジェクトの概念を実現している。
- ビデオオブジェクトの合成:
実際のいくつかのビデオオブジェクトの集合体を仮想的なビデオオブジェクトとして定義できる。
- 記述系・検索表示系の一体化:
データベースの内容を表すビデオオブジェクトの記述を行う部分と検索表示を行う部分が一体化されている。例えば，特定のフレームを含むオブジェクトを検索する際，ビデオを再生しその中から必要なフレームを VideoSQL 質問中に取り込むことができる。また，ビデオオブジェクトの逆集約 (分解) 操作により，1 つのビデオオブジェクトを分解し，所望のフレーム (系列) を検索できる。
- チャート型ビジュアルインタフェース:
ビデオオブジェクトをチャート形式で表示し，この上で種々の直接操作が行える (VideoChart)。
- 包含関係に基づく継承機構:
ビデオフレーム系列間の包含関係に基づいて，属性と属性値を継承 (伝播) させる機構を持つ。

ている。

図 2.1 に OVID におけるビデオオブジェクトモデルについて示す [16]。

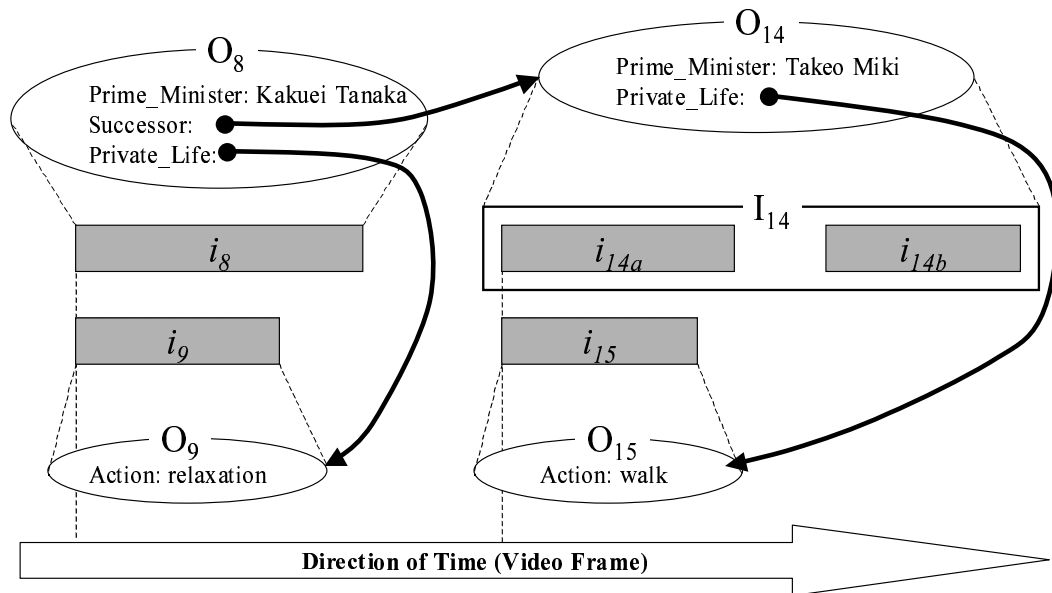


図 2.1: OVID におけるビデオオブジェクトモデル

OVID におけるビデオオブジェクトの構造に対して、次の操作を行うことができる。

1. Interval Projection:
あるビデオオブジェクトの一部を切り出し、新たに新しいオブジェクトとする。
2. Merge:
時間の重なりが無い二つのビデオオブジェクトを結合する。
3. Overlap:
時間の重なりのある二つのビデオオブジェクトの互いに重複する部分を切り出す。

図 2.2 に OVID における Merge 操作、図 2.3 に OVID における Overlap 操作について示す。

OVID における Merge 操作や Overlap 操作は、基本的に同一のビデオクリップに対して定義されたビデオオブジェクトを対象としている。

2.2 オーサリング技術

本研究では、オーサリング技術の技術課題として、アンカーの効率的な設定をあげた。

アンカーの設定方式としては、手動による設定方式、動画像処理を用いた自動設定方式、動画像処理の結果を手動で修正する方式、の3方式が考えられるため、それぞれの方式について関連研究をあげる。

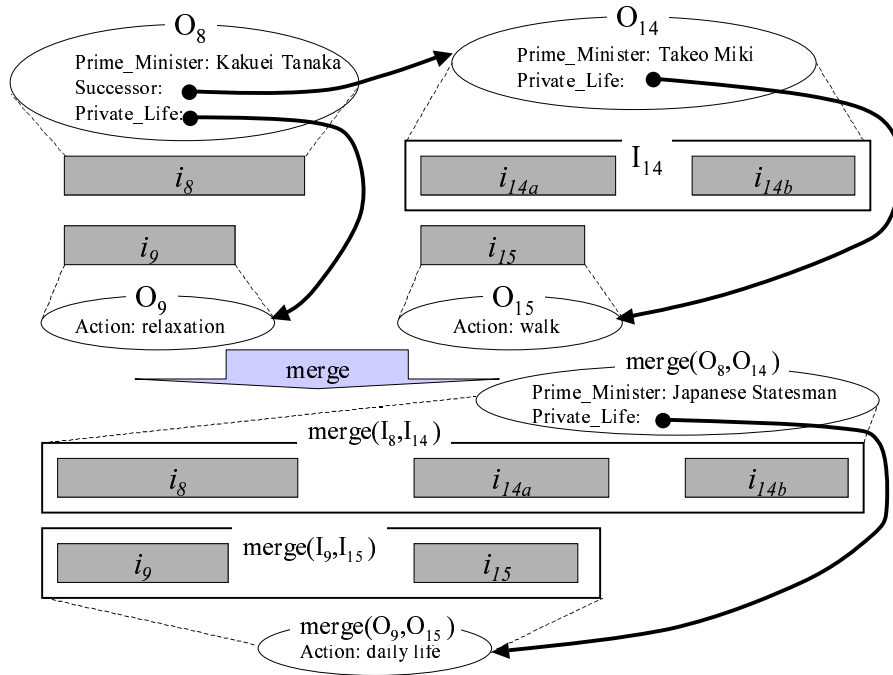


図 2.2: OVID における Merge 操作

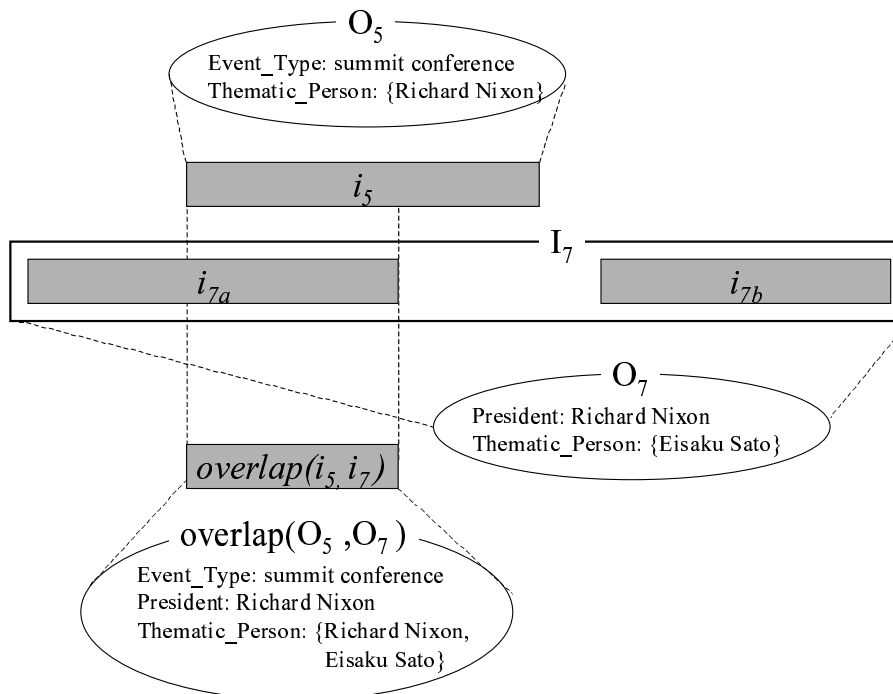


図 2.3: OVID における Overlap 操作

2.2.1 手動による設定方式

手動による設定方式の関連研究として雅，ビデオハイパーメディア，IntelligentPad，CMML をあげる．

雅 NEC の平田，原らの開発したハイパーメディアシステム雅は，従来のハイパーメディアの操作感の良さを生かしつつ DB 概念モデリングの汎用性，拡張性を採用することによる実用規模のシステムへの対応，データベースにおける条件検索や，ハイパーテキスト的なナビゲーション等の多彩なナビゲーション機能の提供，を特徴としている [18,19] ．

雅では，次の 4 種類のナビゲーション方式を提供している [20,21] ．

- ハイパーテキストナビゲーション:
画面上に表示されテキストや画像の特定の部分をクリックすることにより，関連情報を表示する．
- スキーマナビゲーション:
拡張 E-R モデルにより定義されたスキーマ構造の検索とインスタンスの表示ができる．
- 条件検索:
検索条件を指定し，条件を満たすインスタンスの一覧表示を行う．
- 表現 概念ナビゲーション:
表現データ (写真) からその概念データ (説明文) を表示することができる．

雅では，上記ハイパーテキストナビゲーションに関連して，動画中を移動していくオブジェクトをハイパーメディアのノードとしてとして扱うための機構を提案している [22-25] ．

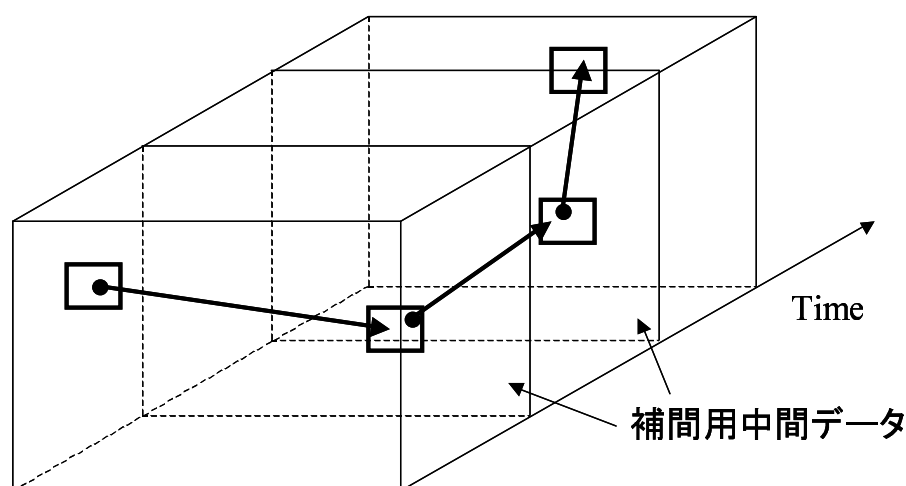


図 2.4: 雅におけるアンカーモデル

これは図 2.4 に示すように，画面を表す二次元及び時間を表す一次元の計三次元からなる時空間状のオブジェクトの軌跡をアンカーとして定義する．雅では，アンカーの形状としては矩形のみをサポートしている．

雅におけるアンカーのオーサリング方式は、提供される動画エディタを使用して、コンテンツ制作者は動画中に移動するオブジェクトのある時刻での領域を指定する。これをオブジェクトの代表的な時刻(フレーム開始・終了時刻, オブジェクトの動きの方向・速さが変化した点等)でオブジェクトを切り出すことができる。コンテンツ制作者が指定した領域と領域の間は、システムが自動的に補間を行う。コンテンツ制作者はすべてのフレームに対して領域を指定する必要が無く、そのオブジェクトが動画中に移動する軌跡の代表点のみを指定すればよい。また、指定したアンカーを直接再生して修正や確認作業を容易に行うことができる。

雅をベースに分散環境に拡張したハイパーメディアオンデマンドシステムである緋においても同様の機構を実現している [26-30]。

ビデオハイパーメディア NTTの坂田らの開発したビデオハイパーメディアはマルチメディアアプリケーション開発環境であり、電子図書館システム、観光案内システム、マルチメディアテストシステム等の映像散策型アプリケーションへの適用を想定している [31-46]。ビデオハイパーメディアでは、動画中に移動するオブジェクトを含む領域をハイパーメディアのアンカーとして扱うことが可能である。

ビデオハイパーメディアのアンカーモデルは雅などと同様のモデルであり、アンカーの形状も矩形のみをサポートしている。

ビデオハイパーメディアの提供する被写体情報エディタを使用して、手動で動画のフレーム内で矩形で対象のオブジェクトを囲み定義する [46]。動画を対象とする際には、位置情報の開始点、中間点、あるいは終了点を指定する。開始点と中間点と終了点のそれぞれの間を線形補間することにより、自動的にアンカーの位置情報を生成する。

IntelligentPad 北大の田中らが開発した IntelligentPad はコンピュータ上で実現できるメディアやツール、プログラムなどをパッドと呼ばれる可視化された部品で表し、パッドを貼りあわせたり機能合成することによって様々なマルチメディアアプリケーションを実現できるシステムである [47,48]。

IntelligentPad には動画ハイパーメディアを実現する HyperMoviePad が提供されており、動画中に登場するオブジェクトを含む領域をハイパーメディアのアンカーとして扱うことが可能である [49,50]。HyperMoviePad を使用することにより、動画に登場するオブジェクトを含む領域に付加されたリンク検索が可能になる。

HyperMoviePad では、雅やビデオハイパーメディアと同様に形状は矩形のみをサポートしている。

HyperMoviePad では、雅やビデオハイパーメディアと同様に、全フレームに対してアンカーの位置や大きさを指定するのではなく、オーサリング時に適当にキーフレームを指定してアンカーの位置、大きさを指定し、キーフレームとキーフレームの間のアンカーの位置と大きさは線形補間によって求める方式となる。

図 2.5 に本システムにおけるアンカーの線形補間の例について示す。

CMML 大阪工大の藤本らが開発した CMML(Clickable Movie Markup Language) は、インタラクティブ動画のモデル化を XML 形式で行い、既存の動画を読み込み動画上に登場するオブジェクトを含む領域に対してアンカーを設定することによりクリックブルムービーを実現することができる [51]。

図 2.6 に CMML の記述例について示す。

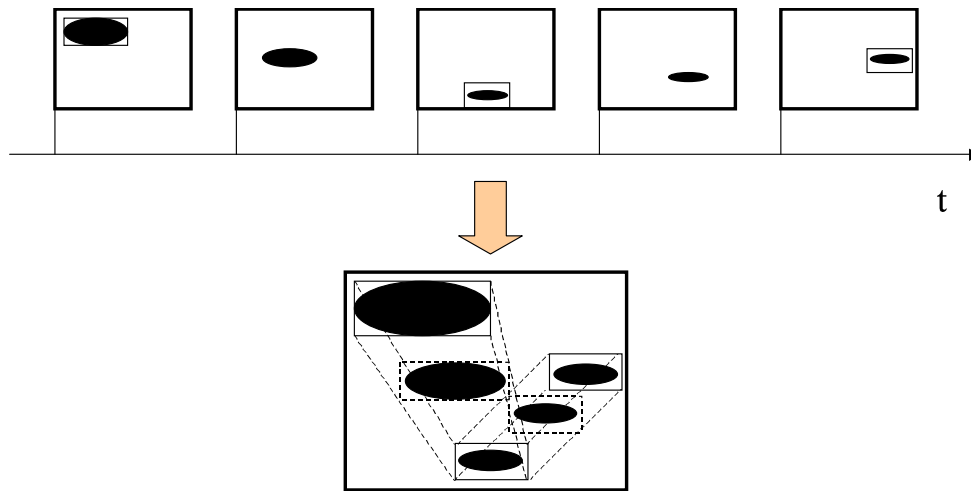


図 2.5: IntelligentPad におけるアンカーの線形補間の例

図 2.6 を見ても分かる通り，CMML のデータモデルは他のシステムと比較して単純であり，提供されたオーサリングツールではアンカーの開始フレームと終了フレームにおける位置と大きさのみ指定することができる．中間のアンカーの位置と大きさは開始フレームと終了フレームのアンカーから線形補間によって求める．

したがって，対象とするオブジェクトが複雑な動きをしている場合は，対応が困難となる．

2.2.2 動画像処理を用いた自動設定方式

動画像処理を用いた自動設定方式の関連研究としてはロボットや自動監視などを目的として数多くの研究がなされているが，ハイパーメディアの制作支援を目的とした研究である，アファイン変換モデルに基づく方式，輪郭モデルに基づく方式，複数モデルの融合による方式，をあげる．

アファイン変換モデルに基づく方式 オブジェクト追跡の最も基本となる方法は，追跡対象のオブジェクト領域をテンプレートとして，次のフレームにおいてテンプレートマッチングを繰り返す方法である．この方法はシンプルであるが，オブジェクトの変形や回転が大きい場合など追跡に失敗する場合が考えられる．

そこで，オブジェクトの変形や回転に対応して，オブジェクト領域の局所的な動きを元にオブジェクト全体の動きや変形を推定するアファイン変換モデルが提案されている [52,53]．アファイン変換モデルにおける，オブジェクトの局所的な動きの検出方式として，こう配法 [54,55] とブロックマッチング法 [56,57] がある．

東芝の金子らの開発したオブジェクト追跡方法は，追跡物体の変形モデルとしてアファイン変換モデルを仮定し，オブジェクト内部の局所的な動きの検出方式としてオブジェクト内部に信頼度の高いブロックを追跡ブロックとして選出し，その追跡ブロックのブロックマッチングの結果からアファイン変換の記述パラメータをロバスト推定により推定する方法である [58,59]．この方式の特徴としては，追跡物体の変形に追従が可能な点と，ノイズにロバストな点があげられる．

```

<cmml>
<moviefile href="Globe.avi"><title>CMML サンプル1</title></moviefile>
<clickable>
<start>
<region>
<left>208</left><top>100</top><right>248</right><bottom>178</bottom>
</region>
<region>
<left>248</left><top>95</top><right>267</right><bottom>172</bottom>
</region>
<time>
<hour>0</hour><min>0</min><sec>1</sec><msec>972</msec>
</time>
</start>
<end>
<region>
<left>63</left><top>109</top><right>101</right><bottom>175</bottom>
</region>
<region>
<left>101</left><top>124</top><right>175</right><bottom>227</bottom>
</region>
<time>
<hour>0</hour><min>0</min><sec>3</sec><msec>452</msec>
</time>
</end>
<info href="bird.bmp">複雑な領域</info>
</clickable>
</cmml>

```

図 2.6: CMML の記述例

図 2.7 に本モデルを用いたオブジェクト追跡方式について示す。

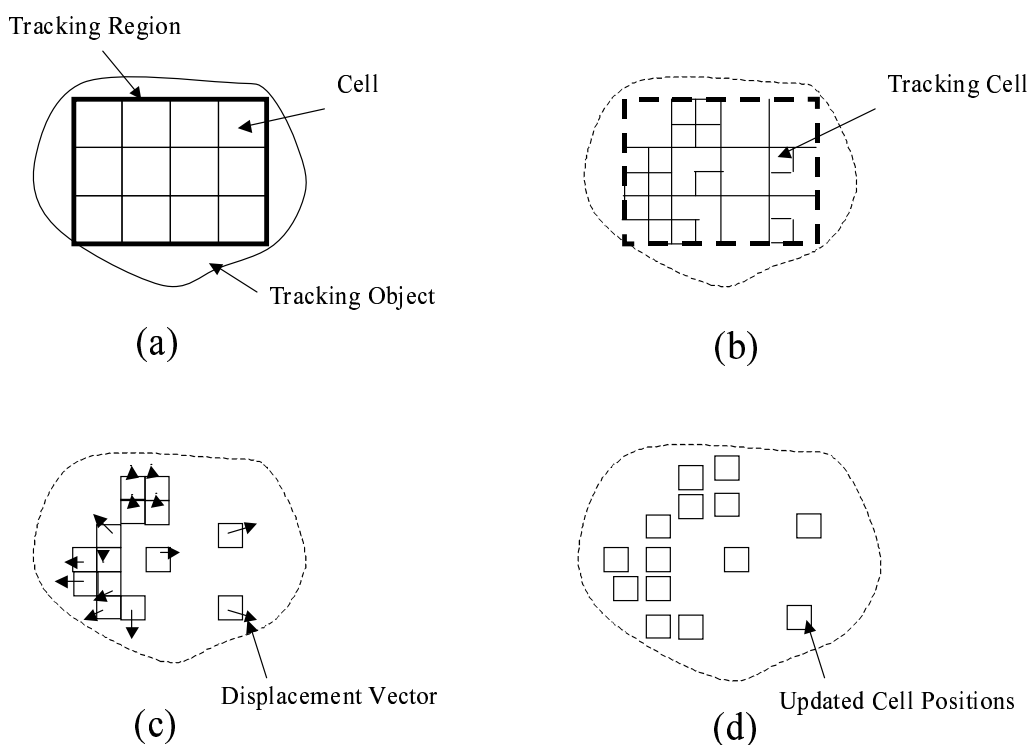


図 2.7: アファイン変換モデルを用いたオブジェクト追跡法の概要

図 2.7 において，(a) にてコンテンツ制作者の手入力にて，追跡したいオブジェクトを指定すると，システムは自動的に対象領域をレベル 0 のセルに分解する．さらに (b) にて各セルを更に下位レベルのセルに分解し，ブロックマッチングの信頼性に基づき，実際に追跡を行うセルの選択を行う．(c) にて階層化ブロックマッチングを行い，移動ベクトル及び信頼度を元にアファイン変換のパラメータを推定する．最後に (d) でアファインパラメータ推定の信頼度が閾値よりも高ければ，パラメータに基づいて追跡セルの位置を更新するとともに，次フレームでの追跡セルの初期値として設定する．

輪郭モデルに基づく方式 動的輪郭法 (Snakes) は，対象となるオブジェクトの周りに配置した輪郭曲線を定義したエネルギー関数を最小化するように時間的に発展させ，最終的にオブジェクトの輪郭に収束させる手法である [60] ．

図 2.8 に動画輪郭法 (Snakes) の概要について図示する．

動画輪郭法 (Snakes) は，明確な輪郭を持つオブジェクトの輪郭を抽出する方式であるが，慶應大の岡田らは複数のオブジェクトなどの明確な輪郭を持たない領域オブジェクトの追跡モデル Amoeba の研究を行っている [61] ．この研究では，明確な輪郭を持たない複数のオブジェクト領域の自動追跡，また領域オブジェクトが分裂した場合の自動検出を目的としたものである．

複数情報の融合による方式 NHK の三須らはデジタル放送の番組上に登場する映像オブジェクトをリモコンやマウスなどでクリックすると，そのオブジェクトに関連する情報を提示するデータ放送

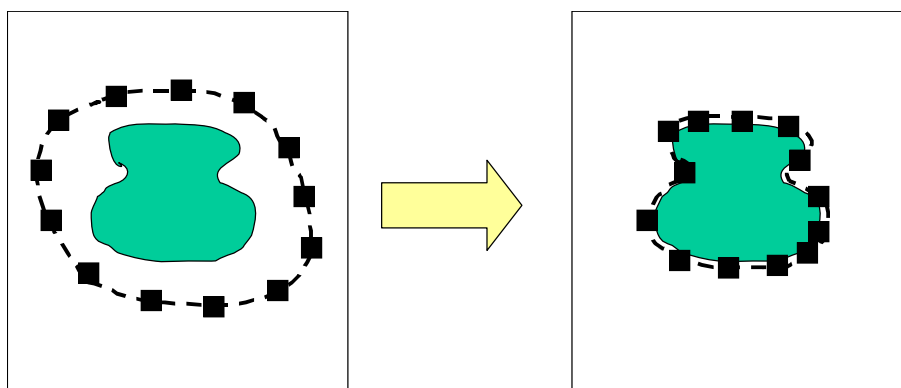


図 2.8: 動画輪郭法 (Snakes) の概要

サービスである情報ナビゲーション型未来放送の研究を行っている [62-77] .

その一環として、映像データに登場するオブジェクトの位置や大きさを編集するオーサリングソフトウェアにて、複数情報の融合によるオブジェクトの自動追跡処理について研究している [63] .

本方法は統合層、追跡層、特徴検出層の3階層にて構成され、最下位の特徴検出層にて、複数種類の特徴検出器 (FD: Feature Detector) を設置する。提供されるFDのとしては、(1) オブジェクト全体のテクスチャに基づく位置、大きさの検出、(2) オブジェクトの一部(頭部など)のテクスチャに基づく位置、大きさの検出、(3) 局所的テクスチャに基づく動きベクトルの検出、(4) 色に基づく位置・大きさの検出、(5) オブジェクトの平均色、共分散に基づく位置、大きさの検出、の5種類である。追跡層では、シーンに含まれる映像オブジェクトと同数の追跡器を置き、各追跡器には Kalman フィルタを置き、特徴検出層の検出結果の信頼度やオブジェクト間の位置関係に応じて適応的に、情報の融合を行うとともに、動きモデルに基づく動き予測を行う。

最上位の統合層では、各オブジェクトの追跡結果を統合して、ラベリングを行うとともに、オクルージョンなどの検出結果を下位層にフィードバックする。

2.2.3 自動設定結果を手動で修正する方式

自動設定結果を手動で修正する方式の関連研究として Cmew をあげる。

Cmew NTTの高田らの開発した Cmew(Continuous Media with the Web) は、連続メディアデータ(MPEG-1ストリーム)にリンク情報をあわせることにより、WWW上で動画情報を起点とした情報ナビゲートを可能とするフレームワークである [78-95]。Cmewを用いると、動画像データに新たな可能性を付加することができ、例えば、Netscape Navigatorと協調する Cmew 対応の MPEG-1 プレーヤーでドラマやコマーシャルビデオに登場する人物や物をクリックすることにより、俳優の情報へのナビゲート、登場人物の洋服やコマーシャル商品のショッピングサイトへのナビゲート、また意見や投書欄へのアクセスなどが可能になる。

Cmewの特徴としては、MPEG-1/SYSTEMストリームのプライベート2パケットにフレーム単位にリンク情報を埋め込むことによって動画データを起点としたリンク機構を実現している [81]。

また、高田らは Cmew への適応を想定したリンク情報のマークアップエディタの研究を行ってい

る [83] . この研究では、動画像解析を適用したオブジェクトの自動追跡を特徴としている . 図 2.9 に Cmew におけるアンカーの設定方式の概要について示す .

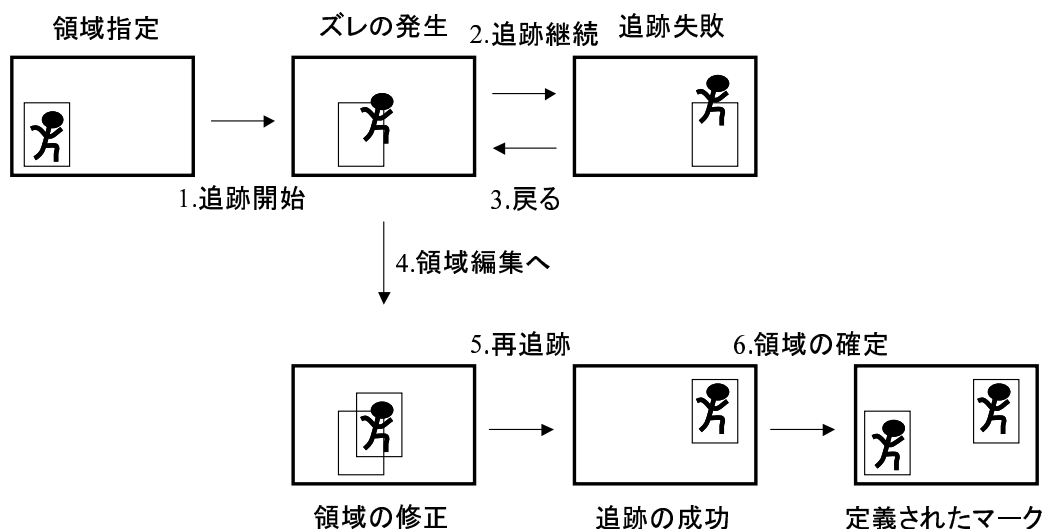


図 2.9: Cmew におけるアンカー設定方式の概要

図 2.9 を見ても分かるように、この研究におけるオブジェクト追跡方式では、あるフレームにおいて追跡したいオブジェクトの領域を指定すると、後は動画像解析により自動的に順方向に追跡処理を行う。自動追跡が失敗した場合はそのフレームに戻り、手作業により位置を修正した後で再度順方向に追跡を行う方式である。

現状の計算機性能と動画像解析技術では、コンテンツ制作者の希望通りに自動追跡を行うことは技術的に不可能であり、必ず誤りが発生する。そのためシステムとして設定作業の有用性を向上させるには、自動追跡により発生した誤りを手作業で修正する方式を如何に組み合わせるかが重要となってくる。高田らのアプローチでは、追跡誤りが発生する度に追跡処理が中断し手作業で修正が必要が出てくるため、対象とする動画データの種類のよっては自動追跡処理中に何度も追跡処理が中断する可能性がある。

2.3 配信技術

本研究では、配信技術の技術課題として、映像と付加情報の同期配信をあげた。関連研究として、Harmony, SMIL, RealMedia, QuickTime, Name-It をあげる。

2.3.1 Harmony

大阪大の藤川、下條らが開発したハイパーメディアシステム Harmony では、ハイパーテキストモデルとオブジェクト指向の概念を組み合わせたハイパーオブジェクトモデルを提案している [96] .

図 2.10 に Harmony におけるハイパーオブジェクトモデルについて示す。

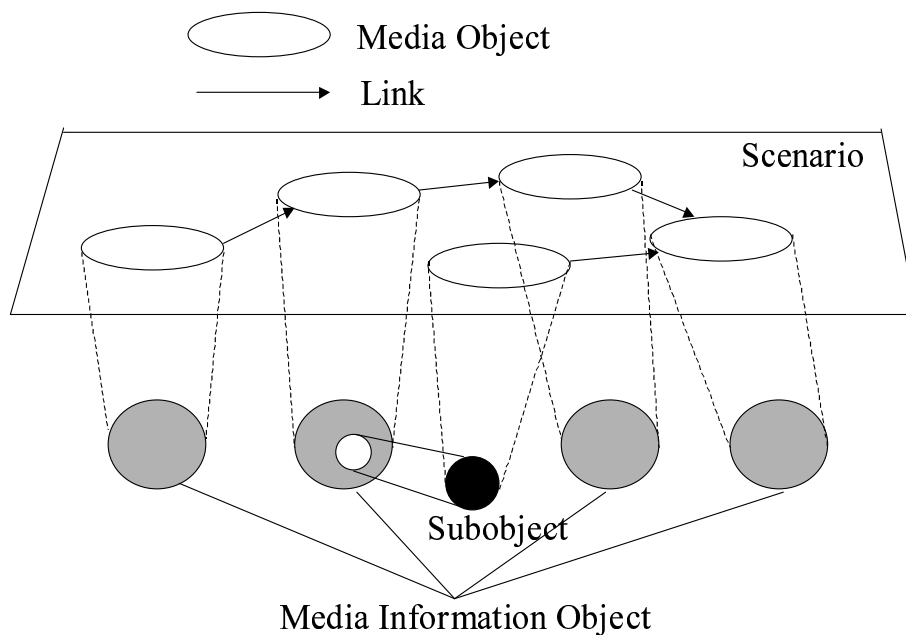


図 2.10: Harmony におけるハイパーオブジェクトモデル

Harmony では、映像、アニメーション、図形、音楽、テキストの 5 種類のメディアがメディアインフォメーションオブジェクトとして定義できる。ここでは、Harmony における映像オブジェクトの概要について説明する。Harmony における映像情報はすべて追記型の光ディスクに蓄積されており、一つの映像オブジェクトは、光ディスク上の任意の連続したシーンを表している。映像オブジェクトは、開始フレームと終了フレームによって表される。

Harmony では、映像情報に現れる人物や物体をサブオブジェクトとしてリンクの始点や宛先に指定することができる。実際には、それらの人物や物体を包含するような長方形を指定することで、映像情報のサブオブジェクトを表している。映像サブオブジェクトの人物や物体の動きや大きさの変化に対する対応については言及されていない。

また、Harmony では、LAN 上に接続された複数のワークステーション間上での動作を前提として、次の同期機構を導入している [97]。

- **開始点同期:**
複数のマルチメディア情報を同期して同時刻に表示するための同期機構。あらかじめプレゼンテーションのシナリオを解析し、プレゼンテーションタイムテーブルに個々のオブジェクトの表示開始時刻を計算しておくことで対応する。
- **実時間同期:**
30 秒のシナリオを 30 秒で再生するための同期機構。ペースマネージャにより、逐次再生状況を確認することで対応する。
- **適応型連続同期:**
実時間同期とは異なり、アニメーションなどの実時間性を優先して内容を間引かずに、再生を遅らせてでも完全に内容を表示する際に使用する同期機構。この場合は、関連する他のメディ

アに進行状況を通知する必要がある。Harmony では、プレゼンテーションマネージャを導入し、各ペースマネージャからのイベントに基づき制御を行う。

また、分散型マルチメディアプレゼンテーションシステム Symphony では、ネットワーク環境の変化への柔軟な対応を目的としている [98,99]。

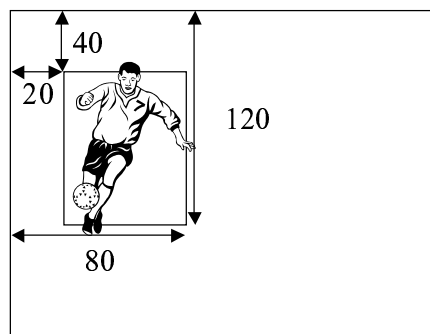
2.3.2 SMIL

W3C(World Wide Web Consortium)にて標準化されている SMIL(Synchronized Multimedia Integration Language)[100] は、音声、ビデオ、テキスト、画像、アニメーションなどのマルチメディア情報を、時間的に同期させて表示するための言語で、SMIL を用いることにより、ファイルの指定や表示位置、再生方法などの記述が可能となっている。SMIL は RealPlayer(RealNetworks) や、QuickTime Player(Apple Computer) でサポートされている。

SMIL では、映像データ中に矩形、円、またはポリゴン形式のホットスポットを記述し、URL 等のハイパーリンクを設定することができる。

図 2.11 に SMIL におけるハイパーリンクの記述例について示す。

```
<video src="video1.rm" region="video_region">  
<area href="http://www.real.com/"           ← クリック時のアクション  
      begin="0s" end="30s"                   ← 開始, 終了時間  
      shape="rect" coords="20, 40, 80, 120" /> ← 表示位置  
</video>
```



shape="rect" coords="20, 40, 80, 120"

図 2.11: SMIL におけるハイパーリンクの記述例

図 2.11 を見れば分かる通り、SMIL におけるハイパーリンクは、表示位置とその開始・終了時間を設定するモデルである。このため、定義区間でのアンカー領域は静止したままである。また、スクリプト中で明示的に時間情報を記述する形式であり、主にオンデマンド型の配信を対象とした方式となる。

2.3.3 RealMedia

RealNetworks の開発したストリーミング映像配信プラットフォームである RealSystem で扱われている RealMedia では、映像中にイベントやイメージマップを多重化して配信することが可能である [101] .

RealMedia がサポートしているイベントについて次に示す .

1. title:
イベント発生により、RealMedia データのタイトルを通知する .
2. author:
イベント発生により、RealMedia データの制作者情報を通知する .
3. copyright:
イベント発生により、RealMedia データの著作権情報を通知する .
4. url:
イベント発生により、指定された URL のハイパーリンクを発行する .

また、RealMedia では、SMIL と同様に、イメージマップをサポートしており、動画データ中にハイパーリンク機能を持つホットスポットを定義することができる . RealMedia のサポートしているイメージマップの例について図 2.12 に示す .



図 2.12: RealMedia のサポートしているイメージマップの例

RealMedia に対してイベントやイメージマップを多重化するには、RMEvents ユーティリティまたは RealProducer が必要である . どちらもエンコード時のリアルタイムでの多重化はサポートしていないため、一旦 RealMedia データを作成した後で、これらのツールを用いて多重化することになる .

2.3.4 QuickTime

Apple Computer の開発したストリーミング映像配信プラットフォームである QuickTime では、動画データのハイパーリンク機能を実現するための機構としてスプライトトラックをサポートしている [102,103] .

スプライトとはアニメーションで、ロゴなどの画像データや映像データをあらかじめ指定したベクタにあわせて動作させる機能を持つ . スプライトは、画像の準備、動きの定義、アクションの設定に

よって定義され、基本的には再生時にビューワ側で位置や大きさが計算されて表示される。

ただ、現状の QuickTime では、ストライプトラックはストリーミング配信には対応しておらず、ダウンロード配信にのみの対応となる。

2.3.5 Name-it

広島市立大の有川らは遠隔地に設置された定点観測カメラの映像をハイパーメディア化する空間ハイパーメディアシステム Name-it を開発した [104,105]。

Name-it では、ズーム、パン、チルトなどのカメラの撮影情報とハイパーメディアで使用するラベルを結びつける仕組みを提供することにより、映像データに対するハイパーメディア機能を実現している。

2.4 応用

動画ハイパーメディアの応用の関連研究として前述の Cmew をあげる。

2.4.1 Cmew

前述の Cmew(Continuous Media with the Web) は、連続メディアデータ (MPEG-1 ストリーム) にリンク情報をあわせることにより、WWW 上で動画情報を起点とした情報ナビゲートを可能とするフレームワークである [78-95]。

Cmew を用いることにより、動画像データに新たな可能性を付加することができ、例えば、Netscape Navigator と協調する Cmew 対応の MPEG-1 プレーヤーでドラマやコマーシャルビデオに登場する人物や物をクリックすることにより、俳優の情報へのナビゲート、登場人物の洋服やコマーシャル商品のショッピングサイトへのナビゲート、また意見や投書欄へのアクセスなどが可能になる。

図 2.13 に Cmew におけるリンク情報の例について示す [81]。

#	1.	2.	3.	4.	5.
rect	http://www.keio.ac.jp/aaa/	1912-1922	-	97, 153, 159, 178	
rect	http://www.keio.ac.jp/bbb/	1912-1930	-	215, 143, 267, 174	
circle	http://www.keio.ac.jp/ccc/	1940-1950	-	43, 31, 62, 32	
poly	http://www.keio.ac.jp/ddd/	400-451	-	165, 71, 75, 89, 88, 129, 232, 136, 245, 96	
rect	http://www.keio.ac.jp/eee/	999-1200	auto	95, 105, 254, 123	

図 2.13: Cmew におけるリンク情報の例

図 2.13 に示したように、Cmew では、アンカーの形状としては、矩形 (rect)、円 (circle)、多角形 (poly) の 3 種類が指定可能であり、リンク情報の各項目としては、

1. 領域種別
2. リンク先 URL

3. リンクが存在するフレーム区間 (時間的領域)
4. 動作
5. 画面中の位置を示す座標

となる。また、動作が“auto”指定された場合、そのリンクが存在するフレームを再生する際に自動的に Web ブラウザでリンク先の情報を表示する。

現状、Cmew はストリーミング再生には対応しておらず、メディアデータのダウンロードが完了してから、再生が開始される仕様となっている。

2.5 本研究の位置付け

本研究の位置付けについて図 2.14 に整理する。

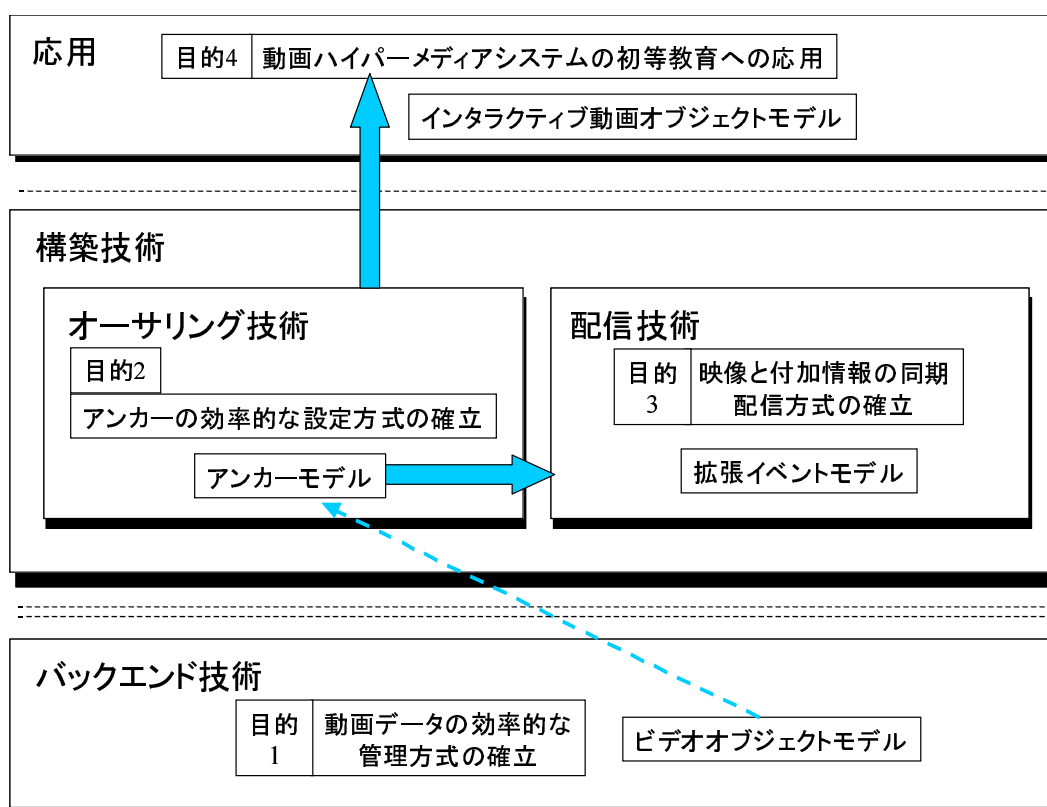


図 2.14: 本研究の位置付け

以下、各技術要素毎に本研究の目的に対する関連研究との比較を行い、本研究の位置付けについて説明を行う。

2.5.1 バックエンド技術

本研究におけるバックエンド技術に関する目的は、動画データの効率的な管理方式の確立であり、その必要条件として、次の要件を設定する。

1. 動画素材の柔軟な蓄積方式を持ち、構造化及びアノテーションにより動画データの効率的な管理を図る
2. 動画データの物理的な形式と独立して、個々の動画応用システム毎に論理的なビューを持ち、複数の動画応用システム間での動画データの共有を図る

関連研究でとりあげた OVID[12] は、映像データからのビデオクリップの検索の効率化を目的としたシステムであり、複数の動画応用システム間での動画データの共有については考慮されておらず、本研究では上記要件を満たす方式の確立を基本方針とする。

2.5.2 オーサリング技術

本研究におけるオーサリング技術に関する目的は、アンカーの効率的な設定方式の確立である。

関連研究を分析した結果、手動方式は非常に手間がかかり、自動方式は現状の動画像解析では、100% 自動化は難しいという結果となった。そこで、自動方式と手動方式を組み合わせることが重要であるが、関連研究でとりあげた Cmew[81] では、追跡誤りが発生する度に追跡処理が中断し手作業で修正する必要が出てくるため、対象とする動画データの種類のよっては自動追跡処理中に何度も追跡処理が中断してしまう可能性がある。

これらの検討結果を元に、以下の2項目を基本方針として設定する。

1. 動画像解析を応用したオブジェクト領域の自動追跡の性能向上
2. 自動追跡の結果がコンテンツ制作者の意図に反した場合を想定した手作業での修正作業を効率化

2.5.3 配信技術

本研究における配信技術に関する目的は、映像と付加情報の同期配信技術の確立であり、そのシステム要件としては、次の要件を設定する。

1. 映像の形式や配信方式に依存しない同期方式であること
2. 付加情報の使用目的にあわせた同期精度が設定可能なこと

関連研究でとりあげた動画ハイパーメディアシステムおよびネットワーク環境での複数メディア間の同期方式では上記要件を満たすものは無く、本研究では上記要件を満たす方式の確立を基本方針とする。

2.5.4 応用

本研究における応用に関する目的は、動画ハイパーメディアの初等教育への応用であり、そのシステム要件としては、次の要件を設定する。

1. インタラクティブ動画による教育効果向上
2. 編集作業の簡易化
3. WWW ブラウザでの利用
4. ストリーミング配信への対応

関連研究でとりあげた動画ハイパーメディアシステムである VHM[46] , IntelligentPad[49] , CMML[51] では、専用のプレーヤーを必要とし WWW ブラウザ上での利用が考慮されていない、Cmew[81] ではストリーミング配信に対応していない、など上記要件を満たすものは無く、本研究では上記要件を満たす方式の確立を基本方針とする。

第3章 ビデオオブジェクトモデル

第3章では、動画ハイパーメディアシステムのバックエンド技術として、動画応用システムを構築する際に基盤となる動画データの効率的な管理方式の確立を目的として、ビデオオブジェクトモデルを提案する。さらに、ビデオオブジェクトの管理機能を持つビデオオブジェクト管理システムの設計、構築を行い、評価システムとして動画ハイパーメディアシステムを構築した結果について述べる。

3.1 はじめに

近年、動画、静止画、音声などのマルチメディア情報を様々な動画応用システムにおいて利用したいという要求が高まっている [106]。しかしながら、現状では、これらのマルチメディア情報を効率的に管理できる動画データベースの構築及び検索技術はまだ確立していない [8]。

例えば、動画ハイパーメディアシステムをはじめとする動画応用システムを構築する場合、様々なソースから動画素材を収集し、それらをデータベースに蓄積し、必要なビデオクリップをデータベースから検索し、利用することは非常に骨の折れる作業である。その理由としては、次の課題が考えられる。

1. 動画データの効率的な管理:
インデックス化されていない大量の動画素材から必要なビデオクリップを効率的に管理することは難しい
2. 複数システム間での動画素材の共有:
動画応用システム毎に動画データに対して異なるビューを必要とするため、複数のシステム間で同一の動画素材を共有することは難しい

本研究では、上記の課題に対処するために、ビデオオブジェクトモデルを提案し、動画応用システムの構築を目的としたビデオオブジェクト管理システムについて述べる。

本システムの特徴は次の2項目である。

1. 動画素材の柔軟な蓄積方式を持ち、構造化及びアノテーションに基づいたモデルを提供し、動画データの効率的な管理を行う
2. 動画データの物理的な形式と独立して、個々の動画応用システム毎に必要な論理的なビューを持ち、複数の動画応用システム間で動画データを共有する

以下、3.2でビデオオブジェクト管理システムのシステム要件と関連研究について述べる。3.3でビデオオブジェクトモデルの提案について述べ、3.4でビデオオブジェクト管理システムの設計と構築について述べる。3.5で本システムの評価システムとして構築した動画ハイパーメディアシステムについて述べ、最後に3.6でまとめを行う。

3.2 システム要件と関連研究

本研究で対象とするビデオオブジェクト管理システムのシステム要件及び関連研究について述べる。

3.2.1 システム要件

ビデオオブジェクト管理システムの必要条件として以下の項目を設定した。

1. 動画素材の柔軟な蓄積方式を持ち，構造化及びアノテーションにより動画データの効率的な管理を図る
2. 動画データの物理的な形式と独立して，個々の動画応用システム毎に論理的なビューを持ち，複数の動画応用システム間での動画データの共有を図る

上記必要条件を満たすためのシステム要件として以下の項目を設定した。

1. 動画データの物理形式に関係無く，動画素材の任意の部分をビデオクリップとして定義できること
2. 定義したビデオクリップ同士を組み合わせることで新しいビデオクリップとして定義できること
3. ビデオクリップに対して設定したアノテーションによる属性検索だけでなく，ビデオクリップの内容によるナビゲーション検索を提供できること
4. すべての動画応用システムから，その物理的な蓄積形式に関係無く同じインタフェースを介して利用できること
5. ビデオクリップに対して大きさや位置などの表示属性を独自のビューとして定義して使用できること

3.2.2 関連研究

Mackay が開発したビデオアノータ EVA[107,108] では，動画応用システム開発者に対して動画データに対するアノテーション機能を提供する。EVA は動画データの解析には便利であるが，アノテーションが付加されたビデオシーン同士を組み合わせたり，分解したりすることはできない。

大本と田中が開発した OVID[16] は，独自のモデルに基づくビデオオブジェクトデータベースシステムである。OVID で定義されているビデオオブジェクトモデルでは，以下の特徴を持つ。

1. ビデオオブジェクト間でアノテーションを共有する “interval-inclusion based inheritance” 機構
2. 複合したビデオオブジェクトに対するメソッドを提供

OVID は，映像データからのビデオクリップの検索の効率化を目的としたシステムであり，本システムの目的の一つである動画応用システムの構築に必要な要件は満たさない。

3.3 ビデオオブジェクトモデルの提案

3.2 で設定したシステム要件に基づいて，動画応用システムを構築する際の動画データの効率的な管理方式の確立を目的として，オブジェクト指向アプローチに基づく形でビデオオブジェクトモデルの設計を行った。

ビデオオブジェクトモデルの設計指針について次に示す。

1. 動画データの任意の部分を仮想的なビデオクリップとして定義する
2. *compose/decompose* メソッドを提供し、複数のオブジェクトのビデオ構造を組み合わせたリ、分解する
3. 属性検索だけでなくビデオクリップの内容によるナビゲーション検索を提供する
4. オブジェクト指向モデルに基づいてモデル化し、すべての動画応用システムから共通的に利用できるインタフェースを提供する
5. ビデオ構造の定義と分離させて定義したアノテーションの種類を使用目的に応じて表示属性と検索属性に分類する

図 3.1 にビデオオブジェクトモデルの概要について示す。

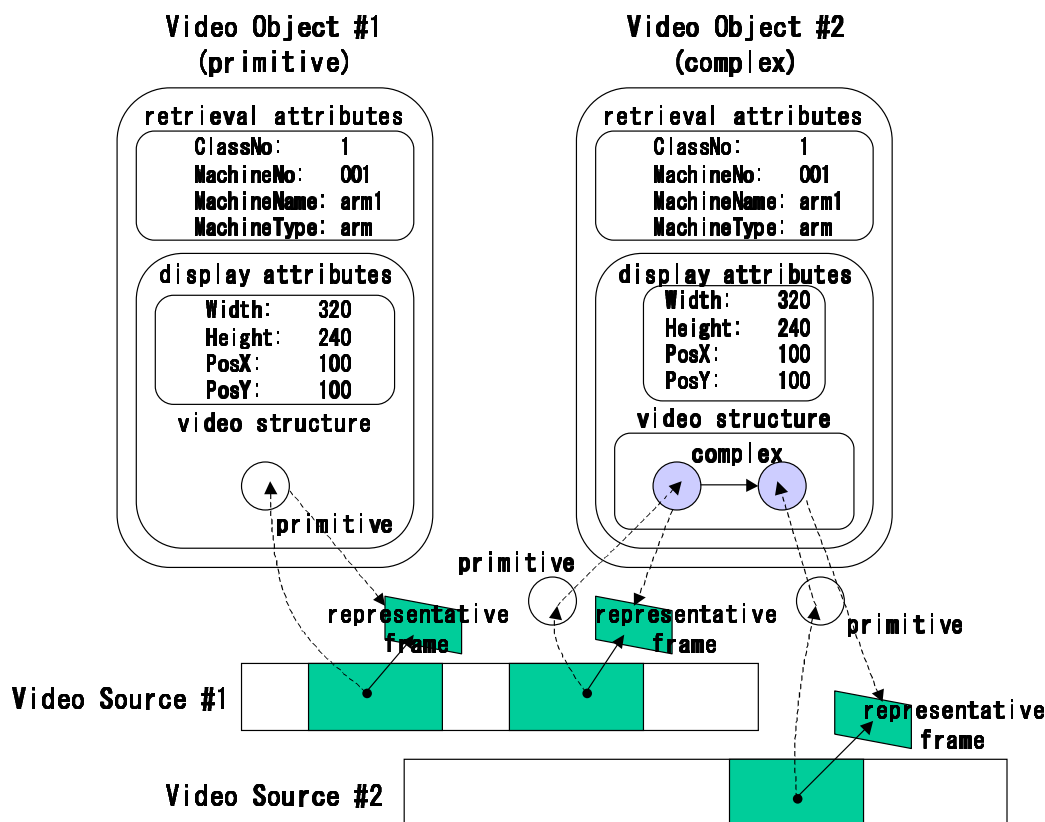


図 3.1: ビデオオブジェクトモデルの概要

図 3.1 に示したとおり、ビデオオブジェクトは、Video Structure、Display Attributes、Retrieval Attributes より構成される。

図 3.1 において Video Source は加工されていない動画データであり、任意のフレームレート、大きさ、圧縮形式で蓄積されている動画データである。ビデオオブジェクトはこの Video Source から任意の区間を切り出した形で Video Structure を定義し、アノテーションとして Display Attributes や Retrieval Attributes を付加したモデルである。

3.3.1 Video Structure

Video Structure はオブジェクト指向モデルによりビデオクリップの構造を定義し，次の 2 種類のオブジェクトのどちらかで構成される．

primitive オブジェクトは Video Structure の基本となるオブジェクトであり，Video Source から適切な部分を切り出して定義したオブジェクトであり，以下の要素から構成される．

1. Object_ID (オブジェクト識別子)
2. Object_Name (オブジェクト名)
3. Video_Source_ID (Video Source 識別子)
4. Start_Frame_No (開始フレーム番号)
5. End_Frame_No (終了フレーム番号)
6. Representative_Frame_ID (代表フレーム識別子)

代表フレーム識別子は primitive オブジェクトごとにそのオブジェクトを代表するフレームの識別子を蓄積する．primitive オブジェクト中に代表フレームの識別子を持つことにより，外部から primitive オブジェクトの内容を代表するフレームの内容をサムネイル形式の静止画として表示させることが可能となるため，ビデオクリップのナビゲーション検索が可能となる．

続いて，complex オブジェクトについて説明する．

complex オブジェクトは，複数の primitive または complex オブジェクトを再生順に並べることによって構成される複合オブジェクトで，以下の要素から構成される．

1. Object_ID (オブジェクト識別子)
2. Object_Name (オブジェクト名)
3. Elements (オブジェクト要素)

complex オブジェクトを用いることにより，異なる動画形式で蓄積された複数の Video Source から定義された複数の primitive オブジェクトを組み合わせる新しいビデオオブジェクトとして利用することができる．

3.3.2 Display Attributes

Display Attributes はビデオオブジェクトを外部から利用する際の表示情報として利用するための属性で，Video Structure に対するアノテーションとして定義する．Display Attributes は，その構成要素として，表示位置 (X,Y) や表示サイズ (width, height) などの属性を持つ．

1. Object_ID (オブジェクト識別子)
2. Object_Name (オブジェクト名)
3. Video_Structure_ID (Video Structure オブジェクト識別子)
4. Position (表示位置 (X, Y))
5. Size (表示サイズ (width, height))

Display Attributes を使用することによって，ビデオオブジェクト単位に，利用する動画応用システム毎に大きさや位置などの表示属性を独自のビューとして定義することが可能となる．

3.3.3 Retrieval Attributes

Retrieval Attributes はビデオオブジェクトの属性検索を目的とした属性で，Video Structure に対するアノテーションとして定義する．Retrieval Attributes は，利用する動画応用システムの目的に応じて自由に定義して利用することが可能である．この属性を使用することにより，ビデオオブジェクトの属性検索が可能となる．

3.3.4 ビデオオブジェクトのメソッド

ビデオオブジェクトはオブジェクト指向モデルに基づいて設計されており，全ての動画応用システムに対して共通のインターフェースを提供する．ビデオオブジェクトのメソッドについて次に定義する．

構造に関するメソッド

compose/decompose メソッドは，ビデオオブジェクトの構造に関するメソッドとして定義する．*compose/decompose* メソッドの概要について図 3.2 に示す．

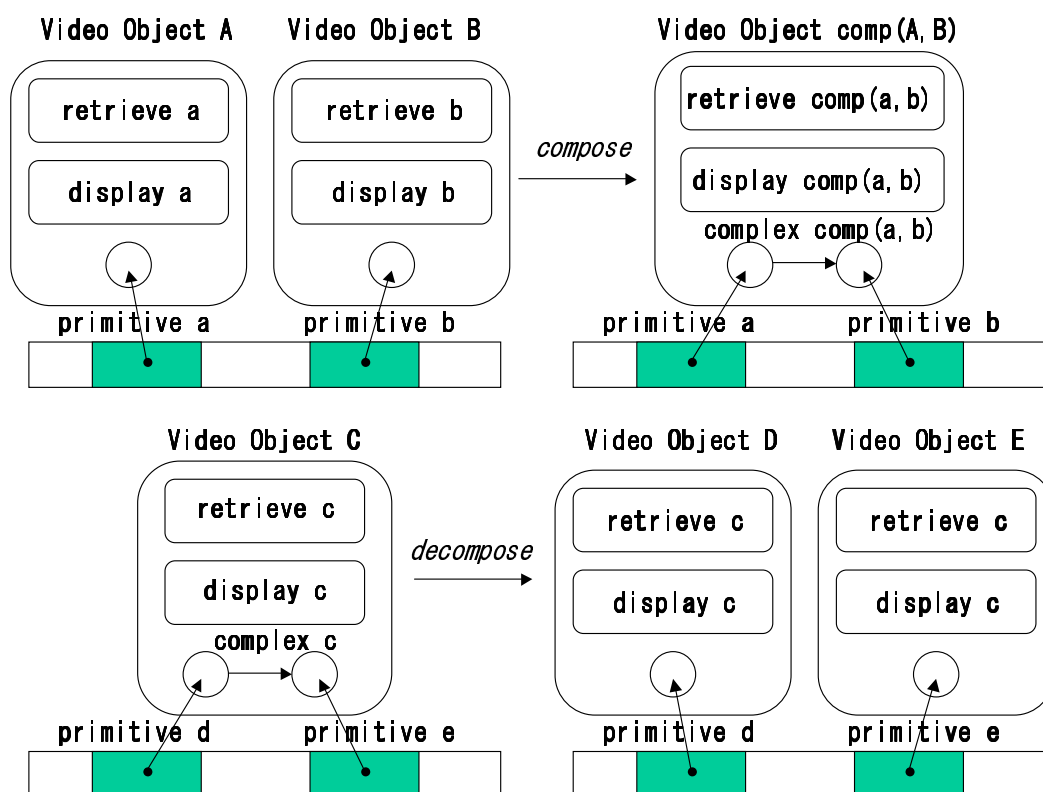


図 3.2: ビデオオブジェクトの持つ *compose/decompose* メソッド

次に，*compose* メソッド，*decompose* メソッドの概要について示す．

compose 複数のビデオオブジェクトの Video Structure 部分を複合して、新しいビデオオブジェクトとして定義するメソッドである。

新しいオブジェクトは複合する前のビデオオブジェクトの Display Attributes や Retrieval Attributes などのアノテーションを継承することができ、新規に定義することも可能である。継承を選択した場合は、アノテーションを継承したいオブジェクトを選択する必要がある。

decompose Video Structure 部分を分解して新しいビデオオブジェクトを作成するメソッドである。新しいオブジェクトは分解前のビデオオブジェクトに対するアノテーションを継承することも、新規に定義することもできる。

次に、*decompose* メソッドの動作について説明する。

1. complex オブジェクトの Video Structure 部分を primitive(または complex) オブジェクトに分解する
2. primitive(または complex) オブジェクトの Video Structure 部分から新しいビデオオブジェクトを作成する

アノテーションに関するメソッド

ビデオオブジェクトのアノテーションに関するメソッドについて以下に説明する。

set 指定したビデオオブジェクトのアノテーションの属性値を設定するメソッドである。このメソッドを用いてビデオオブジェクトにアノテーションである Display Attributes や Retrieval Attributes の属性値を設定する。

get 指定したビデオオブジェクトのアノテーションの属性値を取得するメソッドである。このメソッドを用いてビデオオブジェクトにアノテーションである Display Attributes や Retrieval Attributes の属性値を取得する。

get_rep_frame 指定したビデオオブジェクトを構成する primitive オブジェクトの代表フレームのフレーム識別子を取得するメソッドである。指定したオブジェクトの Video Structure が primitive オブジェクトの場合は 1 個、complex オブジェクトの場合は複数のフレーム識別子を取得する。

このメソッドは、指定したビデオオブジェクトの内容をコンテンツ制作者に伝えることを目的としてビデオオブジェクトの視覚的なナビゲーションに使用される。

制御に関するメソッド

ビデオオブジェクトの制御に関するメソッドについて以下に説明する。

open 指定したビデオオブジェクトを読み込むメソッドである。ビデオオブジェクトの再生時に使うメソッドであり、再生ソフトウェアなどからこのメソッドを呼び出すと、ビデオオブジェクトを読み込み、画面上に Display Attributes の属性値に従って指定したビデオオブジェクトの開始フレームを表示する。

play 指定したビデオオブジェクトを再生するメソッドである。 *open* メソッドで読み込まれたオブジェクトに対して、このメソッドを呼び出すことにより、再生処理を行う。

stop 指定したビデオオブジェクトの再生を停止するメソッドである。 *play* メソッドにより再生中のオブジェクトに対して、このメソッドを呼び出すことにより、ビデオオブジェクトの再生を停止し、表示位置をビデオオブジェクトの開始フレームに戻す。

pause 指定したビデオオブジェクトの再生をその場で一旦停止するメソッドである。 *play* メソッドにより再生中のオブジェクトに対して、このメソッドを呼び出すことにより、ビデオオブジェクトの再生をその場で一旦停止する。

close 指定したビデオオブジェクトの再生処理を終了するメソッドである。このメソッドを呼び出すとビデオオブジェクトの再生処理を終了する。

ビデオオブジェクトの再生時の流れ 次に、ビデオオブジェクトの再生処理の流れについて説明する。

1. *open* メソッドにより、ビデオオブジェクトを読み込む
2. *play* メソッドにより、再生処理を行う
3. *pause* メソッドにより、再生処理を一旦停止する
4. *stop* メソッドにより、再生処理を停止し、表示位置を開始フレームに戻す
5. *close* メソッドにより、ビデオの再生処理を終了する

3.4 ビデオオブジェクト管理システムの設計と構築

3.4 では、3.3 で定義したビデオオブジェクトの管理機能を持つビデオオブジェクト管理システムの設計と構築について述べる。

3.4.1 システム構成

ビデオオブジェクト管理システムのシステム構成について述べる [109,110]。

図 3.3 にビデオオブジェクト管理システムのシステム構成について示す。

本システムでは、Video Storage において、キャプチャ、編集されたビデオデータを Video Source として蓄積する。Video Object Database では、ビデオオブジェクトの編集結果を蓄積する。Video Object Manager は、Video Object Database に蓄積されたビデオオブジェクトを管理する。

3.4.2 対象とする動画データの形式

ビデオオブジェクト管理システムで対象とする動画データの形式について次の通り定義する。

- AVI
- QuickTime Movie
- MPEG-1

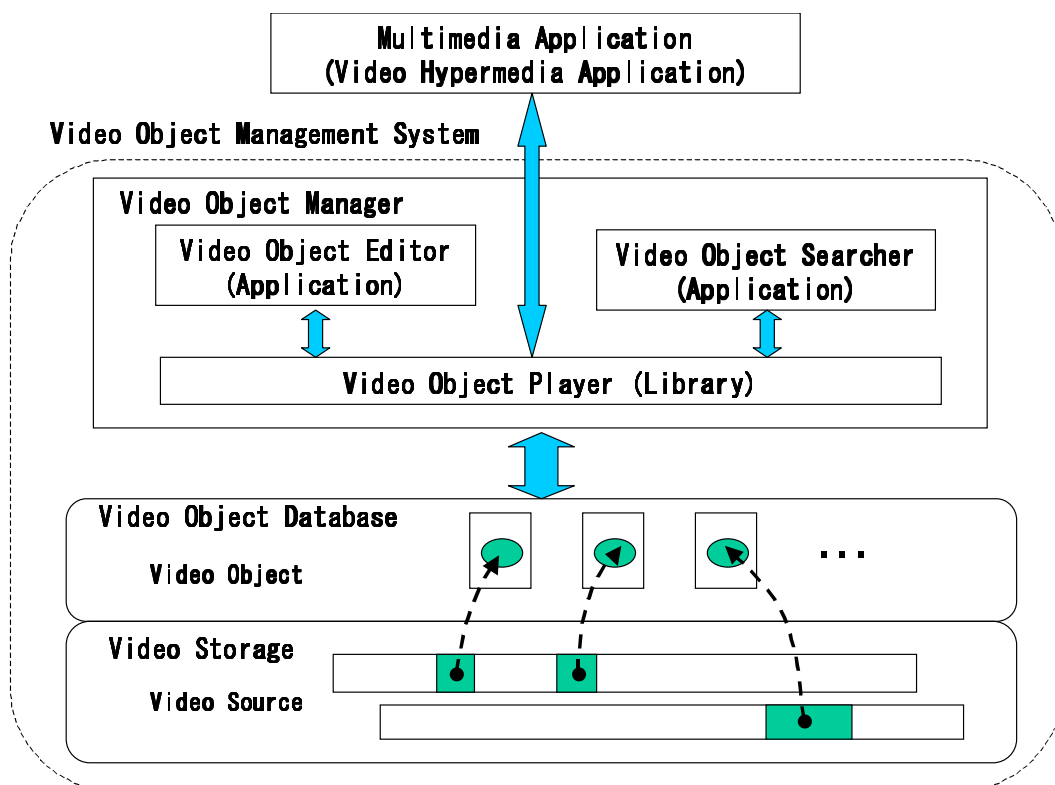


図 3.3: ビデオオブジェクト管理システムのシステム構成

3.4.3 Video Object Manager

Video Object Manager はビデオオブジェクトの管理を目的としたソフトウェアであり、次の3つのソフトウェアより構成される。

1. Video Object Editor:
Video Storage に蓄積された Video Source を参照して新規にビデオオブジェクトを定義し Video Object Database に蓄積するとともに、蓄積したビデオオブジェクトの編集機能を持つ。
2. Video Object Searcher:
Video Object Database に蓄積されたビデオオブジェクトの検索機能を持つ。
3. Video Object Player:
Video Object Database に蓄積されたビデオオブジェクトの再生機能を持つ。

Video Object Editor

Video Object Editor はビデオオブジェクトの定義、編集機能を持つアプリケーションソフトウェアである。

Video Object Editor は、Video Storage に蓄積された Video Source を参照して新規にビデオオブジェクトを定義し Video Object Database に蓄積する機能及び、Video Object Database に蓄積したビデオオブジェクトを編集する機能を持つ。また、後述する Video Object Player を呼び出して、編集したビデオオブジェクトを再生して内容を確認する機能を持つ。

Video Object Editor を用いることにより、ビデオオブジェクトの Video Structure 及びそのアノテーションである Display Attributes 及び Retrieval Attributes の編集を行うことができる。

構築した Video Object Editor では、AVI、QuickTime Movie、MPEG-1 形式の動画データをサポートしている。Video Object Editor にて、ビデオオブジェクトの新規定義を行う際は、動画データの開始フレームと終了フレームを指定する必要があるが、通常、この作業は、Video Object Player を使用して動画データの内容を確認しながら行う。フレーム間圧縮のかかった MPEG-1 形式の映像データを表示する場合、Video Object Player からは GOP (Group Of Pictures) 単位でのアクセスとなり、GOP 単位で開始フレームと終了フレームを指定する仕様となる。一方、AVI、QuickTime Movie の場合は、フレーム単位でのアクセスが可能であり、開始フレーム、終了フレームの指定はフレーム単位となる。

図 3.4 に構築した Video Object Editor の画面イメージについて示す。

Video Object Editor は Microsoft Windows 3.1 上で動作し、Microsoft Excel 5.0 のマクロプログラムとして実現されている。

Video Object Searcher

Video Object Searcher はビデオオブジェクトの検索機能を持つアプリケーションソフトウェアである。

Video Object Searcher のビデオオブジェクト検索機能は、通常の属性検索に加えて映像の内容に基づくナビゲーション検索を組み合わせた方式を提供する。

1. Retrieval Attributes を用いた属性検索:
ビデオオブジェクトの検索のための属性とその属性値を設定すると、対象とする Video Object

	A	B	C	D	E	F	G	H
1								
2	0003	フットボール	prin0003		0	555	444	444
3	0004	ボリス・クレストとスノーボード	con0003		1	377	377	377
4	0005	ロケットとフットボール	con0004		0	333	377	77
5	0005	ロケットとボリス・クレスト	con0005		0	350	400	100
6	0007	宇宙への旅立ち (ロケット)	prin0004		1	500	555	20
7	0009	フットボール	prin0005		0	555	500	220
8	0005	ボリス・クレスト4	prin0006		0	500	600	410
9	0010	宇宙への旅立ち (ロケット)	prin0001		0	350	350	100
10	0011	ロケットとスノーボード	con0001		1	450	500	100
11	0012	フットボール	prin0002		0	350	350	325
12	0013	フットボールとスノーボード	con0002		1	350	350	325
13	0014	宇宙への旅立ち (ロケット)	prin0004		0	555	600	170
14								
15								
16								

図 3.4: Video Object Editor の画面イメージ

Database に蓄積されたビデオオブジェクトの Retrieval Attributes から条件に一致するものを属性検索し、その結果を一覧表示する。

2. *get_rep_frame* メソッドによる代表フレームのナビゲーション検索:
属性検索の結果、複数のビデオオブジェクトが候補として見つかった場合、各ビデオオブジェクトに対して *get_rep_frame* メソッドを呼び出し、代表フレームを取得し、その結果を一覧表示する。画面上に表示された代表フレームの一覧からビデオオブジェクトの内容を確認することによりビデオオブジェクトのナビゲーション検索を行う。
3. 動画オブジェクトの再生:
検索したビデオオブジェクトは後述する Video Object Player を呼び出して、実際に再生して内容の最終確認を行う。

図 3.5 に構築した Video Object Searcher の画面イメージについて示す。

Video Object Searcher は Microsoft Windows 3.1 上で動作し、Microsoft Visual Basic 及び Microsoft Visual C/C++ で実装されている。

Video Object Player

Video Object Player はビデオオブジェクトの再生機能を持つライブラリソフトウェアであり、対象とする動画応用システムから呼び出すことにより、ビデオオブジェクトの再生機能を提供する。Video Object Player は Windows の Dynamic Link Library (DLL) として実装されている。

また、Video Object Editor や Video Object Searcher からビデオオブジェクトの内容確認を目的として、呼び出して使用することも可能である。

Video Object Player を使用したビデオオブジェクトの再生手順について示す。

1. Video Object Player に、*open* メソッドを送る。



図 3.5: Video Object Searcher の画面イメージ

2. Video Object Player は，ビデオオブジェクトの Display Attributes の内容に従って動画データを表示する．
3. Video Object Player に，*play* メソッドを送る．
4. Video Object Player は，ビデオオブジェクトの Video Structure の内容に従って動画データを再生する．
5. Video Object Player に，*close* メソッドを送る．

Video Object Player は Microsoft Windows 3.1 で動作し，Microsoft Visual C/C++ で実装されている．

3.5 動画ハイパーメディアシステムの構築

構築した動画オブジェクト管理システムの評価システムとして構築した動画ハイパーメディアシステムについて述べる．

図 3.6 に，構築した動画ハイパーメディアシステムのシステム構成について述べる．

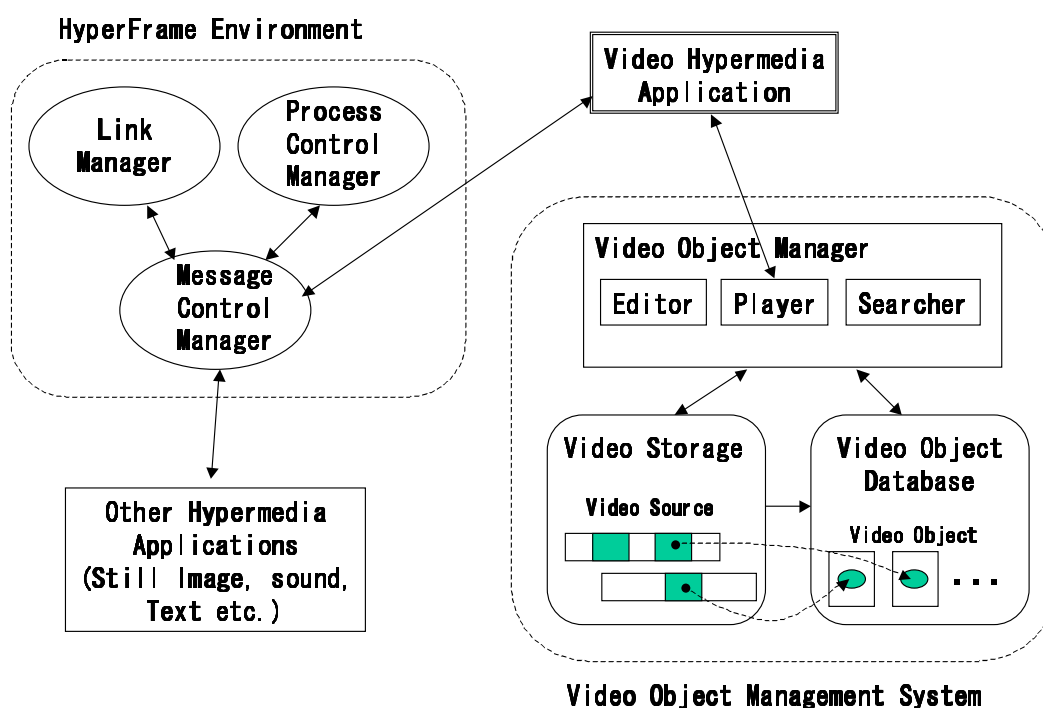


図 3.6: 動画ハイパーメディアシステムのシステム構成

動画ハイパーメディアシステムの構築にあたり，ベースとするハイパーメディア環境として，筆者の開発したハイパーメディア環境である HyperFrame[111,112] を選択した．HyperFrame は，複数のアプリケーションソフトウェア間を統合するためのハイパーメディアフレームワークである．

本研究では，HyperFrame に対応した動画ハイパーメディアソフトウェアを開発し，HyperFrame を用いて他の HyperFrame に対応したハイパーメディアソフトウェアとのリンクを実現することによ

り、動画ハイパーメディアシステムを構築した。

また、開発した動画ハイパーメディアソフトウェアの動画データ管理機能として、前述のビデオオブジェクト管理システムを適用することにより、動画データの効率的な管理を図った。

動画ハイパーメディアソフトウェアからビデオオブジェクトの再生を行う場合、前述の Video Object Player を呼び出すことにより、ビデオオブジェクトの再生を行うとともに、ビデオオブジェクトに対してリンクの起点となるアンカーを設定し、HyperFrame を介して他のハイパーメディアソフトウェアに対するリンクを設定する。動画ハイパーメディアソフトウェアでは、再生中のビデオオブジェクトの primitive オブジェクトの代表フレームの識別子を HyperFrame に送ることによってリンク情報を一意に管理している。

図 3.7 に構築した動画ハイパーメディアシステムにおけるアンカーの概要について示す。

通常、動画データに登場するオブジェクトの動きにあわせて、リンクの起点となるアンカーを設定する場合、オブジェクトの動きにあわせて全フレームにおいてアンカーの位置を編集する必要があり、複雑なアンカー設定作業が必要になってしまう。

そこで、アンカー設定作業を簡易化を目的として、代表フレームを用いた方式* を提案する。ビデオオブジェクトの持つ *get_rep_frame* メソッドを利用して、代表フレームをビデオオブジェクトの primitive オブジェクトの内容を最も表すフレームとして取得し、代表フレームに登場するオブジェクトを含む領域に対して位置と大きさを設定することにより、ハイパーメディアのアンカーとして利用する方針とした。

代表フレームに対して設定したアンカーは位置、大きさが不変のアンカーとして管理され、その有効区間は対応する primitive オブジェクトの設定区間と同等とした。

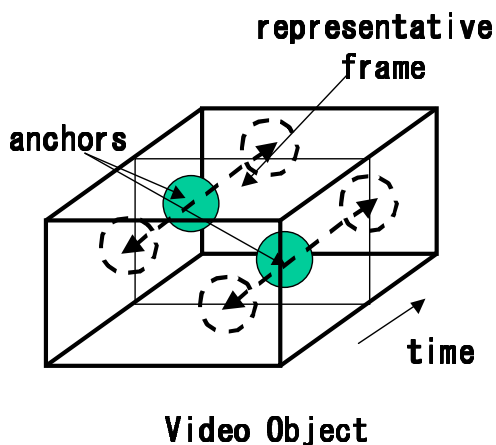


図 3.7: 動画ハイパーメディアシステムにおけるアンカー

図 3.8 に動画ハイパーメディアシステムにおける動画ハイパーメディアソフトウェアの画面イメージについて示す。

動画ハイパーメディアソフトウェアは編集モードと検索モードの2つのモードを持っており、図 3.8 は編集モードの例を示す。編集モードでは、ビデオオブジェクトの再生を一時停止すると、ビデオオ

* 第 4 章にてアンカー設定作業の効率化について検討する。

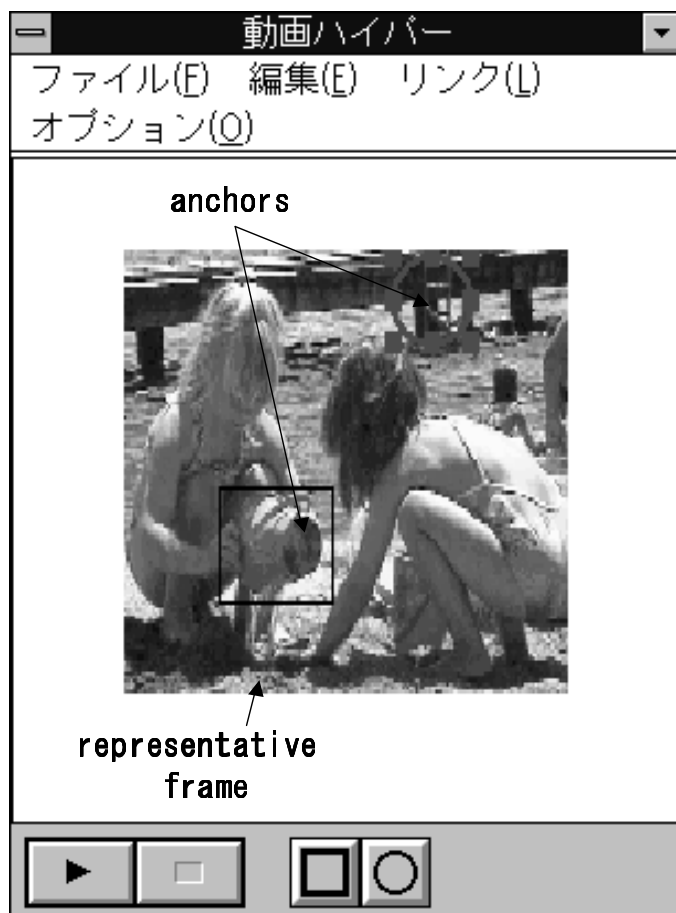


図 3.8: 動画ハイパーメディアソフトウェアの画面イメージ

プロジェクトのその表示位置に応じた代表フレームを表示する．表示された代表フレーム上にアンカーを描くことにより，リンクを設定することができる．

一方，検索モードでは，ビデオクリップの再生を一時停止しても代表フレームを表示せず，現在のフレームを表示したまま再生を一時停止する．ビデオオブジェクトの表示画面上でカーソルをダブルクリックしその領域が代表フレームに設定したアンカーの領域内にある場合，HyperFrame を介してリンク情報を検索し，関連情報の表示を行う．

動画ハイパーメディアソフトウェアは，Microsoft Windows 3.1 で動作し，Microsoft Visual Basic と Microsoft Visual C/C++ で実装されている．

3.6 おわりに

第3章では，動画ハイパーメディアシステムのバックエンド技術として，動画応用システムを構築する際に基盤となる動画データの効率的な管理方式の確立を目的として，ビデオオブジェクトモデルの提案を行った．さらに，ビデオオブジェクト管理システムの設計，構築を行い，評価システムとして動画ハイパーメディアシステムを構築した．

まず，ビデオオブジェクトモデル及びビデオオブジェクト管理システムの検討にあたって，次の項目をシステム要件として設定した．

1. 動画データの物理形式に関係無く，動画素材の任意の部分をビデオクリップとして定義できること
2. 定義したビデオクリップ同士を組み合わせることで新しいビデオクリップとして定義できること
3. ビデオクリップに対して設定したアノテーションによる属性検索だけでなく，ビデオクリップの内容によるナビゲーション検索を提供できること
4. すべての動画応用システムから，その物理的な蓄積形式に関係無く同じインタフェースを介して利用できること
5. ビデオクリップに対して大きさや位置などの表示属性を独自のビューとして定義して使用できること

上記の要件に基づき，ビデオオブジェクトモデルの提案及び，ビデオオブジェクト管理システムの設計，構築を行い，評価用システムとして動画ハイパーメディアシステムの構築を行うとともに，設定したシステム要件に基づいて，次の評価を行う．

1. 動画素材の任意の部分をビデオクリップとして定義可能:
動画素材の任意の区間を切り出して primitive オブジェクトとして新たに定義できる機構を提供した．primitive オブジェクトは，動画素材のポインタのみを持っている仮想的なオブジェクトで動画素材自体は編集の対象では無いので，直接動画素材を編集する方式と比較して，編集作業の効率化が可能となる．
2. ビデオクリップ同士を組み合わせることが可能:
複数のビデオオブジェクトの Video Structure 部分を組み合わせる新しいビデオオブジェクトを作成する *compose* メソッドを提供し，既存のオブジェクトを組み合わせることにより編集作業の効率化が可能となる．

3. ビデオクリップの内容によるナビゲーション検索を提供:
ビデオクリップの代表フレームの識別子を返す *get_rep_frame* メソッドを用いて、ビデオクリップの一覧表示を行う際にあわせて代表フレームを表示することにより、ビデオオブジェクトの検索効率の向上を図る。
4. すべての動画応用システムから同じインタフェースを介して利用可能:
オブジェクト指向アプローチに基づいてビデオオブジェクトの設計を行うことにより、すべての動画応用システムから同一のインタフェースを介して動画データにアクセスする機構を実現できた。
5. ビデオクリップに表示属性を独自のビューとして定義可能:
ビデオオブジェクトモデルの設計を行う際に、アノテーションとして表示属性と検索属性を分離して設定できる機構を提供した。

今後の課題としては、次の点がある。

構築したビデオオブジェクト管理システムを動画応用システムから利用する場合、現状では、Video Object Player を経由したビデオオブジェクトの再生利用インタフェースのみ提供されている。今後、ビデオオブジェクトのエクスポート機能を実現することで、ビデオオブジェクトの編集結果を取り出して動画応用システムの入力データとして使用することが可能となり、適用範囲が広がることが予想される。

第4章 自動追跡と手動修正を組み合わせたアンカー設定方式

第4章では、動画ハイパーメディアシステムのオーサリング技術として、アンカーの効率的な設定方式の確立を目的として、自動追跡と手動修正を組み合わせたアンカー設定方式の提案を行う。さらに、提案した方式に基づいてアンカー設定ソフトウェアを構築し、その有効性について、スポーツ映像を対象として評価実験を行った結果について述べる。

4.1 はじめに

計算機やネットワーク等のハードウェアの性能向上やオペレーティングシステム等のソフトウェア技術の進歩により、計算機上で音声や映像等のマルチメディア情報を扱うことが現実的となってきた。また、これらのマルチメディア情報はCD-ROM、DVD-ROM タイトルをはじめとしたパッケージ形式のコンテンツはもとより、インターネットの爆発的な普及にあわせて、ブロードバンドネットワークやデジタル放送におけるストリーミング形式のコンテンツにまで幅広い分野に適用されつつあり、いつでもどこでもマルチメディア情報を利用できる環境が実現されつつある。

このような背景のもと、従来のテレビ放送をはじめとする一方通行型の映像メディアに対して、利用者が主体的に自分の必要な情報を取得できるインタラクティブな映像メディアが求められるようになってきた。

動画ハイパーメディアシステムは、動画中に登場する人や物などのオブジェクトを含む領域に対して関連情報をリンクとして設定しておき、動画の再生中にその領域をマウス等のポインティングデバイスで直接指定することにより、関連情報を直観的に検索することができるシステムである。

動画ハイパーメディアシステムはインタラクティブな映像メディアを実現するためには比較的単純なシステムであるが、現状あまり普及していない理由の一つとしてコンテンツ制作時における以下の問題が考えられる。動画中に登場するオブジェクトの動きに追従して、リンクの起点となるアンカーの設定を行う必要があるが、現状では効率的なアンカー設定方式が確立していないため、単純に手作業でフレーム毎にオブジェクトの動きにあわせてアンカー設定を行った場合、非常に手間のかかる作業になってしまう。

そこで第4章では、動画中に登場するオブジェクトを含む領域に対して効率的にアンカー設定を行う方式について提案する。提案した方式は動画像解析によって登場するオブジェクトの動きを自動的に追跡するとともに必要に応じて手作業で修正を行う方式であり、次の特徴を持つ。

1. 双方向追跡の適用による追跡効率の向上
2. 中間点削除処理による手作業でのアンカー修正作業の効率化

さらに、本方式に基づいてアンカー設定ソフトウェアを構築し、その有効性についてスポーツ映像を対象としたアンカー設定作業に関する評価実験を行った。以下、4.2 で関連研究、4.3 でアンカー

設定方式の提案，4.4 でアンカー設定ソフトウェアの構築，4.5 で評価実験，4.6 で評価，4.7 で考察，4.8 でまとめを行う。

4.2 関連研究

従来から動画中に登場するオブジェクトを含む領域を指定することにより指定したオブジェクトに関連する情報を検索することが可能な動画ハイパーメディアシステムに関する研究は活発に行われてきた [49,51,81,113,114]。

しかし，動画ハイパーメディアシステムのコンテンツ制作時におけるアンカー設定作業の効率化に着目した研究はあまり行われていないのが現状である。ここでは動画ハイパーメディアシステムにおける動画画像解析を適用したオブジェクトの自動追跡に関する関連研究について述べる。

高田らは動画画像解析を適用したオブジェクトの自動追跡に関する研究を行っている [83]。この研究におけるオブジェクト追跡方式は，あるフレームにおいて追跡したいオブジェクトの領域を指定すると，後は動画画像解析により自動的に順方向に追跡処理を行う。自動追跡が失敗した場合はそのフレームに戻り，手作業により位置を修正した後で再度順方向に追跡を行う方式である。

現状の計算機性能と動画画像解析技術では，コンテンツ制作者の希望通りに自動追跡を行うことは技術的に不可能であり，必ず誤りが発生する。そのためシステムとして設定作業の有用性を向上させるには，自動追跡により発生した誤りを手作業で修正する方式を如何に組み合わせるかが重要となってくる。高田らのアプローチでは，追跡誤りが発生する度に追跡処理が中断し手作業で修正が必要が出てくるため，対象とする動画データの種類のよっては自動追跡処理中に何度も追跡処理が中断してしまう可能性がある。

岡田らは複数のオブジェクトなどの明確な輪郭を持たない領域オブジェクトの追跡に関する研究を行っている [61]。この研究では，明確な輪郭を持たない複数のオブジェクト領域の自動追跡，また領域オブジェクトが分裂した場合の自動検出を目的としたものである。また，この研究では，自動追跡が失敗した場合の修正については考慮されていない。

これらの関連研究事例に対して本研究のアプローチの基本方針として以下の2項目を設定した。

1. 動画画像解析を応用したオブジェクト領域の自動追跡の性能向上
2. 自動追跡の結果がコンテンツ制作者の意図に反した場合を想定した手作業での修正作業を効率化

4.3 アンカーモデルの提案

図 4.1 に本研究にて提案するアンカーモデルについて示す。

提案したアンカーモデルは，3.5 で提案したビデオオブジェクトモデルの代表フレームを用いた方式を拡張したもので，動画中に登場するオブジェクトの動きにあわせてアンカーを設定できるモデルとしている。

図 4.1 を見ても分かる通り，動画中に登場するオブジェクトを含む領域に対して設定されたアンカーは，すべてのフレームに対してデータを持たせるのではなく，開始点と終了点と幾つかの中間点の含まれるフレームに対してデータを持たせ，データを持たないフレームは前後のアンカーを持つフレームのデータから補間計算によって推定する形式とした。本アンカーモデルでは，開始フレーム，終了フレーム，中間フレームのアンカー情報を持つフレームを基準フレームと定義する。また，中間のア

ンカー情報を持たないフレームを標準フレームと定義する．標準フレームにおけるアンカー領域は前後の基準フレームから補間計算によって得ることができる．

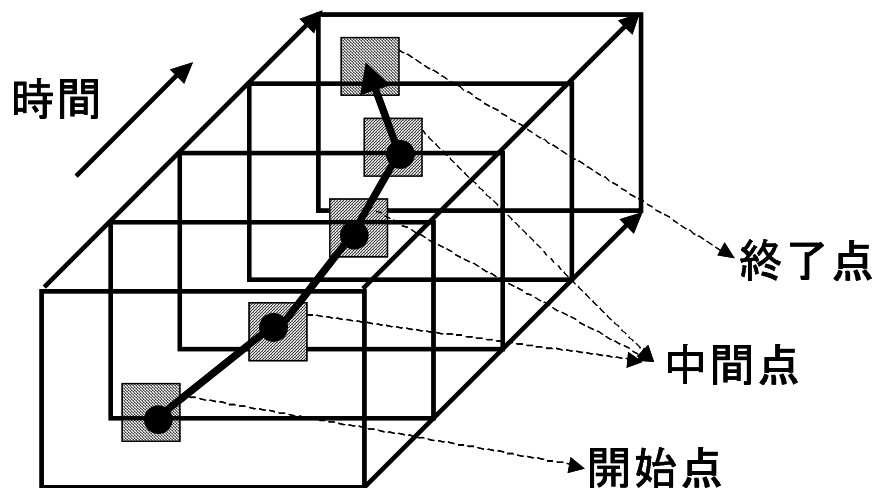


図 4.1: アンカーモデル

本モデルで扱えるアンカーの形状としては，設定効率の簡易化を目的として矩形や円形などの幾何学図形を前提とし，補間計算により前後の基準フレームから標準フレームのアンカーを推定する．

次に，アンカーの出力方式について検討を行った．図 4.2 に本方式におけるアンカーの出力例について示す．

設定したアンカーは元の動画データと対応するテキストデータとして出力する方式とし，図 4.2 に示した通り，アンカーの定義は，[HotSpot]:アンカーの動き，形状の定義，[link]:クリックされた時のアクション，[Effect]:表示情報の 3 つの領域に分けて行う．

[HotSpot] はアンカーの動き，形状の変化について定義する．アンカーの定義はヘッダー領域に，

<アンカー ID:アンカー名:アンカー形状:開始フレーム - 終了フレーム:基準フレーム数>

と記述し，その下に基準フレームの数だけ，フレーム番号とアンカーの座標データを記述する．

アンカーの座標データは，対象とする動画データの解像度に依存しない形式とし，対象とする動画データの解像度を QVGA(320 × 240) サイズに拡大，縮小した座標データを記述する．これにより，アンカー設定処理後に，動画データ自体の解像度の変換処理を行いたい場合や，再生時に表示サイズを変更して表示したい場合に対応可能である．

[Link] はアンカーに対するリンクをクリック時のアクションとして記述する．図 4.2 の場合，AncID1 のアンカーに対して URL のアクションが設定されており，AncID1 のアンカーをクリックすると Web ブラウザに対して引数で指定された URL を通知する．また，AncID2 のアンカーに対しては SEEK のアクションが設定されており，AncID2 のアンカーをクリックすると同じ動画データの指定されたフレームにジャンプする．

[Effect] はアンカーの表示情報について記述し，通常時の表示色，クリック時の表示色などの指定を行う．

```

[HotSpot]
<AncID1:AnchorName1:R:0-64:10>
0:172,87,245,160
10:162,77,245,154
15:178,82,266,161
20:197,95,291,176
25:199,99,298,182
30:182,106,286,191
35:171,97,281,184
45:144,93,264,184
55:91,105,222,200
64:44,116,185,216

<AncID2:AnchorName2:C:8-73:5>
8:163,84,243,164
24:201,96,296,191
48:129,82,247,200
62:73,112,175,214
73:38,104,153,219

[Link]
AncID1:URL:http://www.melco.co.jp/"AnchorInfo":""
AncID2:SEEK:41:""

[Effect]
AncID1:0x000000ff:3:0x00ff0000:1
AncID2:0x000000ff:3:0x00ff0000:1

```

動き、形状の変化

クリック時のアクション

視覚効果

図 4.2: アンカーの出力例

4.4 アンカー設定方式の提案

4.4 では、本研究で提案するアンカー設定方式について述べる。これまで筆者は、動画中に登場するオブジェクトの追跡処理に動画像解析を適用する研究を行ってきた [115-119]。

4.2 で設定したアプローチに基づいてアンカーの設定作業の方式検討を行うにあたり、動画像解析技術を利用したオブジェクトの自動追跡を行うとともに、手作業によるアンカー修正作業を組み合わせることにより、手間のかかるアンカー設定作業の効率化を図ることを目的とする。

また、アンカー設定作業の実用のための目標値としてアンカー自動設定後の手作業による修正の必要なフレーム数の目標値を設定する。目標値設定のための前提条件として、対象とする映像データの長さを 10 分とし、常時平均 1 個のアンカーの設定を行うこととし、アンカー設定において修正作業に要する 1 フレームあたりの時間を 5 秒と仮定する。

実用化を視野に入れて、長さ 10 分の映像データに対するアンカーの修正作業に割ける時間は高々 1 時間と考え、アンカー自動設定後の手作業による修正の必要なフレーム数の目標値を 1 秒あたり 1 ~ 1.5 個と設定する。この条件でのアンカーの修正時間は、1 秒あたり 1 個の場合で 50 分、1.5 個の場合で 75 分となる。

4.4.1 アンカー設定方式

4.3 で提案したアンカーを設定する方式について提案する。基本的には、次に示す動画像解析による自動追跡と手作業による修正を組み合わせた方式とする。

1. 開始点・終了点の設定:

アンカーの開始フレーム，終了フレームにおける位置，大きさを設定する．この操作は，動画を開始フレームまたは終了フレームに位置付け，表示されたオブジェクトに重ねて矩形のアンカーを描くことにより行う．

2. 動画像解析を用いた自動アンカー設定:

1. を行った後，コンテンツ制作者がアンカーの自動設定を指示すると，システムは動画像解析を行い，その結果に従いオブジェクトの動きにあわせて自動的に中間の基準フレームを設定する．

3. 誤りの修正:

コンテンツ制作者は自動アンカー設定の結果を画面上で確認し，必要に応じてアンカーの修正を行う．修正は任意の位置に中間の基準フレームを挿入するか，既に挿入されている基準フレームを削除またはアンカーの位置や大きさを手作業により修正することにより行う．

また，動画像解析を用いた自動アンカー設定処理の流れは以下の方式とする．

1. 順方向追跡:

コンテンツ制作者が設定した開始フレームのアンカーを最初のテンプレートとして，あらかじめ設定されたフレーム間隔でパターンマッチング処理を繰り返して終了フレームまでオブジェクトの追跡を行う．この処理によって，設定されたフレーム間隔毎に中間点が設定される．

2. 追跡結果判定:

追跡の結果得られた終了フレームのアンカー位置と，コンテンツ制作者が指定した終了フレームのアンカー位置が一致していれば 4. に進む．一致していない場合は 3. に進む．

3. 逆方向追跡:

コンテンツ制作者が設定した終了フレームのアンカーを最初のテンプレートとして 1. と逆方向の追跡を行う．追跡中に 1. で得られた軌跡に到達したらその時点で追跡を打ちきる．逆に，追跡中に 1. で得られた軌跡に到達しなかった場合は，全区間の 50 % に到達した時点で逆方向の追跡を打ちきる．

4. 中間点削除:

以上の処理で得られた軌跡をできる限り少ない数の中間点で近似するように，冗長な中間点の削除を行う．

また，本方式の特徴としては次の 2 点があげられる．

1. 双方向の追跡結果の併用:

コンテンツ制作者によるアンカーの存在区間の指定と開始フレーム，終了フレームの設定を同時に行うことで，順方向，逆方向の追跡処理が可能となる．パターンマッチングを用いた追跡の場合，途中で一度追跡を誤るとその後の追跡が無駄になってしまう．本方式では双方向の追跡結果を併用することにより，追跡の精度を向上させることが可能となる．

2. 誤り修正を容易にするための中間点削除:

追跡の結果得られた軌跡に基づき中間点を設定すると，設定されたフレーム間隔毎に中間点が挿入されることになる．このような場合，オブジェクトが直線的な動きをしている場合にはデータが冗長になってしまい，誤りの修正に多くの作業量を必要とする．そこで，本方式では，数フレーム毎に挿入された中間点のうち，冗長な中間点を削除する処理を行うことにより，誤り修正の手順を大幅に削減することができる．

4.4.2 自動追跡アルゴリズム

本提案における順逆方向のオブジェクトの自動追跡アルゴリズムについて説明する。

モデル領域の設定

基準フレーム (時刻 t とする) において矩形で指定されたアンカー領域をマッチングのモデル領域として設定する。コンテンツ制作者は、アンカーを図 4.3(a) に示した通り実際の物体より大きめに領域を指定する傾向があるため、パターンマッチングのモデルとしては、コンテンツ制作者が実際に指定したアンカー領域を 10 % 縮小した領域を利用する (図 4.3(b))。

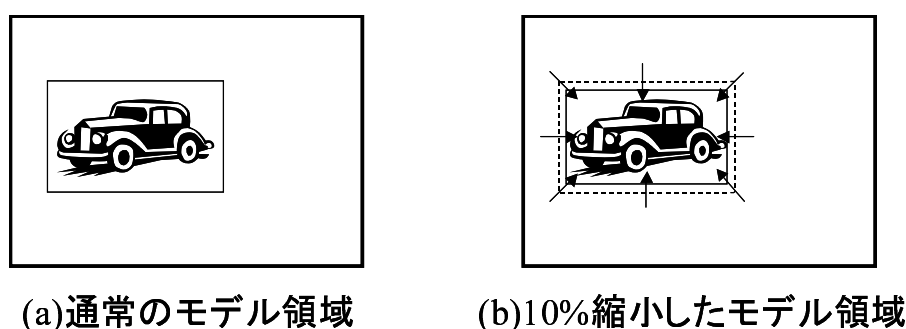


図 4.3: モデル領域の設定

サーチ範囲の設定

基準フレームの次時刻 (時刻 $t + \Delta t$ または $t - \Delta t$: Δt は設定されたフレーム間隔) のフレームをサーチフレームとし、サーチフレームにおいてサーチ範囲を設定する。サーチ範囲はモデル領域を上下左右に所定の画素数だけ拡大した領域である。

マッチング処理

マッチング処理は式 (4.1) に示した正規化相関を用いたパターンマッチングを用いる。

$$r = \frac{N \sum IM - \sum I \cdot \sum M}{\sqrt{[N \sum I^2 - (\sum I)^2][N \sum M^2 - (\sum M)^2]}} \quad (4.1)$$

$$S = \max(r^2, 0) \times 100 \quad (4.2)$$

ただし、

- N :モデルのピクセル数
- I :モデルのピクセル値
- M :対象領域のピクセル値

とする。

式 (4.1) を用いてサーチ範囲内のすべての位置においてモデル領域との相関を計算し、スコア S を算出する。算出したスコア S が予め設定した閾値 (アクセプタンスレベルと呼ぶ) を越えた場合、マッチングが取れたと判断する。

マッチングの結果、すべてのスコアがアクセプタンスレベル以下の場合、サーチフレームを再設定して、再度マッチング処理を行う。例えば、時刻 $t + \Delta t$ においてマッチングが取れなかった場合は、時刻 $t + \Delta t/2$ にサーチフレームを再設定して、再度マッチング処理を行う。

また、スコアがアクセプタンスレベル以上のマッチングが複数存在した場合は、次の手順によりマッチング結果を決定する。

1. アクセプタンスレベル以上のマッチングの取れた領域の位置とスコアをスコアの高い順に上位 20 位まで保持する。
2. このうちスコアが 1 位の領域から決められた範囲内 (スコアマージンと呼ぶ) に存在するものを正解候補とし、正解候補の移動量を算出する。
3. 正解候補のうち、移動量の最も少ない領域をマッチング結果とする。

この理由は、フレーム間隔を細かく設定することを前提条件とすれば、似通った形状のオブジェクトがサーチフレームのサーチ範囲に複数存在する場合は、モデル領域からの移動量が少ないものが正解である可能性が高いと考えたためである。

上記処理によって得られたマッチング結果を新たにモデル領域とし、サーチフレームを新たに基準フレームに設定し、マッチング処理を繰り返すことにより、オブジェクトの自動追跡を行う。

4.4.3 中間点削除アルゴリズム

4.4.2 の自動追跡処理により、すべてのサーチフレームに対してアンカー位置が決定される。しかし、このままでは中間点が設定されたフレーム間隔毎にアンカーの情報が存在することになるため、例えばオブジェクトが直線的な動きをしている場合などはデータが冗長になってしまう。このような形式でデータを保持していた場合、追跡誤りの修正をする際に多くの中間点を修正する必要が生じるため手間が増えてしまう。そのため、直線的な動きをしている区間は、その両端の点のみを中間点として保持しその他の中間点を削除してなるべく少ない中間点数にする必要がある。

中間点削除は線図形の直線近似で用いられる区分的直線近似法を 3 次元に拡張した方式を用いる。

具体的には、開始点の中心と終了点の中心を結ぶ直線をひき、この直線とすべての中間点の中心との距離を計算する。この中で最も距離が離れた点を接点とし、できた 2 つの線に対して同様のことを繰り返して行う。動画は 3 次元空間とみなすことができるので、直線の式は式 (4.3) に示すような 3 次元空間上の直線の式を用いる。

開始フレームのアンカー領域 (開始点) の中心点を $P_s(x_s, y_s, t_s)$ とし、終了フレームのアンカー領域 (終了点) の中心点を $P_e(x_e, y_e, t_e)$ とすると、 P_s から P_e にひいた直線上を通る任意の時刻における点 $P(x, y, t)$ の座標は次のようにして求められる。

$$\begin{aligned} k &= \frac{t - t_s}{t_s - t_e} & (4.3) \\ x &= k(x_s - x_e) + x_s \\ y &= k(y_s - y_e) + y_s \end{aligned}$$

これにより求められる任意の対象時刻における中間点の中心との距離を求める．直線上の任意の時刻における座標は式 (4.3) により求めることができ，求められた座標 P と同一時刻の中間点の中心座標 $P_b(x_b, y_b, t_b)$ との距離 d を式 (4.4) により計算する．このとき，同一時刻を見ているため t の項は 0 となる．

$$d = \sqrt{(x_b - x)^2 + (y_b - y)^2 + (t_b - t)^2} \quad (4.4)$$

式 (4.4) を用い 2 点間の距離を求め，これが最大になった地点を接点とし，再びこの操作を繰り返す．これにより誤差範囲 E_r 以上移動したもののみを中間点として保持し，その他の中間点を削除することができる．

4.5 アンカー設定ソフトウェアの構築

提案したアンカー設定方式を適用した動画ハイパーメディアシステムのアンカー設定ソフトウェアを IBM PC/AT 互換機 (CPU : Pentium3(700MHz), メモリ : 256MB, OS : Windows2000 Professional) 上に構築した．図 4.4 に構築したソフトウェアの画面イメージについて示す．

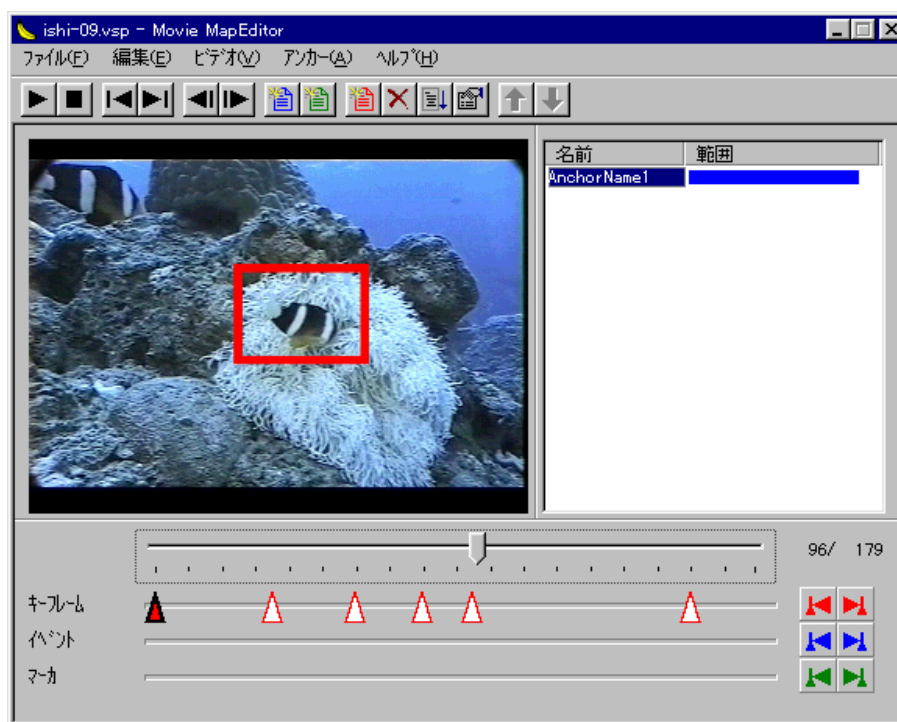


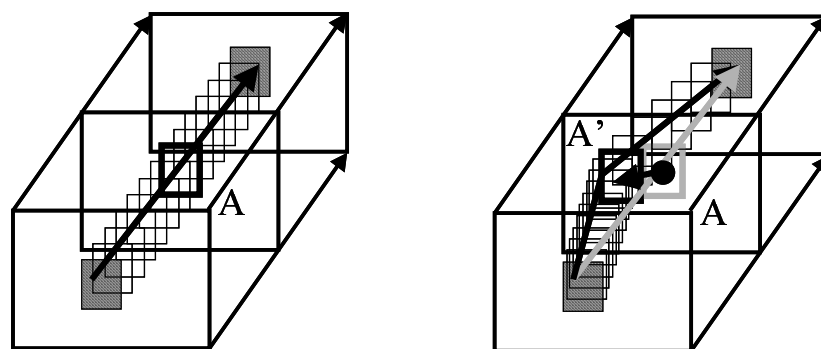
図 4.4: アンカー設定ソフトウェアの画面イメージ

アンカー設定ソフトウェアでは，AVI 形式，または QuickTime Movie 形式の映像データに対するオブジェクトの自動追跡処理を実装した．

アンカー設定ソフトウェアでは，開始フレームと終了フレームにおいて対象とするオブジェクトの位置を直接指定することによって，その間のフレームにおけるオブジェクトの動きは提案した自動追

跡処理によって自動的に設定される．自動追跡処理より設定された中間点は中間点削除処理により冗長な中間点が削除された形でウィンドウ上に表示され，表示されたアンカーを手作業により直接編集することによって修正でき，自動追跡の結果に誤りがある場合でも，自動追跡結果を簡単に修正することができる．

また，図 4.5 に手作業による中間点の編集について示す．



(a)開始点と終了点の設定

(b)中間点の編集

図 4.5: アンカーの編集

具体的には図 4.5(a) に示した通り，設定対象のオブジェクトの開始フレームと終了フレームにおいて位置の指定を行うと，その間のフレームにおけるオブジェクトの位置は開始点と終了点のオブジェクトの位置から線形補間によって推定する．しかし，この場合はオブジェクトが直線的に同じ速度で動く場合でしかアンカーを設定することができない．

そこで，設定対象のオブジェクトの動きにあわせてその間の動きに大きな変化のあった点を中間点として設定することにより，オブジェクトの複雑な動きに対応することが可能となる．

具体的には図 4.5(b) に示した通り，A は A' の位置が正しいオブジェクトの位置を表している場合は，アンカー設定ソフトウェア上で A を A' の位置に動かすことによってその点が自動的に中間点として設定される．また，不要な中間点を指定して削除することも可能である．

この作業を繰り返すことによって，複雑な動きをするオブジェクトに対しても手作業によりアンカーの設定を行うことが可能となる．

4.6 評価実験

4.6.1 評価実験の内容

構築したアンカー設定ソフトウェアを用いて本研究にて提案したアンカー設定方式についての評価を行うにあたり，提案したアンカー設定方式の有効性を示すことを目的として以下の 2 項目についての評価を目的とした評価実験を行った．

1. 追跡アルゴリズムの評価
2. 中間点削除処理の評価

4.6.2 評価方法

評価用データ

将来、ブロードバンドネットワークでの映像配信サービスにおけるコンテンツ制作作業への適用を視野に入れて、評価用データとしてスポーツ映像を選択した。例えば、スポーツ映像のオンデマンド配信サービスに動画ハイパーメディア機能を適用することにより、映像データに登場する選手を直接指定することによって、その選手のプロフィールや成績データを示した Web ページを表示するサービスなどを実現することができる。

評価実験で使用した動画データは表 4.1 に示した 11 種類のスポーツ番組の録画ビデオから動画ハイパーメディアシステムに利用できそうなカットを 10 カットずつ合計 110 カット撰択した。選択したカットの時間は全クリップ合計で 949.577 秒、1 カット平均で 8.63 秒となった。その中から実際にアンカーとして使用できそうなオブジェクトを 10 個ずつ合計 110 個選択した。オブジェクトの選択基準としては、例えば球技であればボールを持っている選手などそのカットの中心となるオブジェクトを選択する方針とした。その結果、オブジェクトの平均存在時間は 5.07 秒となった。動画データは PC 上の AVI ファイルを利用し、表 4.2 に示した形式でキャプチャ作業を行った。

また、評価実験で使用した PC は CPU に Pentium3(700MHz) を使用し、メモリを 256MB 搭載した IBM PC/AT 互換機で、OS は Windows2000 Professional を使用した。

また、評価実験に使用したオブジェクト追跡処理のパラメータは次の通りとした。

- アクセプトランスレベル = 10.0
- フレーム間隔 = 5
- サーチ範囲 = モデルの中心 \pm 40
- スコアマージン = 20
- 中間点削除誤差範囲 E_r = 16

追跡アルゴリズムの評価実験

準備したすべての動画データに対して同一の位置に開始点・終了点を設定し、アンカー自動設定処理を実行した。この際に逆方向の追跡を行う場合と行わない場合についてそれぞれ評価実験を行った。本評価実験では中間点削除処理は行わず 5 フレーム毎に追跡処理を行いその結果について評価を行った。また、アンカーの形状は矩形を選択した。

評価方法は、まず手作業により正解となるアンカー領域の設定を行い、アンカー自動設定処理を行った結果得られた各基準フレーム上に設定されたアンカー領域と正解のアンカー領域との重なり面積を算出した。

得られた基準フレーム上のアンカー領域の精度を示す指標として、各基準フレームにおける適合度 R を式 (4.5) の通り定義し、基準フレーム毎に適合度の算出を行った。

$$R = \frac{S_b}{S_a} \quad (4.5)$$

S_b : 得られた基準フレームのアンカー領域と
正解のアンカー領域の重なり面積

S_a : 得られた基準フレームのアンカー領域の面積

表 4.1: 評価用データ

スポーツ種類	設定対象	平均時間 (秒)	特徴
スキューバダイビング	魚	7.87	設定対象は1匹でズーム有り, 設定対象と背景の一部が重なる場合がある.
テニス	選手	7.39	設定対象は2名. 選手と線審が重なる場合がある.
スピードスケート	選手	12.52	設定対象は2名でほぼ直線的な動作.
サッカー	選手	7.81	設定対象は6~7名で, 選手同士で重なる場合がある.
アメフト	選手	8.08	設定対象は6~7名で, 選手同士・選手と背景の一部が重なる場合がある.
ラリー	車	8.10	設定対象は1台で, ズーム有り.
スキージャンプ	選手	7.04	設定対象は1名で, 設定対象と背景の一部が重なる場合がある.
バスケットボール	選手	9.83	設定対象は6~7名で, 選手同士や選手と審判が重なる場合がある.
フィギアスケート	選手	8.75	設定対象は2名で, 選手同士, 選手と背景の一部が重なる場合がある.
アイスホッケー	選手	8.08	設定対象は6~7名で, 選手同士, 選手と背景の一部が重なる場合がある.
野球	選手	9.49	設定対象は4~5名で, ズーム有り. 選手同士や選手と背景が重なる場合がある.
平均	-	8.63	-

表 4.2: キャプチャパラメータ

圧縮形式	DV
フレームレート	29.97 フレーム/秒
画像サイズ	720 × 480, 24 ビット
オーディオ形式	PCM, 44.100kHz, 16 ビット, ステレオ

図 4.6 に各基準フレームにおける適合度の定義について示す。

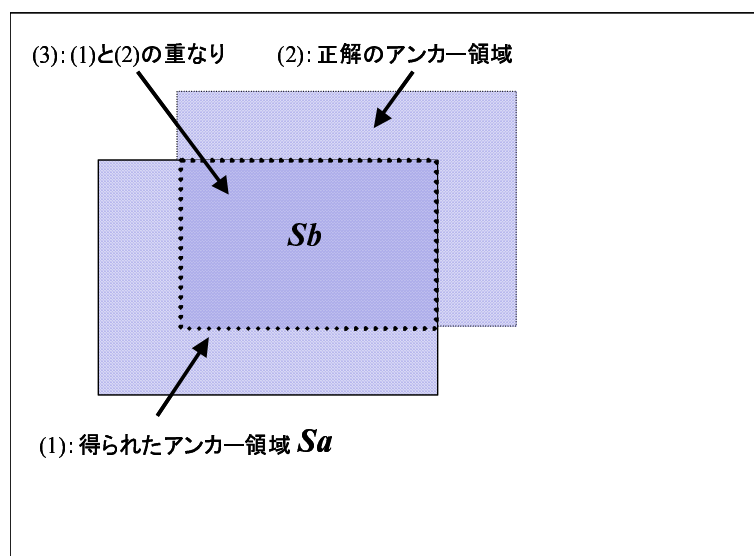


図 4.6: 適合度の定義

中間点削除処理の評価実験

準備したすべての動画データに対して、同一の位置に開始点、終了点を設定し、アンカー自動設定処理を実行した。この際に、中間点削除処理を行った場合と行わない場合について評価実験を行った。また、この際に逆方向の追跡処理は行う方針とした。

評価方法は追跡アルゴリズムの評価実験と同様に、正解となるアンカー領域の設定を事前に行い、各基準フレームで得られたアンカー領域と正解のアンカー領域との重なりを適合度として算出した。

4.7 評価

4.7.1 追跡アルゴリズムの評価

アンカーの自動設定処理を行った結果、トータルで 3645 フレーム (1 カット平均 33.1 フレーム) の基準フレームが設定された。

まず、順方向追跡の適合度が低かった 10 個のビデオクリップに対してその原因について分析を行った後で逆方向追跡を行い、逆方向追跡の効果について評価を行った。その結果について表 4.3 に示す。

表 4.3 の結果を見ても分かる通り、すべてのケースについて逆方向追跡による適合度が向上し、平均で 0.340 ポイント向上する結果が得られた。

次に、スポーツ種類毎に適合度を計算したところ、表 4.4 に示した通り、逆方向追跡を行わなかった場合で平均 0.666、順方向追跡後さらに逆方向追跡を行った場合で平均 0.744 という結果が得られた。

上記結果より、追跡アルゴリズムに逆方向追跡を追加することにより、順方向追跡だけ行う場合と比較して適合度が平均 11.7 % 向上するという結果が得られた。

表 4.3: 順方向追跡の適合度の低かったビデオクリップ

ビデオクリップ名	適合度 A(順方向)	原因	適合度 B(逆方向)	B-A
野球 5	0.195	(a)	0.564	0.369
テニス 6	0.211	(b)	0.491	0.280
フィギアスケート 3	0.211	(a)	0.633	0.422
サッカー 6	0.238	(a)	0.610	0.372
アメフト 8	0.249	(b)	0.693	0.444
ラリー 10	0.261	(a)	0.594	0.333
スキューバダイビング 4	0.268	(a)	0.562	0.294
フィギアスケート 6	0.300	(a)	0.578	0.278
スキージャンプ 8	0.314	(a)	0.611	0.297
サッカー 5	0.322	(a)	0.634	0.312
平均	-	-	-	0.340

(a) 背景の一部との重なり, (b) 選手 (審判) との重なり

表 4.4: 自動追跡の実験結果

スポーツ種類	適合度 (順方向)	適合度 (逆方向)
スキューバダイビング	0.658	0.772
テニス	0.660	0.675
スピードスケート	0.695	0.799
サッカー	0.617	0.709
アメフト	0.758	0.775
ラリー	0.701	0.785
スキージャンプ	0.647	0.684
バスケットボール	0.621	0.673
フィギアスケート	0.641	0.796
アイスホッケー	0.704	0.771
野球	0.626	0.742
平均	0.666	0.744

このように追跡が順方向追跡のみの場合は途中一度でも対象オブジェクトを見失った場合はその後の追跡が無駄になってしまう。このような場合を想定して、順方向追跡に逆方向追跡を組み合わせることにより、アンカーの自動設定処理の精度を向上させることが確認できた。

ただし、全体の 12.7 % にあたる 14 個のオブジェクトに対する逆方向追跡の適合度が順方向追跡の適合度を下まわっていた。その結果を表 4.5 に示す。

表 4.5 にリストアップされたビデオクリップのオブジェクトのうち、特に逆方向追跡の結果が悪くなった「アメフト 6」、「スキージャンプ 7」、「テニス 7」について分析を行ったところ、下記の共通点を得られた。

1. 順方向追跡区間の終わりの方 (逆方向追跡区間の始まりの方) に対象オブジェクトを見失った。
2. オブジェクトを見失った理由として、「アメフト 6」と「テニス 7」は対象オブジェクトが小さく背景の一部と重なった地点から背景を追いかけてしまった、「スキージャンプ 7」は対象オブジェクトの形が変わった地点から背景を追いかけてしまった、と考えられる。図 4.7 に「テニス 7」の逆方向追跡において対象オブジェクトを見失った地点について図示する。

表 4.5: 逆方向追跡結果の適合度が順方向追跡結果を下まわったビデオクリップ

ビデオクリップ名	適合度 A(順方向)	適合度 B(逆方向)	A-B
アメフト 6	0.896	0.539	0.356
スキージャンプ 7	0.912	0.591	0.320
テニス 7	0.791	0.477	0.314
アメフト 3	0.882	0.631	0.251
バスケットボール 10	0.733	0.498	0.235
バスケットボール 5	0.716	0.524	0.193
スキージャンプ 1	0.825	0.679	0.146
サッカー 8	0.839	0.702	0.136
アイスホッケー 6	0.833	0.711	0.122
フィギアスケート 7	0.761	0.715	0.046
スキージャンプ 2	0.693	0.649	0.044
スピードスケート 7	0.903	0.859	0.044
ラリー 2	0.723	0.680	0.043
サッカー 3	0.618	0.595	0.024
平均	-	-	0.162

4.7.2 中間点削除処理の評価

アンカーの自動設定処理を行った結果、トータルで 3645 フレーム (1 カット平均 33.1 フレーム) の基準フレームが設定された。中間点削除処理を行った結果、トータルで 1032 フレーム (1 カット平均 9.38 フレーム) に削減された。まとめると 1 カット平均で 71.7 % の基準フレームが削除されたという結果が得られた。

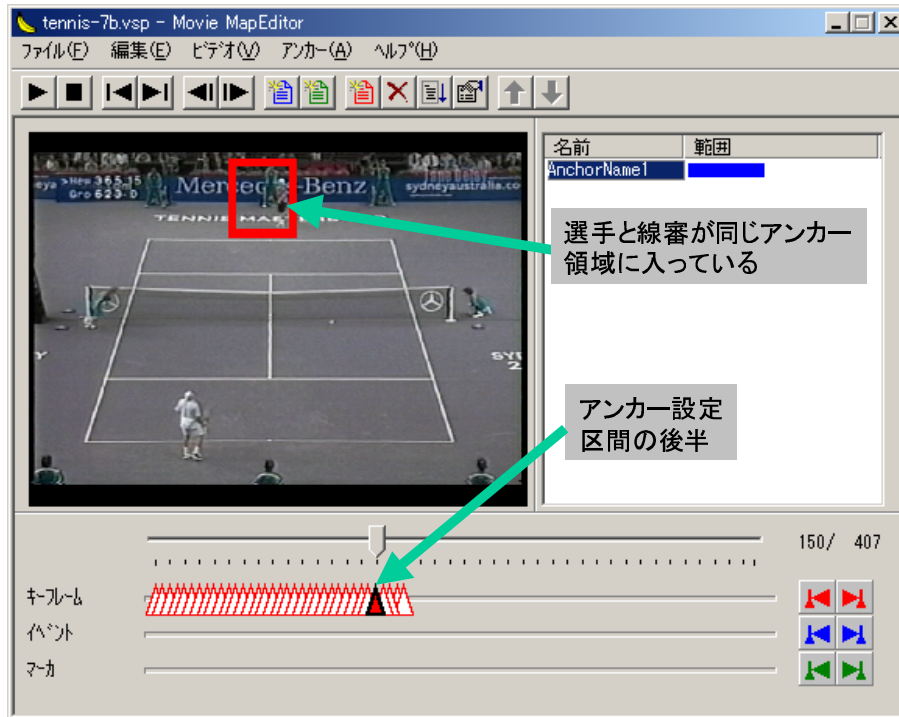


図 4.7: 「テニス 7」の逆方向追跡結果

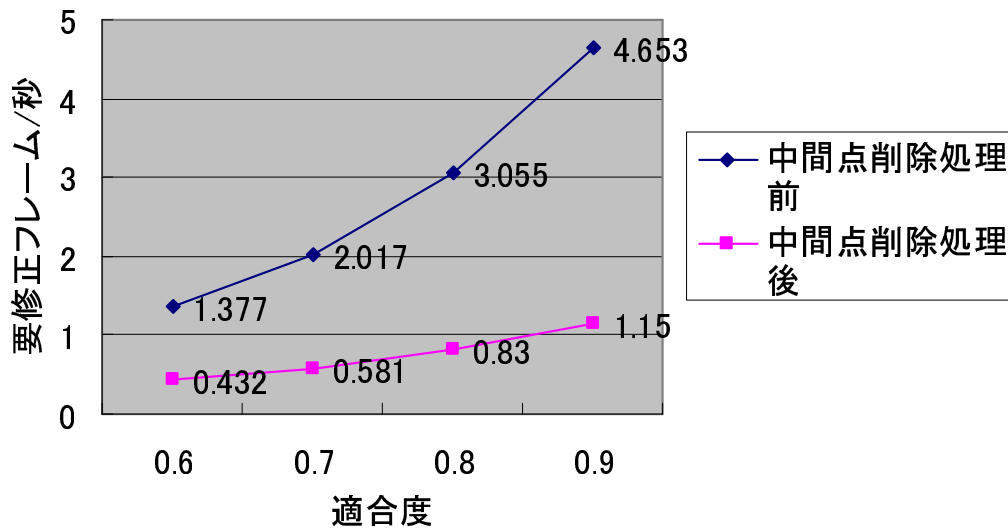


図 4.8: 中間点削除処理の実験結果

得られた評価実験結果に対して、目標とする適合度をそれぞれ0.6, 0.7, 0.8, 0.9に設定し、目標とした適合度を満たさない基準フレームを要修正フレームとして集計した。その集計結果を図4.8に示す。

図4.8の結果から、本方式における適合度の目標値を0.9とした場合、4.7.1の結果から自動設定処理の結果得られる基準フレームの適合度は今回評価実験を行った110カットの平均値で0.744であり、目標適合度の0.9を満足するためには1秒あたり平均4.653個の要修正フレームを手作業により修正する必要がある。この前処理として中間点削除処理を行うことにより、要修正フレームの数が1秒あたり平均1.150個に削減されるという結果が得られた。

1フレームあたりのアンカー修正作業の平均作業量が一定であると仮定すると、中間削除処理により手作業で修正が必要なフレームの数が1秒あたり4.653個から1.150個に削減できるため、中間点削除処理により約75%の作業の効率化を図ることが可能となる。

4.8 考察

4.7の結果に基づき、本研究にて提案したアンカー設定方式について考察する。

以下、従来技術を5フレーム毎に手作業で基準フレームを設定する方式とし、本研究にて提案する方式と編集作業効率の比較を行う。従来技術では5フレーム毎に基準フレームを設定するため編集の必要な基準フレーム数は1秒あたり5.994個となる。

システムの適合度の目標値を0.9とした場合、本研究にて提案する方式に基づく順逆方向の追跡処理を行うことにより、修正の必要な基準フレーム数は1秒あたり4.653個に削減され、従来技術と比較して24%の編集作業の効率化を図ることが可能となる。

さらに本研究にて提案する方式に基づく中間点削除処理を行うことにより、修正の必要な基準フレーム数は1秒あたり1.150個に削減されるため、従来技術と比較して81%の編集作業の効率化を図ることが可能となる。また、この1.150という値は4.3で設定した1~1.5という目標値を達成した。

上記結果から1カット当たりのアンカーを設定した区間の長さの平均値を5秒と仮定した場合、アンカー自動設定処理後1カットあたり平均5.750個の基準フレームを修正することにより、アンカーの自動設定処理と手作業による修正作業を組み合わせた形でのアンカー設定作業が可能となる。

4.9 おわりに

第4章では、動画ハイパーメディアシステムにおいて、リンクの起点となるアンカー設定を効率的に行う方式として動画像解析による自動追跡と手動による修正を組み合わせる方式を提案し、その方式に基づいてアンカー設定ソフトウェアを構築した。

構築したソフトウェアを実際にスポーツ映像を対象として評価実験を行い、有効性についての検証を行った。

評価実験結果をまとめたところ、本方式の特徴である次の有効性について確認することができた。

1. 双方向追跡の適用による追跡効率の向上
2. 中間点削除処理による手作業による修正作業の効率化についての有効性

今後は、さらに効率的なアンカー設定方式について検討を行うとともに、今回提案した方式を動画ハイパーメディアシステムの応用システムのコンテンツ制作に広く適用していく予定である。

第5章 拡張イベントモデルによる映像と付加情報の同期配信方式

第5章では、動画ハイパーメディアシステムの配信技術として、映像と付加情報の同期配信方式の確立を目的として、拡張イベントモデルの提案を行う。さらに、提案したモデルを Windows Media, MPEG-2 TS 形式の映像配信システムに対するエンコード/デコードライブラリとして実装し、実装したライブラリを用いて評価システムを構築し、その有効性について評価実験を行った結果について述べる。

5.1 はじめに

近年のインターネットの急速な常時接続化、広帯域化の流れにより、FTTH, ADSL, CATV などの常時接続で高速なブロードバンドインターネット環境が各家庭で利用可能な時代が近づきつつある[1]。この常時接続化の流れにより家庭では情報を受けるだけでなく発信することが可能に、また広帯域化の流れにより従来中心だった文字、静止画から音声、動画をはじめとする映像情報の配信が可能となりつつある。

一般的に映像情報は情報量が多く圧縮して配信する必要があるためその符号化、復号化処理には多くの計算機資源を必要とする。この処理は連続的に行われるため実時間処理が必要であるが、圧縮技術、ストリーミング技術、ハードウェアの性能向上などによってインターネット上での映像情報の配信が可能となりつつある。

このような背景のもと、文字、静止画、音声、動画などを含む複数のメディアから構成されるコンテンツの配信に対する要求が増加している。例えば、WWW は HTML[7] や HTTP (Hyper Text Transfer Protocol)[120] などの技術を組み合わせることにより、複数のメディアを組み合わせたコンテンツを実現している。また、WWW は単に複数のメディアを複合するだけでなく、ハイパーリンクにより関連する文書を相互に関連付けることができ、コンテンツ利用者は自分の意思でハイパーリンクを辿ることにより、対話的に関連するメディア間の相互アクセスを実現している。

本研究では、ストリーミング技術を用いた映像配信サービスに対するハイパーリンク機能の追加を目的として、映像と関連情報を結びつける付加情報について定義し、映像と付加情報の同期配信方式の設計、実装と評価システムの構築及びその評価について述べる。

本研究における付加情報はハイパーリンク機能の鍵となる情報であり、映像と時間的關係、空間的關係が定義されている情報である。付加情報は映像と同期して配信され、映像の再生にあわせて適切な時間に適切な位置に表示される。コンテンツ利用者が映像上に表示されている付加情報を選択してクリックすると、関連情報の表示などの設定されているアクションが実行される。

本研究では、対象とする映像配信サービスの特徴を分析した結果、次のシステム要件を設定した。

1. 映像の形式や配信方式に依存しない同期方式であること
2. 付加情報の使用目的にあわせた同期精度が設定可能なこと

従来の動画ハイパーメディアの研究では、映像の配信方式に依存しない同期方式に着目した研究はほとんどされていない。また、従来のメディア間同期方式に関する研究では、メディア間の同期を行うための同期情報を設定し、その同期情報を用いてメディア間の同期処理を行う研究が多くされているが、本研究で対象とする映像配信サービスに対するハイパーリンク機能を実現するための映像と付加情報の同期配信に適用可能な研究はされていない。

本研究にて提案する同期配信方式では、映像中に対応する付加情報との時間的関係を定義した同期情報を周期的にイベントとして多重化し、その同期情報を元に映像と付加情報の同期配信を実現する。

提案方式はメディア間の同期にイベントを利用する方式であるが、通常、イベント処理はCPU負荷に大きく影響するため、実装時にはその多重化精度の最小値を考慮する必要がある。特に、動画ハイパーメディアとして映像上に登場するオブジェクトの動きにあわせて付加情報を表示する場合は、実時間で同期処理が必要となり、この数値が大きく影響するため、評価システムを構築し、その実時間性能について評価を行う方針とする。

以下、5.2で映像に対するハイパーリンク機能の定義とシステム要件について述べる。5.3で関連研究について述べ、5.4で設計指針について検討し、その結果に基づき映像と付加情報の同期配信を目的とした拡張イベントモデルの設計を行った。5.5で設計結果に基づき拡張イベントモデルをWindows Media[121]とMPEG-2 TS[122]を対象としてエンコード/デコードライブラリとして実装し、5.6で実装したライブラリを用いて映像と付加情報の同期配信の評価システムを構築した。5.7で構築した評価システムの実時間同期性能についての評価実験を行った。さらに5.8で考察、5.9でまとめを行う。

5.2 ハイパーリンク機能の定義とシステム要件

本研究で対象とする映像配信サービスにおけるハイパーリンク機能について定義し、そのシステム要件についてまとめる。

映像に対するハイパーリンク機能とは、インタラクティブ映像技術を構成する一機能であり、外村は文献[123]にて、「通常静止画の全体あるいは一部分のホットスポットによくはられているハイパーリンクを、映像に対してもはることができるもので、特に動くオブジェクトに対しても、動くホットスポットを定義する機能」と定義している。

本研究では、映像配信サービスに対するハイパーリンク機能の追加を目的として、その鍵となる付加情報を映像と同期して配信する方式について提案する。付加情報は外村の定義のホットスポットに相当し、映像と同期して配信され、映像の再生にあわせて適切な時間に映像上の適切な位置に表示される。コンテンツ利用者が要求に応じて映像上に表示されている付加情報を選択してクリックすると、関連情報の表示などの設定されているアクションが実行される。

次に、本研究で対象とする映像配信サービスに対するハイパーリンク機能における付加情報について定義する。本研究では、その目標とする同期精度の違いから次の2種類の付加情報を定義する。

1. バナー型付加情報:

映像と連動したスポンサーなどの広告表示や字幕サービスなどを目的として、配信された映像と同期してスポンサーのバナー広告や字幕情報を映像上の決められた位置に表示する。同期精度は映像と開始点での同期が必要な付加情報として定義する。

2. ハイパーメディア型付加情報:

コンテンツ利用者への情報提供を目的として、配信された映像と同期して付加情報を配信し、登場するオブジェクトの動きにあわせて映像上に重ねて表示する。同期精度は映像と実時間で

の同期が必要な付加情報として定義する。

本研究で対象とする映像と付加情報の同期配信方式の設計にあたり、そのシステム要件について次の通り設定した。

1. 映像の形式や配信方式に依存しない同期方式であること:
インターネット上での映像配信サービスでは、映像形式としてライブ映像、蓄積映像、また、配信方式としては放送型配信、オンデマンド型配信など複数の形式、配信方式が存在する。このため、提案する同期方式は、映像の形式や配信方式に依存しない方式とする必要がある。
2. 付加情報の使用目的にあわせた同期精度が設定可能なこと:
映像と同期して配信する付加情報はその使用目的によって必要な同期精度が異なる。例えば、映像のシーンに連動した付加情報による字幕の表示は、付加情報の表示の開始点での同期が取れていれば十分であるが、映像に登場するオブジェクトの動きにあわせて付加情報を表示する場合は、映像の再生にあわせて付加情報の位置や大きさを実時間で変えて表示する必要がある。

5.3 関連研究

従来から映像に登場するオブジェクトを指定することにより、関連する情報を検索することが可能な動画ハイパーメディアシステムの研究は活発に行われてきた。しかし、従来の動画ハイパーメディアシステムの研究では、映像の形式や配信方式に依存しない映像と付加情報の同期方式に着目した研究はほとんど実施されていない。

池田らの開発したVHM[46]は、観光案内システムなどの映像からのハイパーリンクを特徴としたマルチメディアアプリケーションを開発するための開発環境であるが、その実行環境がマルチメディアデータベースを前提としており、その配信環境については言及していない。

高田らの開発したCmew[81]は、MPEG-1/SYSTEM プライベート 2 パケット中にリンク情報を埋め込み、映像からのハイパーリンクを実現しているが、現状、Cmewでは、映像はストリーム配信で無くダウンロード配信を前提としているため、配信方式が限定されている。

平田らの開発した緋[29]は、映像からのハイパーリンク機能を特徴としたマルチメディア情報検索システムであるが、緋では、映像はメディアサーバに蓄積され、オンデマンド配信される方式であり、配信方式が限定されている。

また、ライブ映像に対するハイパーリンク機能については、次の研究が実施されている。

有川らの開発したName-it[104]は、遠隔地に設置された定点観測カメラの映像をハイパーメディア化するシステムであり、ある映像に対するカメラの状態パラメータとオブジェクトのラベルを空間的に関連付けることを特徴としておりその同期方式は映像の配信方式には依存しないが、対象映像が定点観測カメラの映像であるため同期精度としては開始点同期程度を想定しており、実時間同期などのより同期精度の高い同期については考慮されていない。

また、ネットワーク環境で複数のメディア間の同期方式に関する研究は数多く行われてきた。

SMIL[100]は、XMLに準拠したタグ言語で、音声やビデオ、静止画、テキスト、アニメーションなどの複数のメディアを組み合わせたマルチメディアプレゼンテーションを記述することができるが、付加情報との同期情報として時刻情報をスクリプトに記述する方式であるため、映像中に時刻情報を持たない放送型の配信には対応できない。

ネットワーク上のメディア間同期で実時間同期に関する研究としては、動画と音声のリップシンクに関する研究がある。現在主流な方式としては、動画も音声も受信側で時間的構造を再構築できるよ

うにタイムスタンプやシーケンス番号を持つ RTP(Real-time Transport Protocol)[124] で配信し、受信側で復元した時間的構造を用いてメディア同期を行う方式がある。この方式は実時間同期に向いているが、同期を必要とするメディアはすべて RTP で配信する必要があり、本研究で想定している同期方式には適用できない。

佐藤らは、映像の形式や配信方式に依存しないメディア同期を考慮したマルチメディア転送プロトコル [125] について研究している。この方式では、特に動画と音声の実時間同期を対象として、メディア間で相対的に同期を行う位置を同期ポイントとし Strict Synchronization, Relaxed Synchronization, Silence Detected Synchronization の 3 種類の同期方式を提案しているが、この方式は動画と音声の実時間同期を想定した方式であり、そのままでは本研究で想定している同期方式には適用できない。

端山らの開発した協調型同期方式に基づくプロセス間同期支援システム NAMI (Network-oriented Administrator for Multimedia Integration)[126] は、同期させたい複数のメディアに対して任意の同期点を設定することによってメディア間の同期を実現するシステムであり、対象メディアの特徴にあわせて同期点の同期精度を変更できることを特徴としている。NAMI システムでは、同期情報を管理する情報サーバと各メディアの処理を行うプロセスにて同期処理を支援する同期マネージャの協調処理に基づいて同期処理を行う。NAMI システムでは、プロセス外部に設置した情報サーバにて同期情報を一元管理する方式のため、映像配信サービスへの適用を想定した場合、マルチキャスト配信などを用いた放送型配信時などの再生側のプロセスが増えた場合などに、情報サーバのスケラビリティが課題となるため、本研究の同期方式には適用できない。

5.4 拡張イベントモデルの設計

5.2 で設定したシステム要件と、5.3 の関連研究の結果に基づいて映像と付加情報の同期配信を目的とした拡張イベントモデルの設計を行う。まず、設計指針を決め、その設計指針に基づいてデータモデル及び同期配信方式の設計を行う。

5.4.1 設計指針

5.2 で設定したシステム要件、及び 5.3 の関連研究の結果に基づいて、本研究における映像と付加情報の同期配信方式の設計指針として映像に多重化するイベントを拡張する方式を提案する。

イベントは、映像中の任意の位置に多重化する方式であるため映像の形式や配信方式に依存しない。映像と付加情報の同期配信に必要な同期情報をイベントとして映像中に多重化することにより同期を実現する。映像中に同期情報を多重化するため、映像をマルチキャスト配信などを用いて放送型配信した場合でも、付加情報との同期が可能となる。

また、イベントを拡張して付加情報の使用目的にあわせて周期的に多重化することにより、映像と付加情報の開始点同期や実時間同期を実現することができる。

5.4.2 データモデルの設計

5.4.1 で設定した設計指針に基づき、映像と同期して配信する付加情報と関連情報のデータモデルについて設計する。図 5.1 に本方式における付加情報と関連情報のデータモデルについて示す。

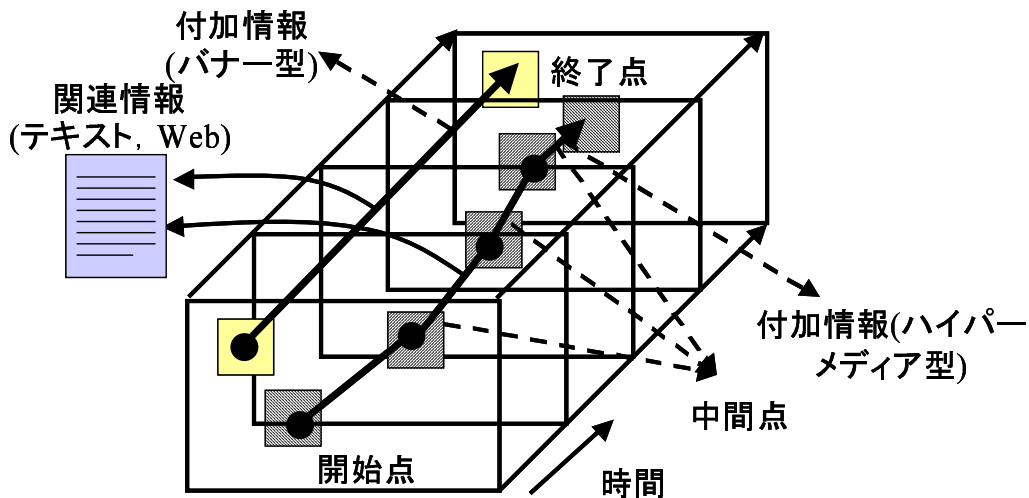


図 5.1: 付加情報/関連情報データモデル

バナー型付加情報

バナー型付加情報は、映像との開始点同期が必要な付加情報として、配信された映像と同期して、開始点から終了点まで位置や大きさが固定の付加情報として定義する。種類は「イメージ」「テキスト」の2種類とする。

バナー型付加情報は、映像と連動したスポンサーなどの広告表示や字幕サービスなどに利用することができる。

ハイパーメディア型付加情報

ハイパーメディア型付加情報は、映像との実時間同期が必要な付加情報として、配信された映像と同期して、対象とするオブジェクトの動きにあわせて、その位置や大きさが可変の付加情報として定義する。種類は「マーク」の1種類とし、基本的には映像上に登場するオブジェクト上にオブジェクトを囲む形で表示される。

ハイパーメディア型付加情報は映像に対するコンテンツ利用者への情報提供に利用することができる。

ハイパーメディア型付加情報のデータモデルは、基本的には、第4章で定義したアンカーモデルに基づいたデータモデルとする。本モデルの特徴は、全フレームに対して位置や大きさのデータを保持せず、開始点から終了点までの区間全体を幾つかの中間点で定義し、それらの点にデータを持たせるモデルとする。データを持たないフレーム(標準フレーム)は前後のデータを持つフレーム(基準フレーム)から補間計算によって求めるモデルとする。

本データモデルにおいて、映像上に登場するオブジェクトの動きにあわせて付加情報を設定する場合、全フレームにデータを設定せずにオブジェクトの動きが大きく変わる点のみ基準フレームとしてデータを設定することにより、付加情報の設定作業を簡易化することができる。

関連情報

関連情報は、付加情報をクリックすることによりコンテンツ利用者に提供されるアクションとして定義する。本モデルで定義した関連情報の形式としては「Web」と「テキスト」の2種類とし「Web」の場合は、検索した関連情報は URL 情報として外部の Web ブラウザに渡され、Web コンテンツが表示される。「テキスト」の場合は関連情報をテキスト情報として映像上に表示する。

5.4.3 同期配信方式の設計

5.4.1 で設定した設計指針に基づき、映像と付加情報の同期配信を目的とした拡張イベントモデルにおける同期配信方式の設計を行う [127]。

図 5.2 に提案した拡張イベントモデルにおける同期配信方式の概要について示す。

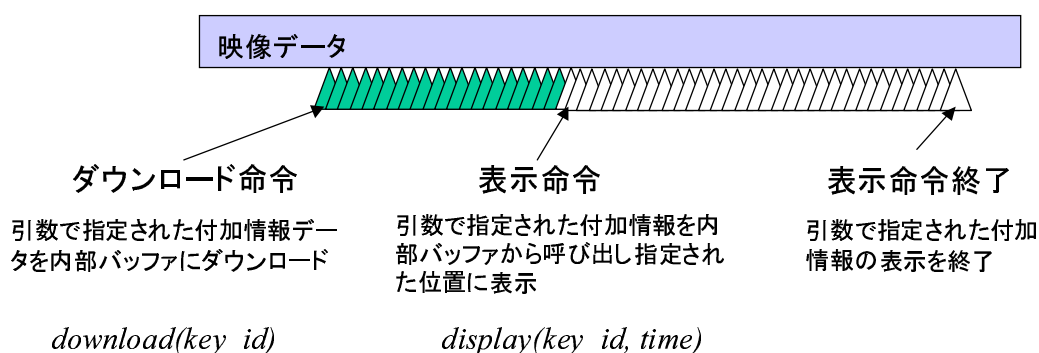


図 5.2: 拡張イベントモデルにおける同期配信の概要

提案した同期配信方式は、映像中に付加情報との同期配信の実現のための同期情報を周期的にイベントとして多重化する方式とする。通常、インターネット上の映像配信サービスでは、映像の配信は UDP/IP ベースのコネクションレス型の配信が用いられるが、映像中に同期情報を周期的に多重化することにより、パケット損失に対応することができる。

また、本方式は映像中には同期に必要な最低限の情報を同期情報として多重化する方式であり、付加情報は映像中に多重化せずに HTTP などの他の方式を用いて配信する方式とする。

本方式で使用する同期情報について次の通り定義する。

1. ダウンロード命令 *download(key_id)*:
再生ソフトウェアでこの命令を受信すると、内部メモリ上の付加情報をチェックし、引数 *key_id* で指定した付加情報が内部メモリ上に存在しない場合はネットワーク経由で付加情報をダウンロードする。表示命令の発行前にダウンロード命令を発行しておくことにより、映像と付加情報の開始点同期を実現する。
2. 表示命令 *display(key_id, time)*:
再生ソフトウェアでこの命令を受信すると、引数 *key_id* で指定された付加情報を内部メモリから呼び出して再生中の映像上の指定された位置に表示する。指定された付加情報が内部メモリに存在しない場合はネットワーク経由でダウンロードしてから表示する。

また、引数 *time* には表示命令の開始からの経過時間を指定する。例えば、ハイパーメディア型の付加情報の表示を行う際には、その経過時間にあわせて付加情報の位置、大きさなどを算出する。表示命令を周期的に発行することにより映像と付加情報の実時間同期を実現する。

本方式の特徴について次にまとめる。

- イベント拡張型であり、映像の形式や配信方式に依存しない同期方式である
- 表示命令の発行前にダウンロード命令を発行することにより、映像と付加情報の開始点同期を実現
- 周期的に表示命令を発行することにより、映像と付加情報の実時間同期を実現
- 周期的に同期情報を多重化することにより、パケット損失などが起こりうるコネクションレス型の映像配信に対応可能

また、図 5.3 に本方式を用いた映像と付加情報の同期配信の流れについて示す。

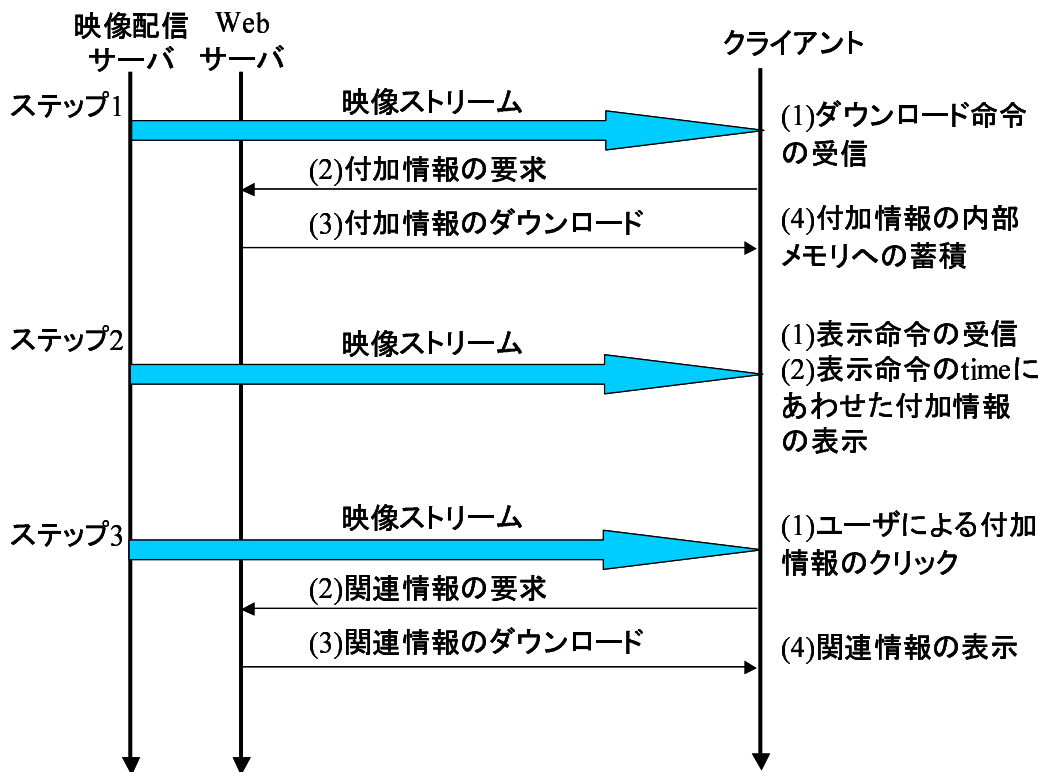


図 5.3: 映像と付加情報の同期配信の流れ

1. 付加情報のダウンロード (ステップ 1):
再生ソフトウェアで映像中に多重化されたダウンロード命令を受信すると、ネットワーク経由で付加情報をダウンロードし内部メモリに持つ。既に内部メモリに該当する付加情報を保持している場合は、何も行わない。

表 5.1: Windows Media のイベント情報

データ構造	データ形式	内容
iIndex	short	イベントのインデックス
bstrType	BSTR	イベントの種別
bstrData	BSTR	イベントの内容

2. 付加情報の同期表示 (ステップ 2):

再生ソフトウェアで映像中に多重化された表示命令を受信すると、内部メモリの付加情報を経過時間から表示位置、大きさを算出して映像上に重ねて表示する。
 実時間同期の必要なハイパーメディア型付加情報の場合は、この命令を受信する毎に付加情報の表示の更新を行う。

3. 関連情報の表示 (ステップ 3):

再生ソフトウェアで表示された付加情報をコンテンツ利用者がクリックした場合、関連情報をネットワーク経由で受信して表示処理を行う。

提案した拡張イベントモデルは対応する映像配信プラットフォームにおいてエンコード/デコードライブラリとして実装され、エンコードライブラリは映像と付加情報の関連付けを行うオーサリングソフトウェアに、デコードライブラリは再生ソフトウェアに適用することにより、映像と付加情報の同期配信を実現することができる。

5.5 拡張イベントモデルの実装

5.4 で提案した拡張イベントモデルを、現在インターネットにおける映像配信プラットフォームとして広く普及している Windows Media と、デジタル放送で採用され今後インターネット上でも普及する可能性の高い MPEG-2 TS を対象としてエンコード/デコードライブラリとして実装した。

通常、イベント処理は CPU 負荷に大きく影響するため、各プラットフォームにおいてその多重化精度の最小値が規定されている。この値は付加情報の実時間同期性能に大きく影響するため、実装にあたっては、その数値を考慮する形で実装し、評価システムを構築してその実時間性能について評価実験を行う方針とした。

5.5.1 Windows Media 版の実装

Windows Media の場合、通常のビデオ、オーディオに加えて、スクリプト領域が用意されており、このスクリプト領域に対してユーザ定義のイベントを定義することが可能である。Windows Media におけるイベント情報について表 5.1 に示す。

Windows Media プラットフォームで推奨されているイベントの多重化精度の最小値は 100 msec である。そこで、イベントの多重化精度の最小値を 100 msec としてエンコード/デコードライブラリを実装し、評価実験で実装したライブラリの実時間同期性能について評価することにした。

表 5.1 に従い、キャプチャカードから入力されたライブ映像または蓄積された AVI 形式の映像を入力し、入力された映像の Windows Media 形式へのエンコード処理を行う際に、周期的に同期情報

表 5.2: MPEG-2 TS におけるイベント情報

データ構造	ビット列表記	内容
event_msg_group_id	uimsbf	メッセージグループ識別子
time_mode	uimsbf	時間モード
event_msg_type	uimsbf	メッセージ種別
event_msg_id	uimsbf	メッセージの識別子
private_data_byte	uimsbf	プライベートデータ

をユーザ定義のイベントとしてスクリプト領域に多重化する機能を持つエンコードライブラリを実装した。

デコードライブラリは、Windows Media 形式の映像を受信、表示を行うとともに、多重化された同期情報を分離、解釈し、必要に応じて付加情報を配信サーバから HTTP を用いてダウンロードし、同期表示する。

また、通常の Windows Media Player で同期情報の多重化された Windows Media 形式の映像を受信した場合は、同期情報はユーザ定義のイベントとして扱われるため、映像のみ再生される。

5.5.2 MPEG-2 TS 版の実装

デジタル放送で採用されている MPEG-2 TS にはイベントメッセージ伝送方式が規定されており、その一つである汎用イベントメッセージ記述子にユーザ定義のイベントを定義して使用することができる。そこで映像と付加情報の同期配信に必要な同期情報をイベントとして多重化して配信することにより、拡張イベントモデルの適用が可能となる。表 5.2 に MPEG-2 TS におけるイベント情報について示す。

また、イベントの多重化精度の最小値は Windows Media の場合と同等である 100 msec が推奨されており [128]、MPEG-2 TS 版の場合も最小値を 100 msec としてエンコード/デコードライブラリを実装した。

MPEG-2 TS へのイベントの多重化処理はリアルタイム処理を行う場合は通常専用の多重化装置を必要とするが、今回は、蓄積された MPEG-2 TS データを読み出して、表 5.2 に示したイベント情報を周期的に多重化する機能を持つエンコードライブラリをソフトウェアとして実装した。

デコードライブラリは、Windows Media 版と同様 MPEG-2 TS 形式の映像を受信、表示を行うとともに、多重化された同期情報を分離、解釈し、必要に応じて付加情報を配信サーバから HTTP を用いてダウンロードし、同期表示する機能を持つ。

5.6 評価システムの構築

5.6.1 評価システムの構成

5.5 で実装した拡張イベントモデルのエンコード/デコードライブラリを使用して、映像と付加情報の同期配信を行う評価システムを構築した。評価システムのベースとする映像配信プラットフォーム

としては、現在インターネット上で広く利用されている Windows Media を選択した。

図 5.4 に構築した評価システムの構成について示す。

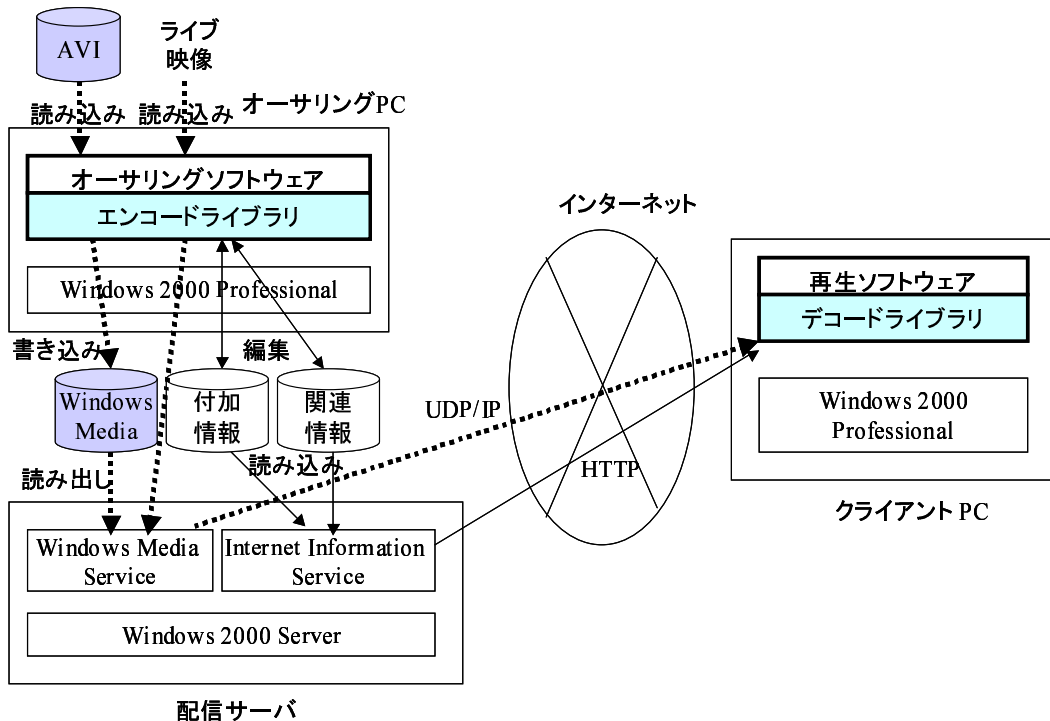


図 5.4: 評価システムの構成

オーサリング環境

オーサリング環境としては、付加情報、関連情報の編集および映像のエンコードを行うオーサリング PC を設置した。

オーサリング PC は、CPU が Pentium4(2GHz)、メモリを 1GB 搭載した IBM PC/AT 互換機で、OS は Windows 2000 Professional を使用し、ライブ形式または AVI 形式の映像を読み込み、付加情報を画面上に表示させその位置、大きさや関連情報の編集を行い、Windows Media 版のエンコードライブラリを用いてエンコードを行う際にあわせて同期情報の多重化を行うオーサリングソフトウェアを構築した。

オーサリングソフトウェアで編集した Windows Media 形式の映像、付加情報データ、関連情報データは配信サーバに登録する。

図 5.5 に構築したオーサリングソフトウェアの画面イメージについて示す。

5.6.2 配信環境

配信環境としては、映像及び付加情報、関連情報をインターネットを介してクライアント PC に配信を行う配信サーバを設置した。



図 5.5: オーサリングソフトウェアの画面イメージ

配信サーバは，CPU が Pentium3(1.26GHz)，メモリを 2GB 搭載した IBM PC/AT 互換機で，OS は Windows 2000 Server を使用し，映像，付加情報，関連情報の配信はすべて OS 付属のサービスを利用する．

5.6.3 再生環境

再生環境としては，インターネット経由で映像や付加情報，関連情報の受信，表示を行うクライアント PC を設置した．

クライアント PC は，CPU が Pentium4(2GHz)，メモリを 1GB 搭載した IBM PC/AT 互換機で，OS は Windows 2000 Professional を使用し，Windows Media 版のデコードライブラリを用いて配信サーバから Windows Media 形式の映像を受信し，多重化された同期情報に基づいて配信サーバから付加情報のダウンロード及び同期表示を行う再生ソフトウェアを構築した．図 5.6 に構築した再生ソフトウェアの画面イメージについて示す．

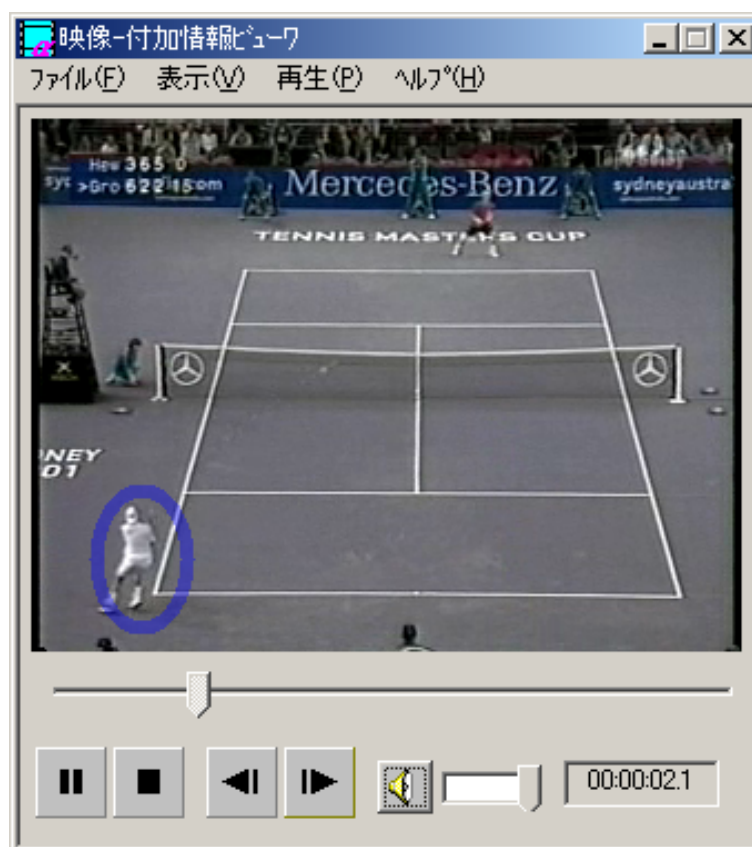


図 5.6: 再生ソフトウェアの画面イメージ

図 5.6 において，コンテンツ利用者が映像上に表示された付加情報をクリックすると，関連情報を検索して表示する．

表 5.3: 評価用データの形式

種別	形式	フレームレート	画面サイズ
データ 1	AVI	30 フレーム/秒	320 × 240
データ 2	AVI	29.97 フレーム/秒	720 × 480

表 5.4: 評価用データの概要

種別	ショットの概要	ショット数	平均移動速度
データ 1	矩形オブジェクト (等速移動)	1	30 ドット/秒
		1	60 ドット/秒
		1	90 ドット/秒
		1	120 ドット/秒
データ 2	野球：攻撃	5	11.59 ドット/秒
	野球：守備	5	37.72 ドット/秒
	野球：走塁	5	61.09 ドット/秒

5.7 評価実験

5.7.1 評価方針と評価用データ

5.6 で構築した評価システムの実時間同期性能の評価を目的として以下の評価実験を行った。

5.5 でも言及した通り、Windows Media または MPEG-2 TS に対して本方式を実装した場合、イベントの多重化精度の最小値が 100 msec となる。本評価実験では、構築した評価システムにおける映像と付加情報の実時間同期性能の評価を目的とする。

評価実験に際して、実時間同期性能の評価を目的としたハイパーメディア型の付加情報の評価用データとして次の 2 種類のデータを準備した。以下、表 5.3 に評価用データの形式について、表 4 に評価用データの概要について示す。

評価用データの内容は、データ 1 が基本性能の評価用データで、データ 2 が評価システムのスポーツ映像の配信サービスへの適用を想定した評価用データである。

5.7.2 同期配信方式の評価

基本性能評価

オーサリングソフトウェアを用いて、データ 1 を使用して平均移動速度の異なるオブジェクトに対して、同期情報の多重化精度を 100, 200, 500 msec と変更してハイパーメディア型付加情報の設定を行った評価用データを準備した。また、エンコード時の圧縮パラメータとしては、表 5.5 に示したパラメータを使用した。

再生ソフトウェアを用いて、5 名のユーザに対してオブジェクトの平均移動速度の異なる複数の評価用データを配信サーバからネットワーク経由で配信し、表示性能をユーザの満足度として評価して

表 5.5: 評価用の圧縮パラメータ

圧縮形式	圧縮レート	フレームレート	画面サイズ
Windows Media Video/Audio V8	1 Mbps	30 フレーム/秒	320 × 240

もらう実験を行った．その結果を図 5.7 に示す．

ユーザの満足度の算出方法は，再生ソフトウェアで映像と付加情報の同期表示を行い，その表示性能について対象オブジェクトと付加情報のずれなどを元に「オブジェクトと付加情報の表示がずれている:0」、「オブジェクトと付加情報の表示がずれていない:1」、「1と0の間:0.5」の3段階の数値データで評価を行い，全ユーザの平均値を評価結果とする方針とした．

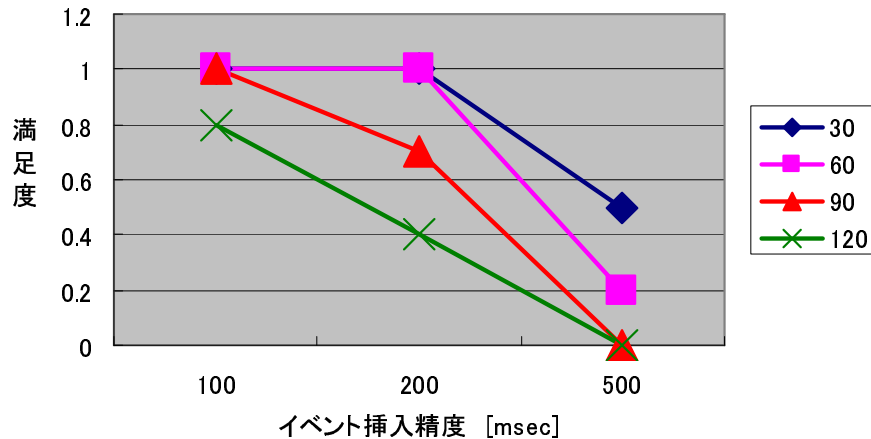


図 5.7: 基本性能評価結果

スポーツ映像に対する有効性評価

オーサリングソフトウェアを用いて，データ 2 を使用して同期情報の多重化精度を 100，200，500 msec と変更してハイパーメディア型付加情報の設定を行った評価用データを準備した．エンコード時の圧縮パラメータとしては，基本性能評価の際と同様に表 5.5 に示したパラメータを使用した．

再生ソフトウェアを用いて，基本性能評価実験と同じ 5 名のユーザに複数の評価用データを配信サーバからネットワーク経由で配信し，ユーザによる表示性能をユーザの満足度として評価してもらう実験を行った．その結果を図 5.8 に示す．

ユーザの満足度の算出方法は，基本性能評価の場合と同じ方式とした．

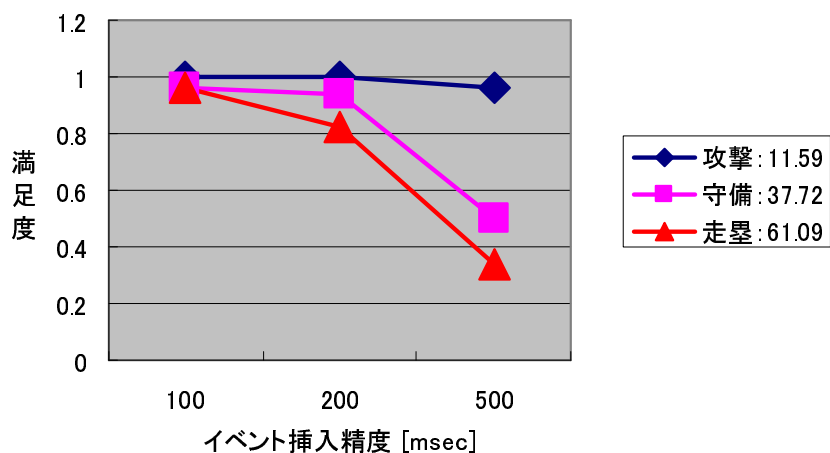


図 5.8: スポーツ映像に対する評価結果

5.8 考察

5.7 で実施した評価実験の結果に基づき、実装した拡張イベントモデルの実時間同期性能について考察する。

5.7 で実施した評価実験では、平均移動速度の異なるオブジェクトを持つ基本評価用データ及びスポーツ映像を対象としてオーサリングソフトウェアを使用してハイパーメディア型の付加情報の設定した異なる同期情報の多重化精度を持つ評価用の映像を準備し、再生ソフトウェアを利用して、その多重化精度と表示性能の関係についての評価を行った。

図 5.7 の結果を見ても分かる通り、本評価実験のユーザの満足度の目標値を 0.8 とした場合、同期情報の多重化精度が 100 msec あれば、対象とするオブジェクトの平均移動速度が最も速い 120 ドット/秒のオブジェクトに付加情報を設定して同期配信を行った場合でもその同期表示性能を満たしていることが分かった。一方、200msec、500msec の場合は、オブジェクトの移動速度が速くなるにつれて徐々に対応できなくなることが分かった。

今回の評価実験ではエンコードライブラリはエンコード時に表 5.5 に示した圧縮パラメータを使用しており、フレームレートが 30 フレーム/秒で画面サイズが 320 × 240 の映像表示画面を 2~3 秒程度で横切るオブジェクトに対しても、ほぼ 3 フレーム単位で同期情報を多重化することにより映像と付加情報の同期配信、表示が十分可能であることが分かった。

基本性能評価の結果を踏まえて、図 5.8 の結果を見たところ、本評価実験のユーザの満足度の目標値を 0.8 とした場合、こちらも同期情報の多重化精度が 100 msec あれば、対象とするオブジェクトの平均移動速度の最も速い走塁ショットに登場するオブジェクトに付加情報を設定して同期配信した場合でも、十分にその実時間での同期性能を満たしていることが分かった。

以上の理由から、同期情報の多重化精度が 100 msec あれば、基本性能としてはフレームレートが最大 30 フレーム/秒で画面サイズが 320 × 240 の映像に対して、平均移動速度が最大 120 ドット/秒程度のオブジェクトに設定した付加情報の同期配信が可能であることが分かった。また、通常のスポーツ映像に登場するオブジェクトの平均移動速度は最大 100 ドット/秒程度であることが予想され、本

評価システムをスポーツ映像を対象とした映像と付加情報の同期配信サービスに適用することが可能であることが分かった。

5.9 おわりに

第5章では、インターネット映像配信サービスへのハイパーリンク機能の追加を目的としてバナー型、ハイパーメディア付加情報の同期配信を対象サービスとしてシステム要件を検討し次の要件を設定した。

1. 映像の形式や配信方式に依存しない同期方式であること
2. 付加情報の使用目的にあわせた同期精度が設定可能なこと

システム要件に基づいて拡張イベントモデルを提案，設計し，Windows Media 及び MPEG-2 TS を対象プラットフォームとしてエンコード/デコードライブラリとして実装し，Windows Media 版ライブラリを元に評価システムを構築し，その実時間同期性能について評価実験を行った。

評価実験結果について考察を行った結果，対象とする映像配信プラットフォームにおいてイベントの多重化精度が現状の 100 msec あれば，スポーツ映像を対象とした映像と付加情報の同期配信サービスに適用可能であることが検証できた。

今後は，MPEG-2 TS 版ライブラリを用いた評価システムを構築するとともに，構築したシステムの実用化を行っていく予定である。

第6章 初等教育への応用を目的とした動画ハイパーメディアシステム

第6章では、動画ハイパーメディアシステムの応用として、動画ハイパーメディアシステムの初等教育への応用を目的として、初等教育への応用を目的とした動画ハイパーメディアシステムの設計と構築を行う。さらに、構築した動画ハイパーメディアシステムを、実際に初等教育への応用を対象として評価実験を行った結果について述べる。

6.1 はじめに

計算機やネットワーク等のハードウェアの性能向上やオペレーティングシステム等のソフトウェア技術の進歩により、計算機上で音声や映像等のマルチメディア情報を扱うことが現実的となってきた。また、WWWの登場と爆発的な普及により、世界中のサーバ上で公開されている情報がリンクによって相互に接続されるようになった。

このような背景のもと、小中学校等の教育現場においても、文部科学省、総務省をはじめとする中央省庁の積極的な政策により、2001年までに全国の全公立小中高校がインターネットに接続される計画であり、WWWなどのインターネット技術を活用した教育のためのインフラが整いつつある。

小中学校等の教育現場では、文部科学省、総務省をはじめとした中央省庁が、全国の小中学校に対するインターネットへの接続に関する積極的な政策を実施しており、文部科学省の調査では、2004年3月31日現在で、全国の22,913校ある小学校のうち99.7%の22,837校が、10,292校ある中学校のうち99.9%の10,282校が、インターネットに接続されており、このうち小学校の68.0%の15,519校が、中学校の72.4%にあたる7,445校が接続回線速度400kbps以上の高速インターネットで接続されており、WWWなどのインターネット技術を活用した教育に利用されている[9,10]。

現在の教育の現場では、テレビやビデオをはじめとした映像教材がよく使用されているが、集合教育となりどうしても全員の進捗度をあわせる必要があり、教育効果が上がらないという問題がある。一方、WWWは生徒の自主性や個人の進捗度にあわせて学習が行える利点があり、2002年度より全国の小中学校にて本格的に実施されている総合的な学習の時間でも、積極的に活用されている。

教育現場での利用を目的としたWWWへの動画データの適用について着目した場合、現状のWWWでは、リンク先の情報として動画データを指定し、コンテンツ利用者がそのリンクを呼び出すことによって、リンクされた動画データを再生することは当たり前になってきている。しかし、動画データをリンク元にして、動画データに対して設定されたアンカーをクリックしてリンクされた情報を検索する動画ハイパーメディアシステムはほとんど普及していない。

そこで、本研究では、教育現場での利用を目的とした動画ハイパーメディアシステム VisualSHOCK MOVIE を構築し、初等教育に適用してシステムの有効性についての評価を行う。構築した動画ハイパーメディアシステム VisualSHOCK MOVIE は、AVI、QuickTime Movie、MPEG-1等の動画データの読み込み、アンカー、イベント、マーク等のインタラクティブ情報の追加・編集、及び編集し

たインタラクティブ動画の RealMedia 等のストリーム形式の動画データへの変換，VisualSHOCK MOVIE Player による Web ブラウザ上での再生，といった枠組みを提供するものである．以下，6.2 でシステム要件及び関連研究，6.3 で VisualSHOCK MOVIE の設計と構築，6.4 で初等教育への応用及び評価実験，6.5 でまとめを行う．

6.2 システム要件及び関連研究

6.2.1 システム要件

本研究では，初等教育への応用を想定した動画ハイパーメディアシステム VisualSHOCK MOVIE の設計及び構築を行う．まず，VisualSHOCK MOVIE の設計にあたり，前提とするシステム要件を次の通り定めた．

1. インタラクティブ動画による教育効果向上:
動画データに対してマルチメディア教育に必要なインタラクティブ情報の追加ができること．
2. 編集作業の簡易化:
計算機の扱いに不慣れな教師でも簡単に動画ハイパーメディアコンテンツの編集作業が行えること．
3. Web ブラウザ上での利用:
制作した動画ハイパーメディアコンテンツは，教育現場で通常使用している Web ブラウザ上で動作すること．Web ブラウザは総合的な学習の時間においても，生徒が自主的に「調べる」学習を行う際に積極的に活用されている．
4. ストリーミング配信への対応:
制作した動画ハイパーメディアコンテンツは，ローカルの CD-ROM や DVD-ROM だけでなく，学校内 LAN や，インターネットなどのネットワーク経由での利用も可能なこと．

6.2.2 関連研究

動画ハイパーメディアシステムに関する研究は現在まで活発に行われている．例えば，坂田らが開発した VHM は映像を利用したウォークスルーを目的としたシステムとなっている [42]．VHM では，映像や音声のデータをカット毎に分割して蓄積し，再生される際の順序と条件をシナリオとして記述しておき，映像データが分岐点まで再生されると，コンテンツ利用者にどちらに進むか促すダイアログを表示し，選択結果に応じてリンク先の動画の再生を行うシステムである．VHM でのリンクは主に動画から動画へのリンクを対象としているため，Web ブラウザ上の利用は考慮されていない．

また，田中らは，動画内に登場するオブジェクトに対するアンカー，リンク設定，編集を行うためのアーキテクチャについて提案した [49]．このアプローチにより動画に登場するオブジェクトに付加されたリンク検索が可能になるが，このアーキテクチャはスタンドアロンシステムを対象としており，Web ブラウザ上の利用は考慮されていない．

藤本らは，インタラクティブ動画のモデル化を XML 形式で行い，インタラクティブ性を付加するためのオーサリングツールを開発した [51]．このアプローチで提案されている CMML では，既存の動画からクリッカブルムービーを編集することができる．しかし，現状では，アンカーの動きが直線的な一回の移動しかサポートされておらず複雑な動きに対応できない，再生は専用のソフトウェアを

使用しリンク先は bitmap 形式のイメージデータしか表示できない，扱える動画形式が AVI 形式しか扱えない，などの問題がある．

高田らは WWW との親和性を前提とした研究を行っている [81]．この研究では，連続メディアデータ内または連続メディアデータ外に対して URL 等のインターネット情報へのリンク情報または Java アプレットなどのプログラムをリンク情報として付加することができる．しかし，現状では，リンクの起点となるアンカーの動きが考慮されていない，ストリーミング配信に対応していない，などの問題がある．

6.3 VisualSHOCK MOVIE の設計及び構築

6.3.1 設計指針

6.2 で設定したシステム要件及び関連研究の考察結果をもとに，VisualSHOCK MOVIE の設計指針について次の通り定めた．

1. マルチメディア初等教育に必要な動画データに追加可能なインタラクティブ情報についての検討を行い，その結果に基づいてモデル化を行う．
2. 計算機に不慣れな教師が使用することを前提条件として，動画像処理によるオブジェクト追跡処理と手作業による修正作業を組み合わせることによって，アンカー設定作業の効率化を図る．
3. 制作した動画ハイパーメディアコンテンツは，通常教育現場で使用している Web ブラウザ上での利用を前提とする．
4. 制作した動画ハイパーメディアコンテンツは，同じソースからローカルの CD-ROM や DVD-ROM で利用できる形式と，LAN 環境やインターネット環境などのネットワーク経由で利用できる形式の両者が制作可能であること．

6.3.2 インタラクティブ動画オブジェクトモデル

6.3.1 で設定した設計指針に従い，マルチメディア初等教育に必要なインタラクティブ動画の要件について検討した．

1. 動画に登場するオブジェクトの動き，形状の変化に追従してアンカー領域を設定し，コンテンツ利用者の指示により設定されたアクションが実行されること．
2. コンテンツ利用者の指示ではなく，コンテンツ制作者による制作時に設定することにより，動画の再生があるフレームに到達すると自動的に設定されたアクションが実行されること．
3. 動画の任意のフレームに名前を付けることにより，フレーム番号ではなく名前で検索が行えること．

次に，上記要件に基づき，VisualSHOCK MOVIE におけるインタラクティブ動画オブジェクトのモデル化を行った．図 6.1 に VisualSHOCK MOVIE におけるインタラクティブ動画オブジェクトモデルの概要について示す．

インタラクティブ動画オブジェクトモデルの持つインタラクティブ情報について説明する．

1. アンカー：
動画中に登場するオブジェクトの動きや形状の変化に追従したリンクの起点となる領域．アン

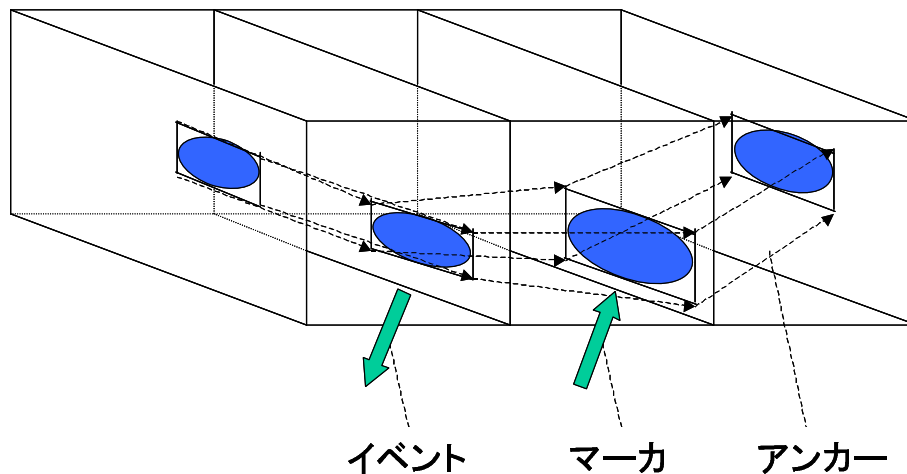


図 6.1: VisualSHOCK MOVIE のインタラクティブ動画オブジェクトモデル

カーを使用することによって、動画に登場するオブジェクトに対するリンクを実現することができ、コンテンツ利用者の指示によりアンカーにリンク情報として設定されたアクションを実行することができる。

2. イベント:

動画中の任意のフレームに設定し、動画の再生がイベントを設定したフレームに到達すると、イベントに設定されたアクションを自動的に実行することができる。

3. マーカ:

動画中の任意フレームに対して設定することができる。外部アプリケーションからマーカを呼び出すことにより、外部アプリケーションから任意のフレームの頭出し再生を行うことができる。

上記インタラクティブ情報のうちアンカーについては、第4章で提案したアンカーモデルに準拠する方針とした。

次に、6.3.1にて設定した設計指針に従い、VisualSHOCK MOVIE で扱う動画データの形式について表6.1の通り定めた。基本的な考え方は、ファイル保存の形式の動画データを読み込み、アンカー、イベント、マーカ等の編集作業を行ったあと、CD-ROMやDVD-ROMなどで使用する場合はそのまま、LAN環境やインターネット環境などのネットワーク経由で使用する場合はストリーム形式に変換することにより、コンテンツ制作作業を一度行えば、その編集結果を使用する環境によって使い分けることが可能となる。

また、動画データに対するアンカー、イベント、マーカなどのインタラクティブ情報の設定方式について次の通り検討を行った。その結果について表6.2に示す。

インタラクティブ情報の設定方式の基本的な考え方は、ファイル保存の形式の動画データであるAVI形式、QuickTime Movie形式、MPEG-1形式については、編集作業の簡易化を考慮して動画データと別に保存する形式とした。

また、ストリーム形式の動画データである、RealMedia形式については、通常はファイル保存の形式の動画データからの変換処理が必要であり、今回は、AVI形式、QuickTime Movie形式の動画

表 6.1: VisualSHOCK MOVIE が対応する動画形式

ファイル保存の形式	ストリーム形式
AVI 形式	RealMedia 形式
QuickTime Movie 形式	
MPEG-1 形式	

表 6.2: インタラクティブ情報の設定方式

	設定方式		インタラクティブ情報		
	データ外	データ内	アンカー	イベント	マーカ
AVI 形式		×			
QuickTime Movie 形式		×			
MPEG-1 形式		×			
RealMedia 形式	×				×

データからの変換時に、2.3.3 で説明した内容に基づいて、RealMedia 形式の動画データ中にイメージマップ、イベントとして記述する方式とした。

6.3.3 インタラクティブ動画オブジェクトの出力形式

6.3.2 のインタラクティブ動画オブジェクトモデルに基づいて、動画データに対してアンカー、イベント、マーカなどのインタラクティブ情報を追加したインタラクティブ動画オブジェクトの出力形式について設計を行った。

図 6.2 に 6.3.2 でのモデル化に基づいて定義した VisualSHOCK MOVIE におけるインタラクティブ動画オブジェクトの出力形式例について示す。

図 6.2 において、イベント定義は [Event] に記述し、

イベント名
 設定されたフレーム番号
 イベント起動時のアクション
 イベントを通知するフレーム名

を:で区切った形式とする。

同様に、マーカ定義は [Marker] に記述し、

マーカ名
 設定されたフレーム番号

を:で区切った形式とする。

一方、アンカー定義は、動き、形状の変化:[HotSpot]、クリック時のアクション: [Link]、視覚効果:[Effect] の 3 つの領域に分けて行う。

```

[Event]
Event1:39:URL:'http://www.visualshockcom/','EventInfo'

```

← イベント定義

```

[Marker]
Marker1:24

```

← マーカ定義

```

[HotSpot]
<AncID1:AnchorName1:R:0-64:10>
0:172,87,245,160
10:162,77,245,154
15:178,82,266,161
20:197,95,291,176
25:199,99,298,182
30:182,106,286,191
35:171,97,281,184
45:144,93,264,184
55:91,105,222,200
64:44,116,185,216

<AncID2:AnchorName2:C:8-73:5>
8:163,84,243,164
24:201,96,296,191
48:129,82,247,200
62:73,112,175,214
73:38,104,153,219

```

← アンカー定義1
(動き、形状の変化)

```

[Link]
AncID1:URL:'http://www.melco.co.jp/','AnchorInfo':"
AncID2:SEEK:41:"

```

← アンカー定義2
(クリック時のアクション)

```

[Effect]
AncID1:0x000000ff:3:0x00ff0000:1
AncID2:0x000000ff:3:0x00ff0000:1

```

← アンカー定義3
(視覚効果)

図 6.2: VisualSHOCK MOVIE のインタラクティブ動画オブジェクトの出力形式例

[HotSpot] はアンカーの動き，形状の変化について定義する．アンカーの定義は，< > で囲まれた領域に，

アンカー ID
アンカー名
アンカー形状 (R: 矩形, C: 円形)
開始フレーム - 終了フレーム
基準フレームの数

を記述し，その下に基準フレームの数だけ，アンカーの座標データを記述する．

[Link] はアンカーに対するリンク情報をクリック時のアクションとして記述する．ここで定義できるアクションは，

URL：指定された URL を呼び出す
SEEK：同じ動画データの指定されたフレームにジャンプする
PLAYER：別の動画データを呼び出す

である．

[Effect] はアンカーの表示情報について記述し，通常時の表示色，クリック時の表示色などの指定を行う．

[HotSpot] の定義から分かる通り VisualSHOCK MOVIE では，編集作業の簡単化を目的として，アンカーの存在区間の全フレームに対してアンカーの位置情報を持たせるのではなく，基準フレームにのみ位置情報を持たせ，位置情報を持っていないフレームは，前後の基準フレームから補間計算を行うことにより，アンカーの位置を推定する方式を採用した．

6.3.4 インタラクティブ動画オブジェクトの編集

6.3.3 で設計したインタラクティブ動画オブジェクトを編集するソフトウェアである Movie MapEditor の設計と構築について述べる．

Movie MapEditor の設計指針としては，6.3.1 で設定した設計指針に基づき，計算機に不慣れな教師をコンテンツ制作者に想定して，アンカー編集作業の簡易化を目的として，第 4 章で提案したアンカー設定方式を適用する方針とした．本方式では，動画像解析を利用したオブジェクトの自動追跡を行うとともに，手動による修正作業を組み合わせることにより，編集作業のうち特に手間のかかるアンカー設定作業の簡易化を図ることができる．

図 6.3 に構築した Movie MapEditor の画面イメージについて示す．

次に，Movie MapEditor におけるアンカー設定処理の流れについて示す．

1. 開始点，終了点の設定:
画面上でアンカーの開始フレーム，終了フレームおける位置，大きさを設定する．この操作は，動画を開始フレームまたは終了フレームに位置付け，表示されたオブジェクトに重ねて矩形または円形のアンカーを描くことにより行う．
2. 動画像解析を用いた自動追跡:
 1. を行った後，コンテンツ制作者がアンカーの自動設定処理を選択すると，システムは動画像解析を行い，その結果に従いオブジェクトの動きにあわせて自動的に中間の基準フレームを設定する．

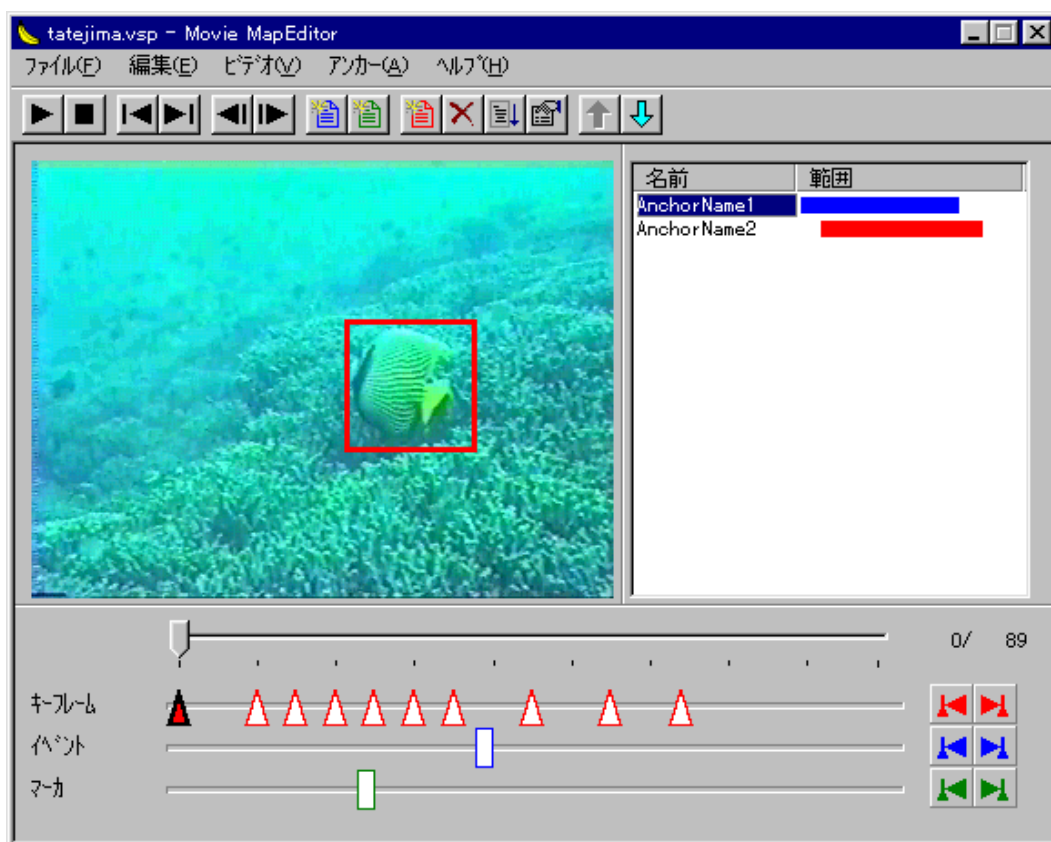


図 6.3: Movie MapEditor の画面イメージ

3. 追跡誤りの修正:

コンテンツ制作者は自動アンカー設定処理の結果を画面上で確認し、必要に応じてアンカーの修正処理を行う。修正処理は任意の位置に中間の基準フレームを挿入するか、既に挿入されている基準フレームを削除またはアンカーの位置、大きさを修正することにより行う。

Movie MapEditor における動画像解析を用いた自動アンカー設定処理の流れについて説明する。

1. 順方向追跡:

コンテンツ制作者が設定した開始フレームのアンカーを最初のテンプレートとして数フレーム間隔でパターンマッチングを繰り返して終了フレームまでオブジェクトの追跡を行う。

2. 追跡結果判定:

追跡の結果得られた終了フレームのアンカー位置と、コンテンツ制作者が指定した終了フレームのアンカー位置が一致していれば 4. に進む。一致していない場合は 3. に進む。

3. 逆方向追跡:

コンテンツ制作者が設定した終了フレームのアンカーを最初のテンプレートとして 1. と逆方向の追跡を行う。追跡中に 1. で得られた軌跡に到達した時点、または追跡処理が設定区間全体の 50 % に到達した時点で追跡処理を打ちきる。

4. 中間点削除:

1. から 3. の処理で得られた軌跡をなるべく少ない数の中間点で近似するように、冗長な中間点の削除処理を行う。

また、自動アンカー設定方式の特徴としては次の 2 点があげられる。

1. 双方向の追跡結果の併用:

コンテンツ制作者のアンカーの存在区間の指定と開始フレーム、終了フレームの設定を同時に行うことにより、順方向、逆方向の追跡が可能となる。パターンマッチングを用いた追跡の場合、途中で一度追跡を誤るとその後の追跡は無駄になるという問題があるが、本方式では双方向の追跡結果を併用することで、追跡の精度が向上する。

2. 誤り修正を容易にするための中間点削除:

追跡の結果得られた軌跡に基づき中間点を設定すると、数フレーム毎に中間点が挿入されることになる。このような場合、対象のオブジェクトが直線的な動きをしている場合にはデータが冗長になってしまい、誤りを修正する際に多くの作業量を必要とする。そこで、本方式では、数フレーム毎に挿入された中間点のうち、冗長な中間点を自動削除する処理を行うことにより、追跡誤り修正の手順を大幅に削減することができる。

6.3.5 インタラクティブ動画オブジェクトの再生

6.3.4 で構築した Movie MapEditor で編集したインタラクティブ動画オブジェクトの再生について述べる。

VisualSHOCK MOVIE では、6.3.1 で設定した設計指針に従い、次の 2 種類の再生方式を提供する。

1. VisualSHOCK MOVIE Player を Web ブラウザのコンポーネントとして提供し、HTML コンテンツでの一部として編集したインタラクティブ動画オブジェクトを再生する。

2. 編集したインタラクティブ動画オブジェクトを 2.3.3 の手順に基づいてイベント、イメージマップの機能を持つ RealMedia 形式に変換し、RealPlayer で再生する。

表 6.3 に、VisualSHOCK MOVIE におけるインタラクティブ動画オブジェクトの再生方式について示す。

表 6.3: インタラクティブ動画オブジェクトの再生方式

	Internet Explorer		Netscape Communicator	
	AVI, Quick-Time Movie, MPEG-1	RealMedia	AVI, Quick-Time Movie, MPEG-1	RealMedia
VisualSHOCK MOVIE Player ActiveX コントロール		×	×	×
VisualSHOCK MOVIE Player プラグイン	×	×		×
RealPlayer	×		×	

VisualSHOCK MOVIE Player

VisualSHOCK MOVIE Player は、WWW 上で動作するコンポーネントとして動作し、HTML で記述されたマルチメディアコンテンツの一部としてインタラクティブ動画オブジェクトを Web ブラウザ上で再生する機能を持つ。

動画データの形式は、AVI、QuickTime Movie、MPEG-1 に対応し、表 6.3 に示した通り、ActiveX コントロール版と NetScape プラグイン版があり、使用するブラウザの種類にあわせてそれぞれ Internet Explorer、Netscape Communicator 上で動作する。

図 6.4 に VisualSHOCK MOVIE Player の利用イメージについて示す。

図 6.4 を見ても分かるように、VisualSHOCK MOVIE Player では、動画データに登場するオブジェクトを含む領域に対して設定されたアンカーを映像データと重ねて表示する。コンテンツ利用者が、アンカーをクリックすると、関連情報を検索して表示する。

VisualSHOCK MOVIE Player におけるアンカーの表示機能は、次の特徴を持つ。

1. Player 側でのアンカー表示/非表示の選択機能
2. アンカー領域内にカーソルが入ったときのカーソル形状の自動変更機能

図 6.5、6.6、6.7 に、VisualSHOCK MOVIE ActiveX コントロールの持つプロパティ、メソッド、イベントについて定義する。

例えば、図 6.2 で編集したインタラクティブ動画オブジェクトを再生する HTML ファイルにおいて、AncID1 のアンカーをクリックした時のアクションは、VBScript に対するイベントとして図 6.8 の通り記述することができる。

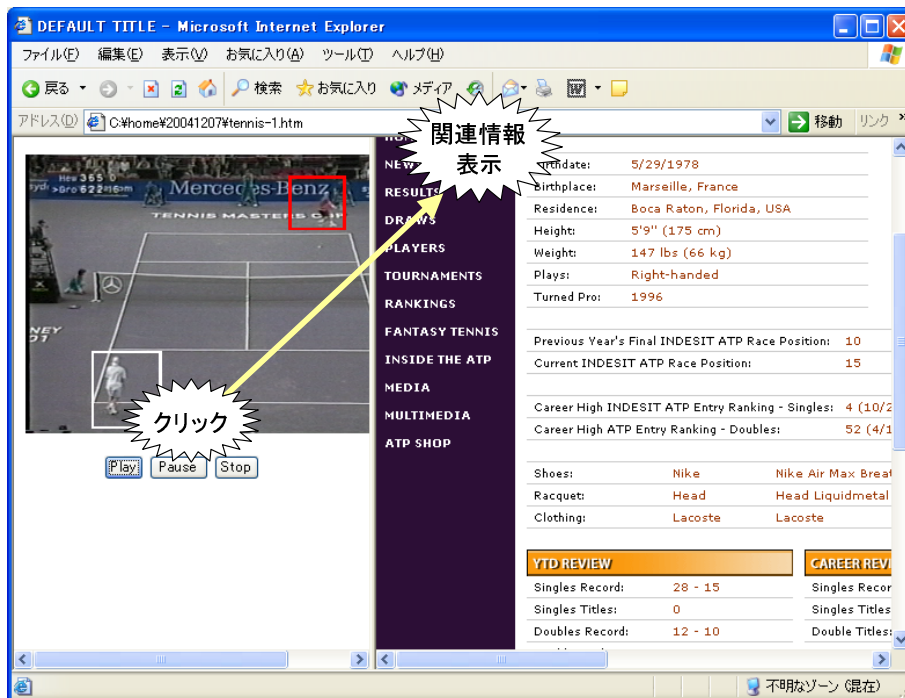


図 6.4: VisualSHOCK MOVIE Player の利用イメージ

RealPlayer

6.3.4 で構築した Movie MapEditor で編集したインタラクティブ動画オブジェクトをストリーム形式の映像データに変換することにより，LAN 環境やインターネット環境などのネットワーク経由で制作した動画ハイパーメディアコンテンツを利用することができる。

VisualSHOCK MOVIE では，編集したインタラクティブ動画オブジェクトを RealMedia 形式に変換する機能を持つ。具体的には，映像データを RealMedia に変換する際に，あわせてインタラクティブ動画オブジェクトのアンカー，イベントを 2.3.3 の手順に基づいて RealMedia のイメージマップ，イベントに多重化する。

変換した RealMedia を RealPlayer で再生する際に，多重化されたイメージマップ，イベントに対処して処理を実行することができる。図 6.9 に VisualSHOCK MOVIE における RealPlayer の利用イメージについて示す。

図 6.9 を見ても分かる通り，VisualSHOCK MOVIE では，編集したインタラクティブ動画オブジェクトを RealMedia に変換する機能を持ち，変換後の RealMedia を RealPlayer で再生する際に，動画データに登場するオブジェクトを含む領域に対して設定されたアンカーを映像データと重ねて表示する。コンテンツ利用者が，アンカーをクリックすると，関連情報を検索して表示する。

RealPlayer におけるアンカーの表示機能は，次の特徴を持つ。

1. Player 側ではアンカーは常に表示
2. アンカー領域内にカーソルが入ったときのカーソル形状の自動変更機能

図 6.9 では RealPlayer 単独のアプリケーションとして使用した例について示しているが，Visual-

プロパティ名	説明
<i>Long ImageSourceHeight</i>	動画ファイルのオリジナルの高さを表示する
<i>Long ImageSourceWidth</i>	動画ファイルのオリジナル幅を表示する
<i>Long Duration</i>	動画ファイルの長さをフレーム数で表示する
<i>Long FrameRate</i>	動画ファイルのフレームレートを表示する
<i>BSTR AnchorID</i>	動画に設定されているアンカーIDのリストを表示する
<i>BSTR AnchorName</i>	動画に設定されているアンカー名のリストを表示する
<i>Long AnchorNum</i>	動画に設定されているアンカー総数を表示する
<i>BSTR EventName</i>	動画に設定されているイベント名のリストを表示する
<i>Long EventNum</i>	動画に設定されているイベント総数を表示する
<i>BSTR EventPosition</i>	動画に設定されているイベントのフレーム番号を表示する
<i>BSTR MarkerName</i>	動画に設定されているマーカ名のリストを表示する
<i>Long MarkerNum</i>	動画に設定されているマーカ総数を表示する
<i>BSTR MarkerPosition</i>	マーカのフレーム番号を表示する
<i>Long ActivateMethod</i>	アンカーをアクティブにするイベントの種類を指定する
<i>Boolean AutoPlay</i>	動画が読み込まれた後、自動的に再生を開始するかどうか指定する
<i>Boolean AutoRepeat</i>	動画の再生を自動的に繰り返すかどうか指定する
<i>Boolean AutoRewind</i>	動画の再生が完了後、自動的に巻き戻すかどうか指定する
<i>Boolean ChangeCursor</i>	カーソルがアンカーの上に来た時変化させるかどうか指定する
<i>Long CurrentPosition</i>	動画の現在のフレームを指定する
<i>BSTR CursorFile</i>	カーソルファイルの名前を指定する
<i>Boolean EnableContextMenu</i>	コンテキストメニューを表示するかどうかを指定する
<i>Boolean FullScreenMode</i>	動画の再生範囲をフルスクリーンに拡大するかどうかを指定する
<i>BSTR MovieFile</i>	VSPファイルのパスを指定する
<i>Long PlaybackRate</i>	動画の再生速度を指定する
<i>Boolean ShowAnchor</i>	アンカーを表示するかどうかを指定する
<i>Long ShowAnchorSetting</i>	Visual Effect設定を有効にする
<i>Long Volume</i>	音声ボリュームを指定する

図 6.5: VisualSHOCK MOVIE Player ActiveX コントロール (a) プロパティ一覧

メソッド名	説明
<i>Void Play(FrameNo)</i>	指定したFrameNoから動画を再生する
<i>Void Pause()</i>	現在位置で停止する
<i>Void Stop()</i>	再生を停止して、AutoRewindの設定を元に位置をリセットする
<i>Boolean EnableAnchor(AnchorID)</i>	指定したAnchorIDから、開かれている動画ファイルの位置を捜し出して選択する
<i>Boolean EnableMarker(MarkerName)</i>	指定したMarkerNameにジャンプする
<i>BSTR GetAnchorAction(AnchorID)</i>	指定したAnchorIDに設定されているアクションを返す
<i>BSTR GetAnchorIDList(FrameNumber)</i>	指定したFrameNumberに設定されているアンカーのアンカーIDのリストを返す
<i>BSTR GetAnchorNameList(FrameNumber)</i>	指定したFrameNumberに設定されているアンカーのアンカー名のリストを返す
<i>BSTR GetAnchorProps(AnchorID, FrameNumber)</i>	指定したFrameNumber内にあるAnchorIDのプロパティを返す
<i>BSTR GetEventAction(EventName)</i>	指定したEventNameに設定されているアクションを返す
<i>Void AboutBox()</i>	“About”ダイアログボックスを表示する

図 6.6: VisualSHOCK MOVIE Player ActiveX コントロール (b) メソッド一覧

イベント名	説明
<i>Void OnActivateAnchor(AnchorID)</i>	<i>AnchorID</i> がアクティブになったときに発生する
<i>Void OnAnchorAppear(AnchorID)</i>	動画の再生が <i>AnchorID</i> の開始フレームに到達したときに発生する
<i>Void OnAnchorDisappear(AnchorID)</i>	動画の再生が <i>AnchorID</i> のエンディングフレームに到達したときに発生する
<i>Void OnCursorIn(AnchorID)</i>	マウスカーソルが <i>AnchorID</i> の境界線内に入ったときに発生する
<i>Void OnCursorOut(AnchorID)</i>	マウスカーソルが <i>AnchorID</i> の境界線外に出たときに発生する
<i>Void OnError(ErrorNum)</i>	<i>ErrorNum</i> のエラーを発生したときに発生する
<i>Void OnMarker(MarkerName)</i>	<i>MarkerName</i> が設定されているフレームに動画の再生が到達したときに発生する
<i>Void OnVSEvent(EventName)</i>	<i>EventName</i> が設定されているフレームに動画の再生が到達したときに発生する
<i>Void MouseDown(short Button, short Shift,OLE_XPOS_PIXELS x, OLE_YPOS_PIXELS y)</i>	マウスイベントの <i>MouseDown</i> が発生したときに発生する
<i>Void MouseMove(short Button, short Shift,OLE_XPOS_PIXELS x, OLE_YPOS_PIXELS y)</i>	マウスイベントの <i>MouseMove</i> が発生したときに発生する
<i>Void MouseUp(short Button, short Shift,OLE_XPOS_PIXELS x, OLE_YPOS_PIXELS y)</i>	マウスイベントの <i>MouseUp</i> が発生したときに発生する
<i>Void OnMovieFinish()</i>	動画が再生を終了したときに発生する
<i>Void OnMovieStart()</i>	動画が再生を開始したときに発生する
<i>Void OnMovieStop()</i>	動画が停止、あるいは一時停止したときに発生する
<i>Void OnPositionChange(oldPosition, NewPosition)</i>	動画の再生位置が変わったときに発生する
<i>Void OnStateChange(oldState, NewState)</i>	動画再生の状態が変化したときに発生する
<i>Void OnTimer()</i>	タイマーイベントで定期的に発生する

図 6.7: VisualSHOCK MOVIE Player ActiveX コントロール (c) イベント一覧

```
<SCRIPT LANGUAGE="VBScript">
<!--
Sub VSMovie21_OnActivateAnchor(AncID)
  Select Case AncID
    Case "AncID1"
      parent.AnchorInfo.location.href="http://www.melco.co.jp/"
    Case "AncID2"

    Case Else

  End Select
End Sub
-->
</SCRIPT>
<!--VBScript ends here -->
```

図 6.8: アンカークリック時のアクション設定

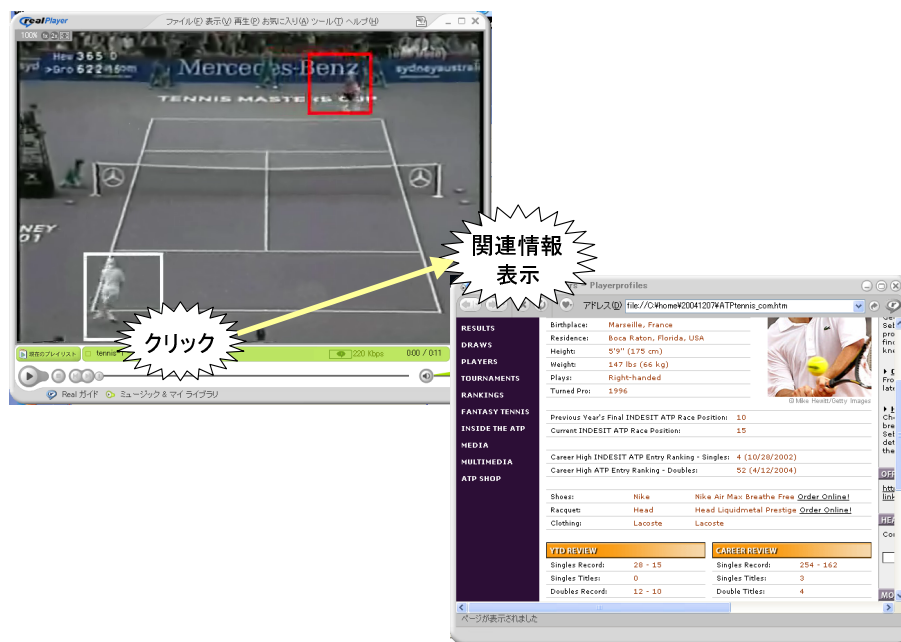


図 6.9: VisualSHOCK MOVIE における RealPlayer の利用イメージ

SHOCK MOVIE Player 同様 Web ブラウザのコンポーネントとして使用することも可能であり，その場合は，Web ブラウザの一部に映像データを表示する．

6.4 初等教育への応用及び評価実験

VisualSHOCK MOVIE は初等教育への応用を目的とした動画ハイパーメディアシステムとしてシステム要件の抽出を行い，その要件に基づいて設計，構築を行った．構築したシステムの評価を行うにあたり，前述した設計指針に対する妥当性とシステムの有効性の検証を目的として，実際に小学生むけの初等教育への応用を対象としてコンテンツ利用及びコンテンツ制作に関する評価実験を実施した．

コンテンツ利用に関する評価実験は，学習コンテンツ，動画ウォークスルーコンテンツ，の2種類の評価用コンテンツを準備し，実際に小学生に利用してもらい，利用後にアンケート調査を行う形で実施した [129, 130] ．

図 6.10 にコンテンツ利用に関する評価実験のシステム概要について示す．

評価用コンテンツは，動画データはネットワーク経由での利用を前提として RealMedia 形式に変換し，RealPlayer を Web ブラウザ上に埋め込む形で制作した．図 6.10 において，Web サーバ，RealServer を教育研究所に設置し，CATV ネットワークを介して，高知市内の小中学校のパソコン教室に設置した PC から利用する形で評価実験を実施した．

一方，コンテンツ制作に関する評価実験は，計算機に不慣れな教師を想定したコンテンツ制作者に対して，Movie MapEditor を使用して実際にコンテンツを制作してもらい，利用後にアンケート調査を行う形で実施した．

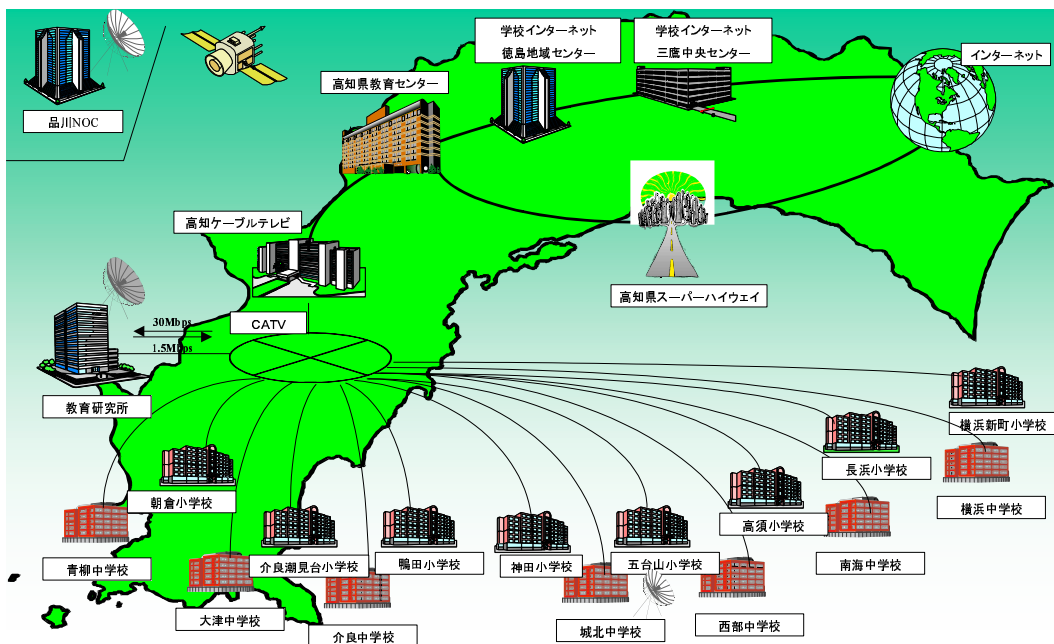


図 6.10: コンテンツ利用に関する評価実験のシステム概要 [130]

6.4.1 コンテンツ利用に関する評価実験

学習コンテンツに関する評価実験

高知市の小学生を対象に高知の伝統工芸を紹介する社会科学学習用のコンテンツを VisualSHOCK MOVIE(図 6.11(a) 参照) と、従来技術により動画の下部にリストを表示しリスト中の文字列をクリックして関連情報を表示する方法(図 6.11(b) 参照) により制作した。

図 6.11 において、(a) では動画上に登場するオブジェクトに対して設定されたアンカーをクリックすることによって、(b) では動画の下のフレームに表示されたリスト中の文字列をクリックすることにより、関連情報を表示することができる。

制作した評価用コンテンツを実際に小学生に利用してもらい、利用後にアンケート調査を実施した [129,130]。

評価実験は次の 2 方式に分けて実施した。

1. 実験 A: 高知市立鴨田小学校の生徒 24 名に対して評価実験を実施した。生徒を 13 名と 11 名の 2 つのグループに分け、前者は VisualSHOCK MOVIE を使用した (a) を、後者は従来技術を使用した (b) を利用してもらい、並行する形で評価実験を実施した。
2. 実験 B: 高知市立高須小学校の生徒 15 名に対して評価実験を実施した。一人の生徒が VisualSHOCK MOVIE を使用した (a) と従来技術を使用した (b) を同時に利用してもらい、両者を比較する形で評価実験を実施した。

図 6.12 に実験 A の結果を図 6.13 に実験 B の結果を示す。

図 6.12(a)、図 6.13(a) より、VisualSHOCK MOVIE を使った場合は、従来技術と比較して、動画内の関連情報を持つ物体の判別や、関連情報表示の操作手順が簡単であることが分かった。逆に図

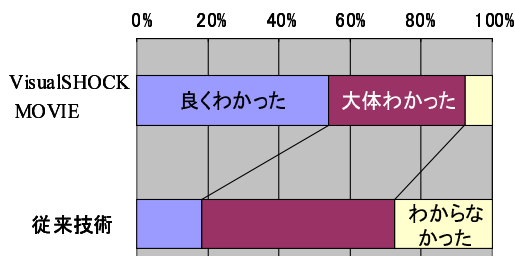


(a) VisualSHOCK MOVIEを適用したコンテンツ

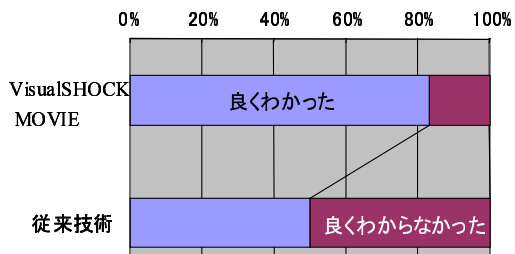


(b) 従来技術を適用したコンテンツ

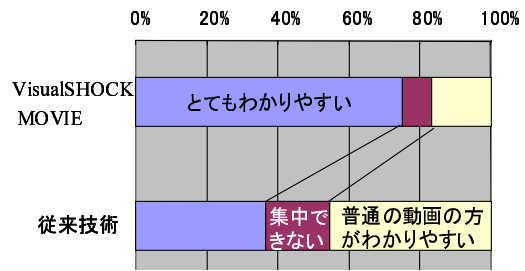
図 6.11: 学習コンテンツの画面イメージ



(a) 動画内のどこに詳しい説明がついているかわかりましたか？



(b) 動画の内容は理解できましたか？



(c) 内容のわかりやすさについて 普通の動画(テレビ)と比較して下さい

図 6.12: 実験 A の結果

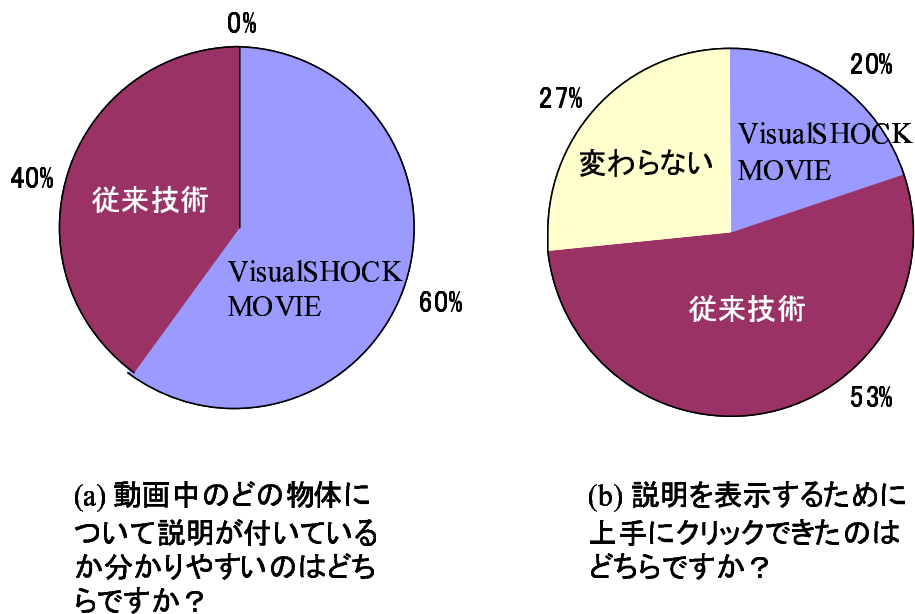


図 6.13: 実験 B の結果

6.13(b) より，アンカー領域が動画内で動いていることにより，従来技術と比較して，物体のクリックが難しいことが分かった．

教育効果の点から考察すると，本評価実験の範囲においては，内容の理解度は VisualSHOCK MOVIE で制作したコンテンツが従来のコンテンツと比較して高いという結果を得た (図 6.12(b)(c)，図 6.13(a) 参照) ．

動画ウォークスルーコンテンツに関する評価実験

高知市の小学生を対象に高知の牧野植物園の園内を動画ウォークスルーするコンテンツを VisualSHOCK MOVIE(図 6.14(a) 参照) と，従来技術により動画の下部にリストを表示しリスト中の文字列をクリックして動画データの適切な位置にジャンプさせる方法 (図 6.14(b) 参照) により制作した．

図 6.14 において，(a) では動画上に登場するドアなどに対して設定されたアンカーをクリックすることによって，(b) では動画の下のフレームに表示されたリスト中の文字列をクリックすることによって，動画データの適切な位置にジャンプさせることができ，植物園内を撮影した動画データを見ることができる．

制作したコンテンツを実際に小学生に利用してもらい，利用後にアンケート調査を実施した [129,130] ．評価実験は次の 2 方式に分けて実施した ．

1. 実験 C: 高知市立鴨田小学校の生徒 31 名に対して評価実験を実施した．生徒を 15 名と 16 名の 2 つのグループに分け，前者は VisualSHOCK MOVIE を使用した (a) を，後者は従来技術を使用した (b) を利用してもらい，並行する形で評価実験を実施した ．
2. 実験 D: 高知市立高須小学校の生徒 22 名に対して評価実験を実施した．一人の生徒が VisualSHOCK MOVIE を使用した (a) と従来技術を使用した (b) を同時に利用してもらい，両者を



(a) VisualSHOCK MOVIEを適用したコンテンツ

(b) 従来技術を適用したコンテンツ

図 6.14: 動画ウォークスルーコンテンツの画面イメージ

比較する形で評価実験を実施した。

図 6.15 に実験 C の結果を図 6.16 に実験 D の結果を示す。

図 6.15(a), 図 6.16(a) より, VisualSHOCK MOVIE を使った場合は, 従来技術と比較して, 植物園の仮想ウォークスルー効果が高いことが分かった。逆に図 6.15(b), 図 6.16(b) より, 植物園のどこを歩いているか見失ってしまう可能性が高いことが分かった。

教育効果の点から考察すると, 本評価実験の範囲においては, 仮想ウォークスルー効果による臨場感が, VisualSHOCK MOVIE で制作したコンテンツが従来のコンテンツと比較して高いという結果を得た (図 6.15(a)(c), 図 6.16(a) 参照)。

6.4.2 コンテンツ制作に関する評価実験

計算機に不慣れな教師を想定したコンテンツ制作者 4 名に対して, Movie MapEditor を使用して実際にコンテンツを制作してもらい, 表 6.4 に示した評価項目について, 1 から 5 までの 0.5 単位で 10 段階評価のアンケート調査を行った。

表 6.4 の結果を見ても分かる通り, Movie MapEditor におけるアンカー自動設定機能及びアンカー修正機能については高いユーザ評価が得られた。また, ユーザからの修正点として, 次のコメントが得られた。

1. イベントとマーカの使い方の違いが分かり難い
2. アンカーの形状で円形と矩形以外を使いたい
3. アンカーに対してもう少し細かい視覚効果を設定したい

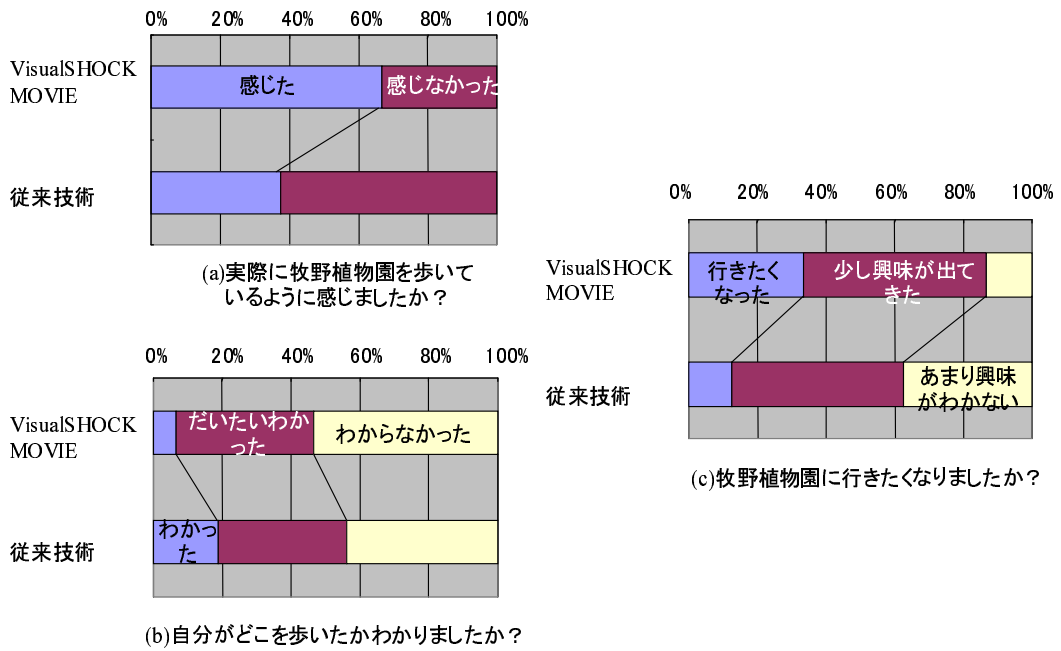


図 6.15: 実験 C の結果

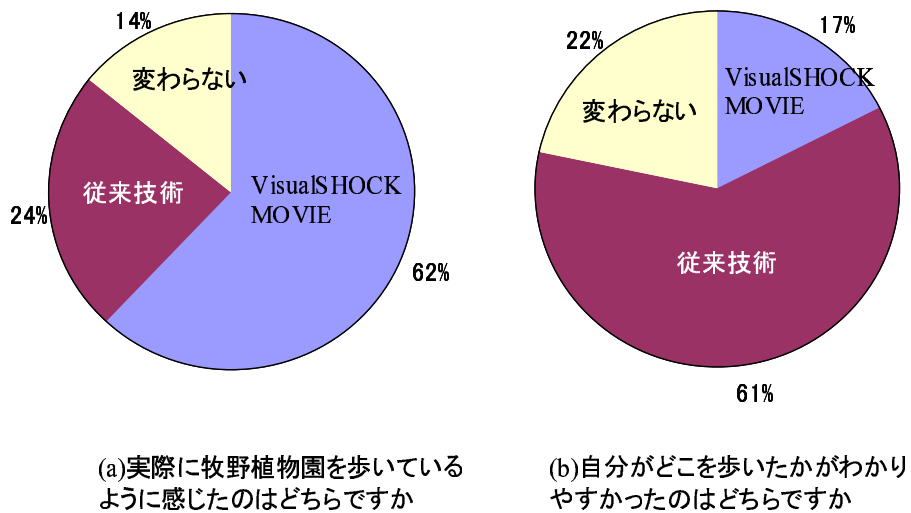


図 6.16: 実験 D の結果

表 6.4: コンテンツ制作実験の結果

評価項目	ユーザ A	ユーザ B	ユーザ C	ユーザ D	平均
アンカー自動設定機能	5	4.5	5	5	4.875
マニュアルによるアンカー修正機能	5	5	5	5	5
アンカーのアクション設定機能	4	4	4	4	4
アンカーの視覚効果設定機能	4	3.5	4	4	3.875
イベント編集機能	4	4	4	4	4
マーカ編集機能	3	3.5	4	4	3.625
RealMedia 変換機能	4	3.5	4	4	3.825
HTML 出力機能	4	4	4	4	4
全体的な操作性	4	4	5	5	4.5
ソフトウェアの応答速度	4	4	4	4	4
総合評価	4.5	4	4	4.5	4.25

6.5 おわりに

第6章では、マルチメディア教育での利用を目的とした動画ハイパーメディアシステム VisualSHOCK MOVIE の設計、構築について述べ、初等教育への応用を想定した評価実験について述べ、有効性について検証を行った。

6.2 で設定したシステム要件に基づき、構築した VisualSHOCK MOVIE の評価を行う。

1. インタラクティブ動画による教育効果向上:
6.4 で実施した評価実験の結果を考察したところ、本評価実験の範囲においては、VisualSHOCK MOVIE を用いて制作した学習コンテンツや動画ウォークスルーコンテンツは、従来技術で制作したコンテンツと比較して高い教育効果を確認することができた。
2. 編集作業の簡易化:
動画ハイパーメディアシステムにおけるコンテンツ制作のうち最も手間のかかるアンカー設定作業の簡易化を目的としてインタラクティブ動画オブジェクトの編集機能を持つ Movie MapEditor の設計、構築を行った。また、6.4 で実施した評価実験の結果を考察したところ、動画像解析による自動追跡と手動による修正を組み合わせることにより、従来方式の場合、手間のかかるアンカー設定作業の効率化を実現することができた。
3. Web ブラウザ上での利用:
通常の HTML コンテンツとして Web ブラウザ上でインタラクティブ動画オブジェクトを再生することを目的として、VisualSHOCK MOVIE Player の設計、構築を行った。使用する Web ブラウザの種類にあわせて Player を準備することにより、通常使用している Web ブラウザのマルチメディアコンテンツの動画再生機能を強化する形で利用できるようになった。
4. ストリーミング配信への対応:
制作したコンテンツを CD-ROM や DVD-ROM 上利用する場合は、AVI、QuickTime Movie、MPEG-1 形式の動画データをそのまま使用し、LAN 環境やインターネット環境などのネットワーク環境で利用する場合は、編集結果を再編集せずにストリーム形式の映像データに変換

することが可能であり，利用する環境にあわせて最初からコンテンツを作り直す必要が無くなった．

評価を行った結果，設定したシステム要件については対応することができた．

第 6 章にて構築したシステムでは，ストリーミング配信への対応方式として，編集したインタラクティブ映像オブジェクトを RealMedia 形式に変換することにより対応したが，この方式では次の課題が残っている．

1. RealMedia への変換時に，映像データ上にアンカーを描画した形でエンコードするため，RealPlayer で再生を行う際にアンカーの表示を消すことができない．
2. RealMedia のイメージマップは静止画のイメージマップを拡張した仕様となっているため，設定対象のオブジェクトの動きが激しい場合は，その動きに追従することができない．

今後，第 5 章で確立した拡張イベントモデルに基づく映像と付加情報の同期配信方式を適用することにより，上記課題への対処が可能である．

第7章 結論

本研究では、動画ハイパーメディアシステムの構築方式とその応用について研究を行った。

本研究では、動画ハイパーメディアシステムの構成要素と技術課題について整理し、本研究の目的について次の通り整理した。

1. 動画データの効率的な管理方式の確立
2. アンカーの効率的な設定方式の確立
3. 映像と付加情報の同期配信方式の確立
4. 動画ハイパーメディアシステムの初等教育への応用

以下、それぞれの目的に対する研究成果について述べる。

第3章にて、動画ハイパーメディアシステムのバックエンド技術において、動画データの効率的な管理方式の確立を目的として、ビデオオブジェクトモデルの提案を行った。さらに動画応用システムにおける効率的な動画データ管理を目的としてビデオオブジェクト管理システムを設計、構築し、評価システムとして動画ハイパーメディアシステムを構築した。構築したビデオオブジェクト管理システムを動画ハイパーメディアシステムの構築に適用した結果、次のシステム要件について、有効性を確認することができた。

1. 動画素材の任意の部分をビデオクリップとして定義可能:
動画素材の任意の区間を切り出して primitive オブジェクトとして新たに定義できる機構を提供した。primitive オブジェクトは、動画素材のポインタのみを持っている仮想的なオブジェクトで動画素材自体は編集の対象では無いので、直接動画素材を編集する方式と比較して、編集作業の効率化が可能となる。
2. ビデオクリップ同士を組み合わせることが可能:
複数のビデオオブジェクトの Video Structure 部分を組み合わせる新しいビデオオブジェクトを作成する *compose* メソッドを提供し、既存のオブジェクトを組み合わせることにより編集作業の効率化が可能となる。
3. ビデオクリップの内容によるナビゲーション検索を提供:
ビデオクリップの代表フレームの識別子を返す *get_rep_frame* メソッドを用いて、ビデオクリップの一覧表示を行う際にあわせて代表フレームを表示することにより、ビデオオブジェクトの検索効率の向上を図る。
4. すべての動画応用システムから同じインタフェースを介して利用可能:
オブジェクト指向アプローチに基づいてビデオオブジェクトの設計を行うことにより、すべての動画応用システムから同一のインタフェースを介して動画データにアクセスする機構を実現できた。
5. ビデオクリップに表示属性を独自のビューとして定義可能:
ビデオオブジェクトモデルの設計を行う際に、アノテーションとして表示属性と検索属性を分離して設定できる機構を提供した。

第4章にて、動画ハイパーメディアシステムのオーサリング技術において、アンカーの効率的な設定方式の確立を目的として、自動追跡と手動修正を組み合わせたアンカー設定方式の提案を行った。提案した方式は、動画画像解析によってアンカーの対象のオブジェクトの動きを自動的に追跡するとともに、追跡結果を必要に応じて手作業で修正を行う方式であり、その方式に基づいてアンカー設定ソフトウェアを構築した。構築したソフトウェアを実際にスポーツ映像を対象として評価実験を行った結果、本方式の特徴である次の項目について有効性を確認することができた。

1. 双方向追跡の適用による追跡効率の向上
2. 中間点削除処理による手作業でのアンカー修正作業の効率化

第5章にて、動画ハイパーメディアの配信技術において、映像と付加情報の同期配信方式の確立を目的として、拡張イベントモデルを提案を行った。提案したモデルでは、映像中に対応する付加情報との時間的関係を定義した同期情報を周期的にイベントとして多重化し、その同期情報に基づいて、映像と付加情報の同期配信を実現する方式であり、次の特徴を持つ。

1. 蓄積映像、ライブ映像などの映像の形式やダウンロード型配信、オンデマンド型配信、放送型配信などの配信方式に依存しない同期方式である
2. 開始点同期、実時間同期などの付加情報の使用目的にあわせた同期精度が設定可能である

提案した拡張イベントモデルを Windows Media 及び MPEG-2 TS を対象プラットフォームとしてエンコード/デコードライブラリとして実装し、そのうち Windows Media 版ライブラリを使用して評価システムを構築し、その実時間同期性能について評価実験を行った結果、提案方式の有効性について検証できた。

第6章にて、動画ハイパーメディアシステムの応用を想定して、動画ハイパーメディアシステムの初等教育への応用を目的として、初等教育への応用を目的とした動画ハイパーメディアシステムの設計、構築を行った。構築した動画ハイパーメディアシステムを初等教育への応用を想定したコンテンツ利用及びコンテンツ制作に関する評価実験を行った結果、本方式のシステム要件として設定した次の項目について有効性を確認することができた。

1. インタラクティブ動画による教育効果向上
2. 編集作業の簡易化
3. Web ブラウザ上での利用
4. ストリーミング配信への対応

表7.1に本研究における開発ソフトウェアの対応動画形式に関してまとめる。

本研究では、動画ハイパーメディアシステムの構築方式とその応用に関して、その構成要素であるバックエンド技術、オーサリング技術、配信技術、応用技術において技術課題を抽出し、その技術課題に対応する形で目標を設定し、研究を実施した。第3章から第6章にかけて得られた研究成果についてまとめたところ、設定した目標を達成することが確認でき、その有効性について確認することができた。

今後は、本研究の研究成果により確立した技術を組み合わせることにより、動画ハイパーメディアシステムの実用化にむけた研究を行っていく所存である。

表 7.1: 本研究における開発ソフトウェアの対応動画形式

構成要素	開発ソフトウェア	入力	出力
応用	Movie MapEditor	AVI QuickTime Movie MPEG-1	RealMedia
	VisualSHOCK MOVIE Player (ActiveX, プラグイン)	AVI QuickTime Movie MPEG-1	–
配信技術	オーサリングソフトウェア エンコードライブラリ	AVI, Live Video MPEG-2 TS	Windows Media MPEG-2 TS
	再生ソフトウェア デコードライブラリ	Windows Media MPEG-2 TS	–
オーサリング 技術	アンカー設定ソフトウェア	AVI QuickTime Movie	–
バックエンド 技術	Video Object Editor Video Object Searcher	AVI QuickTime Movie	–
	Video Object Player 動画ハイパーメディアソフトウ ェア	MPEG-1	

謝辞

本論文をまとめるにあたり，社会人博士課程の2年間にわたりご指導・ご鞭撻を賜りました岡田謙一教授（慶應義塾大学 理工学研究科 開放環境科学専攻）に深く感謝の意を表したいと思います．岡田謙一教授には，筆者が学部4年の学生として当時の北川研究室に配属されてから修士2年で松下研究室を修了するまでの3年間と，就職後に社会人博士課程に入学してからの合計5年間，大変お世話になりました．また，情報処理学会 放送コンピューティング研究グループの運営委員や DICOMO(マルチメディア，分散，協調とモバイル) シンポジウムの実行委員などの学会活動でも一緒させて頂いており，感謝しております．

重野寛助教授（慶應義塾大学 理工学研究科 開放環境科学専攻）には，修士2年での松下研究室での1年間と，社会人博士課程での岡田研究室での2年間の合計3年間，お世話になりました．

その他に，開放環境科学専攻の先生方には，暖かくご指導頂きました．特に，小澤慎治教授（慶應義塾大学 理工学研究科 開放環境科学専攻），山本喜一助教授（慶應義塾大学 理工学研究科 開放環境科学専攻）には，博士論文審査において貴重なご意見やご指導を頂きました．

本論文は，筆者の三菱電機株式会社 情報技術総合研究所における研究開発成果を中心に博士論文としてまとめ直したものです．

筆者の所属する三菱電機株式会社 情報技術総合研究所 マルチメディアネットワーク技術部の上司である，厚井裕司教授（岩手大学 工学部 情報システム工学科），菅隆志氏，黒田健児氏（三菱電機株式会社 本社 開発業務部長），岡進氏，泉丙完氏（三菱電機株式会社 三田製作所），中島宏一氏には，社会人博士課程への入学を勧めて頂き，また，社会人博士課程在学中も多大なるご支援を頂きましたことに感謝いたします．

また，筆者の大学時代の指導教授であり，社会人になってからも学会などでお世話になっています松下温教授（東京工科大学 コンピュータサイエンス学部長），及び筆者が三菱電機株式会社に入社以来お世話になっています水野忠則教授（静岡大学 情報学部 情報科学科），坂下善彦教授（湘南工科大学 工学部 情報工学科），小泉寿男教授（東京電機大学 理工学部 情報システム工学科），宮地泰造教授（東海大学 電子情報学部 情報メディア学科）に感謝いたします．

脇本浩司氏は第3章，第4章，第6章の共同研究者であり，本研究に関して全般的に議論，ご指導頂きました．神田準史郎氏は第3章，第4章の共同研究者であり，特に，動画像処理に関連する研究に関してご協力頂きました．第3章の研究の実施に関しては，吉良賢治氏（三菱電機株式会社 インフォメーションシステム事業推進本部），永沼和智氏（三菱電機インフォメーションシステムズ株式会社），高梨郁子女氏にご協力頂きました．第5章の研究の実施に関しては，福田雅裕氏（三菱電機株式会社 知的財産センター），山田淳氏（三菱電機株式会社 社会インフラ事業本部 社会eソリューション事業所），松本佳宏氏（ケイ・オプティコム株式会社）にご協力頂きました．横里純一氏には，第6章の評価実験に関してご協力頂きました．また，本評価実験は，高知市立高須小学校，高知市立鴨田小学校の方々にご協力頂きました．

上記以外にも，本研究を進めるにあたり，筆者の所属した情報電子研究所，情報システム研究所，情報技術総合研究所，官公 CCV 事業推進センター，情報システム製作所の上司，同僚の方々に感謝

したいと思います。

慶應義塾大学 理工学研究科 開放環境科学専攻 岡田・重野研究室の学生の皆様には、研究に関する議論以外にも、公私ともにお世話になり大変感謝しています。特に、社会人博士課程で密に情報交換させて頂いた古市昌一氏(三菱電機株式会社 鎌倉製作所)、稲村浩氏(株式会社エヌ・ティ・ティ・ドコモ)、野村恭彦氏(富士ゼロックス株式会社)に感謝します。

最後に、筆者が生まれて以来、常に愛情深く見守って頂いた父、母、祖母と、仕事と学業の両立に全面的に協力頂いた妻の敦子と、社会人博士課程の在学中に誕生した長男の直宏に感謝の意を表したいと思います。

なお、本研究の一部は、通信・放送機構「目的適応型マルチキャスト通信ネットワークアーキテクチャ研究開発」及び、通信・放送機構「通信・放送融合における高品質コンテンツおよび付加情報の配信の実用化に資する技術開発」の一環として実施したものです。

参考文献

- [1] 総務省 (編): 平成 16 年度版 情報通信白書,
<http://www.johotsusintokei.soumu.go.jp/whitepaper/ja/h16/index.html>
- [2] Jacob Nielsen: Hypertext & Hypermedia, ACADEMIC PRESS (April 1990)
- [3] 原良憲: ハイパーメディア入門, bit, Volume 22, Number 8, pp.35-42 (August 1990)
- [4] 田中克己: ハイパーメディアとは何か, Computer Today, Volume 7, Number 4, pp.4-12 (July 1990)
- [5] 金子朝男: ハイパーメディアの研究動向, 情報処理学会誌, Volume 34, Number 1, pp.60-71 (January 1993)
- [6] Jakob Nielsen(著), 篠原 稔和 (訳), 三好 かおる (訳): マルチメディア & ハイパーテキスト原論
インターネット理解のための基礎理論情報デザインシリーズ, 東京電機大学出版局 (September 2002)
- [7] W3C: W3C HTML Home Page,
<http://www.w3c.org/MarkUp/>
- [8] S. W. Smoliar and HongJiang Zhang: Content based video indexing and retrieval, IEEE
Multimedia, Volume 1, Number 2, pp.62-72 (Summer 1994)
- [9] 郵政省 電気通信局 高度通信網振興課: 学校における複合アクセス網活用型インターネットに
関する研究開発について,
<http://www2.nict.go.jp/ns/s810/gakko/school1.html>
- [10] 文部科学省: 学校における情報教育の実態等に関する調査結果,
http://www.mext.go.jp/b_menu/houdou/16/07/04072101.htm
- [11] 田中克己, 下條真司: 動画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
ビデオ動画の扱い -, 情報処理学会 アドバンスト・データベース・システム・シンポジウム
ADBS'93, pp.55-62 (December 1993)
- [12] 大本英徹, 田中克己: ビデオデータベースの概念モデリングとビジュアルインターフェイスにつ
いて, 電子情報通信学会 データ工学研究会 DE-89-50, pp.17-24 (March 1990)
- [13] 大本英徹, 渥美純, 田中克己: A Video Object Model and Its Application to a Video Database
System, 電子情報通信学会 データ工学研究会 DE-90-28, pp.33-40 (December 1990)

- [14] 大本英徹, 奥田太郎, 田中克己: 空間的・時間的オブジェクトの包含関係に基づく継続機構とその物理的記憶構造, 情報処理学会 データベース・システム研究会 DBS-84-26, pp.219-228 (July 1991)
- [15] 大本英徹, 田中克己: マルチメディアデータベースにおける連続オブジェクトのモデリング, 日本ソフトウェア科学会 オブジェクト指向計算ワークショップ WOOC'92, pp.149-164 (July 1992)
- [16] Eitetsu Oomoto and Katsumi Tanaka: OVID: Design and Implementation of a Video-Object Database System, IEEE Transactions on Knowledge and Data engineering, Volume 5, Number 4, pp.629-643 (August 1993)
- [17] 大本英徹, 田中克己: 伝播ビュー: オブジェクト指向データベースにおける半順序関係に基づくインスタンス間継承機構, 情報処理学会論文誌 Volume 34, Number 8, pp.1802-1819 (August 1993)
- [18] 原良憲, 的場ひろし, 平田恭二, 高野元, 杉渕史子, 松浦宏, 中村克男, 引間晃: ハイパーメディアプラットフォーム”雅(みやび)”の概要, 情報処理学会 データベースシステム研究会 DBS-90-4, pp.29-38 (September 1992)
- [19] 原良憲: ハイパーメディア構築・利用ツール”雅”-システム概要-, 情報処理学会 第46回全国大会 7G-6, pp.4-225-4-226 (March 1993)
- [20] 平田恭二, 原良憲: ハイパーメディアシステム「雅」におけるメディアナビゲーション方式, 情報処理学会 アドバンスト・データベース・システム・シンポジウム ADBS'93, pp.17-26 (December 1993)
- [21] Kyoji Hirata, Yoshinori Hara, Naoki Shibata and Fusako Hirabayashi: Media-based Navigation for Hypermedia Systems, Hypertext'93 Proceedings, pp.159-173 (November 1993)
- [22] 高野元, 的場ひろし, 原良憲: ビデオ・ハイパーメディアのナビゲーション方式, 計測自動制御学会 第8回ヒューマンインタフェースシンポジウム, pp.607-612 (October 1992)
- [23] 川崎成人, 平田恭二, 高野元, 原良憲: ハイパーメディア「雅」における分散動画実装方式, 情報処理学会 第49回全国大会 1W-4, pp.4-189-4-190 (September 1994)
- [24] 平田恭二, 川崎成人, 原良憲: 動画ハイパーメディアシステム「雅」における電子魚図鑑システムの構築, 情報処理学会 第49回全国大会 1W-7, pp.4-195-4-196 (September 1994)
- [25] 平田恭二, 川崎成人, 高野元, 原良憲: ネットワーク環境下における動画ハイパーメディア実装方式, 情報処理学会 アドバンスト・データベース・システム・シンポジウム ADBS'94, pp.165-173 (December 1994)
- [26] 川崎成人, 平田恭二, 岩崎未知, 高野元, 原良憲: ハイパーメディアオンデマンド, NEC 技報, Volume 49, Number 1, pp.145-150 (January 1996)
- [27] 前田雅人, 川崎成人: 電子魚図鑑, NEC 技報, Volume 49, Number 1, pp.102-106 (January 1996)

- [28] Yoshinori Hara, Kyoji Hirata, Hajime Takano and Shigehito Kawasaki: Hypermedia Database “Himotoki” and Its Applications, Proceedings of the Twelfth International Conference on Data Engineering ICDE1996, pp.372-379 (February 1996)
- [29] Kyoji Hirata, Yoshinori Hara, Hajime Takano and Shigehito Kawasaki: Content-oriented Integration in Hypermedia Systems, Hypertext’96, The Seventh ACM Conference on Hypertext, pp.11-21 (March 1996)
- [30] Kyoji Hirata, Sougata Mukherjea, Yusaku Okamura, Wen-Syan Li and Yoshinori Hara: Object-based Navigation: An Intuitive Navigation Style for Content-oriented Integration Environment, Hypertext 97, The Eighth ACM Conference on Hypertext, pp.75-86 (April 1997)
- [31] 佐藤哲司, 片岡良治, 井上潮: ビデオリアリティ: 映像を用いた情報検索手法の高度化, 情報処理学会 データベースシステム研究会 DBS-99-38, pp.281-284 (July 1994)
- [32] 片岡良治, 紺谷精一, 三浦史光, 佐藤哲司, 井上潮: 実写映像を用いたマルチメディア情報システム「ビデオリアリティ」の構築, 電子情報通信学会 データ工学研究会 DE-94-78, pp.1-8 (February 1995)
- [33] 坂田哲夫, 小島明, 佐藤哲司: 映像散策のためのハイパービデオの提案, 電子情報通信学会論文誌 (D-II) Volume J78-D-II, Number 7, pp.1-8 (July 1995)
- [34] 坂田哲夫, 木原民雄, 小島明, 佐藤哲司: 映像散策のためのビデオハイパーモデルの提案, 電子情報通信学会 データ工学研究会 DE-95-35, pp.65-72 (July 1995)
- [35] 岩淵明, 片岡良治, 平野泰宏, 花籠靖: ビデオハイパーメディア技術, NTT R & D, Volume 45, Number 1, pp.9-14 (January 1996)
- [36] 佐藤哲司, 坂田哲夫, 小島明: ビデオハイパーメディア構築支援技術, NTT R & D, Volume 45, Number 1, pp.15-20 (January 1996)
- [37] Tetsuo Sakata, Akira Kojima and Tetsuji Sato: Hyper Video: Video Based Hypermedia for Walk-Through Applications, Proceedings of Multimedia Japan ’96, pp.148-154 (March 1996)
- [38] 花籠靖, 小島明, 佐藤哲司: ビデオハイパーメディアによるビジュアル電子図書館の構築, 情報処理学会 データベースシステム研究会 DBS-107-3, pp.17-24 (March 1996)
- [39] 岸田義勝, 木原民雄, 平野泰宏, 岩淵明, 寺中勝美: ビデオハイパーメディアの観光案内システム「TakeMe」への応用, 情報処理学会 マルチメディア通信と分散処理研究会 DPS-76-10, pp.55-60 (May 1996)
- [40] 坂田哲夫, 佐藤哲司: ビデオハイパーメディアにおけるシナリオ構造の拡張, 情報処理学会 データベースシステム研究会 DBS-109-39, pp.233-238 (July 1996)
- [41] 平野泰宏, 岩淵明, 花籠靖, 寺中勝美: 映像散策のためのビデオハイパーメディア - インタラクティブプレーヤの映像表示方式, 情報処理学会 マルチメディア通信と分散処理ワークショップ, pp.221-228 (October 1996)

- [42] 坂田哲夫, 柴垣斉, 佐藤哲司: 映像散策のためのビデオハイパーメディア - クライアントサーバ型 VHM のアーキテクチャ, 情報処理学会 マルチメディア通信と分散処理研究会 DPS-80-4, pp.19-24 (January 1997)
- [43] 平野泰宏, 坂田哲夫, 星隆司, 柴垣斉: 映像散策のためのビデオハイパーメディア VHM 統合開発環境の全体構想, 情報処理学会 マルチメディア通信と分散処理研究会 DPS-80-5, pp.25-30 (January 1997)
- [44] 中村仁之輔, 宮本昌征, 千原誠, 岩淵明, 平野泰宏: ビデオハイパーメディアを適用した駅向けマルチメディア情報案内システム, 情報処理学会 マルチメディア通信と分散処理研究会 DPS-80-6, pp.31-36 (January 1997)
- [45] 山本俊司, 池田哲夫, 星隆司, 柴垣斉, 坂田哲夫, 長岡慶三, 中村知靖: ビデオハイパーメディア (VHM) の教育分野への応用 - 遠隔マルチメディアテストシステムへの応用, 情報処理学会 マルチメディア, 分散, 協調とモバイルワークショップ DiCoMo'97, pp.347-352 (July 1997)
- [46] 池田哲夫, 平野泰宏, 湯口徹, 星隆司: マルチメディアアプリケーションのデータモデルと開発環境の考察, 情報処理学会論文誌, Volume 40, Number 1, pp.2-12 (January 1999)
- [47] 田中譲: IntelligentPad とミームメディア, 電子情報通信学会論文誌 D-II, Volume J79-D-II, Number 4, pp.451-459 (April 1996)
- [48] 田中譲: ミームメディア・アーキテクチャ IntelligentPad とその応用, 情報処理学会誌, Volume 38, Number 3, pp.222-231 (March 1997)
- [49] 田中一生, 田中譲: ハイパームービーのアーキテクチャとその応用, 情報処理学会 情報メディア研究会 IM-28-16, pp.73-77 (November 1996)
- [50] 田中一生, 田中譲: Intelligent Pad におけるハイパームービー・フレームワーク, 日本ソフトウェア科学会 インタラクティブシステムとソフトウェアに関するワークショップ WISS'97, pp.9-14 (December 1997)
- [51] 藤本英明, 吉岡信夫, 大村皓一: クリックブルムービーとそのオーサリングツール, 情報処理学会 ヒューマンインタフェース研究会 HI-91-8, pp.49-56 (October 2000)
- [52] Minoru Etoh: Promotion of block matching: Parametric representation for motion estimation, Proceedings of the 14th International Conference on Pattern Recognition, pp.282-285 (August 1998)
- [53] Yoshihisa Adachi, Minoru Asada and Takayuki Nakamura: Correlation-Based Visual Tracking enhanced by Affine Motion Description, Proceedings Workshop on Machine Vision and Applications MVA'96, pp.75-78 (November 1996)
- [54] Berthold K.P. Horn and Brian G. Schnuck: Determining Optical Flow, Artificial Intelligence, Volume 17, Number 1, pp.185-203 (August 1981)
- [55] J. L. Barron, D. J. Fleet and S. S. Beauchemin: Performance of optical flow techniques, International Journal of Computer Vision, Volume 12, Number 1, pp.43-77 (February 1994)

- [56] P.J. Burt, C. Yen and X. Xu: Local correlation measures for motion analysis: A comparative study, Proceedings Conference on Pattern Recognition and Image Processing PRIP'82, pp.269-274 (June 1982)
- [57] S.L. Tanimoto: Template matching in pyramids, Computer Graphics and Image Processing, Volume 16, Number 4, pp.356-369 (August 1981)
- [58] 金子敏充, 堀修: ロバスト統計を用いたハイパーメディア作成のための高速移動物体追跡法, 情報処理学会 コンピュータビジョンとイメージメディア研究会 CVIM-113-1, pp.1-8 (November 1998)
- [59] 金子敏充, 堀修: 小領域のブロックマッチングを複数用いたロバストなオブジェクト追跡法, 電子情報通信学会論文誌 D-II, Volume J85-D-II, Number 7, pp.1188-1200 (July 2002)
- [60] M. Kass, A. Witkin and D. Terzopoulos: Snakes: Active contour modules, International Journal of Computer Vision, Volume 1, Number 4, pp.321-331 (January 1988)
- [61] 岡田重樹, 山本喜一: 動画ハイパーメディアシステムにおけるオブジェクト追跡法, 日本ソフトウェア科学会 第14回大会 B4-4, pp.197-200 (September 1997)
- [62] 金次保明, 苗村昌秀, 境田慎一, 鄭文涛, 三須俊彦: オブジェクト連動データ放送システムの検討, 2001年電子情報通信学会総合大会 A-16-28, p.351 (March 2001)
- [63] 三須俊彦, 苗村昌秀, 境田慎一, 鄭文涛, 金次保明: 複数情報の融合によるサッカー選手のロバストな追跡法, 電子情報通信学会 画像工学研究会 IE2001-47, pp.23-30 (July 2001)
- [64] 金次保明, 苗村昌秀, 鄭文涛, 三須俊彦: オブジェクト連動データ放送システムの検討, 2001年映像情報メディア学会年次大会, pp.176-177 (August 2001)
- [65] 鄭文涛, 三須俊彦, 苗村昌秀, 金次保明: 固定全景映像カメラを併用した映像オブジェクト抽出法, 電子情報通信学会ソサイエティ大会 D-12-40, pp.215 (September 2001)
- [66] 鄭文涛, 苗村昌秀, 三須俊彦, 和泉吉則, 金次保明: 2台のカメラを用いた動オブジェクト抽出手法: 電子情報通信学会 2001年度映像メディア処理シンポジウム IMPS2001 (November 2001)
- [67] 苗村昌秀, 鄭文涛, 三須俊彦, 和泉吉則, 金次保明: 映像オブジェクトに連動した放送サービス「情報ナビゲーション型未来放送」, 2001年映像情報メディア学会 冬季大会 4-10, pp.80 (December 2001)
- [68] 金次保明, 長島聖子, 山田光穂, 清水俊宏: 指さしポインターにおけるカーソル位置の特定法, 電子情報通信学会 画像工学研究会 IE-2001-181, pp.55-60 (January 2002)
- [69] 三須俊彦, 鄭文涛, 苗村昌秀, 和泉吉則, 金次保明: 情報ナビゲーション型未来放送のオーサリングツール, 映像情報メディア学会 放送方式研究会 BCS2002-3, pp.13-18 (February 2002)
- [70] 鄭文涛, 三須俊彦, 苗村昌秀, 和泉吉則, 金次保明: 複数カメラの協調動作による動オブジェクトの抽出, 電子情報通信学会 画像工学研究会 IE-2001-235, pp.23-30 (March 2002)

- [71] 金次保明, 苗村昌秀, 三須俊彦: オブジェクト連動データ放送システムの TS 伝送実験, 電子情報通信学会総合大会 D-11-75, p.75 (March 2003)
- [72] 長島聖子, 金次保明: オブジェクト連動データ放送システムにおける MIDI データの伝送実験, 電子情報通信学会総合大会 D-11-76, p.765 (March 2003)
- [73] 鄭文涛, 三須俊彦, 苗村昌秀, 和泉吉則, 金次保明: 複数カメラの協調動作による動オブジェクトの抽出, 電子情報通信学会論文誌 D-II, Volume J86-D-II, Number 3, pp.397-408 (March 2003)
- [74] 金次保明, 三須俊彦, 高橋正樹, 合志清一, 苗村昌秀: オブジェクト連動データ放送システムの開発とその記述方式, 情報処理学会 オーディオビジュアル複合処理研究会 2003-AVM-41, pp.17-22 (August 2003)
- [75] 金次保明, 三須俊彦, 高橋正樹, 合志清一, 苗村昌秀: オブジェクト連動データ放送システムの開発, 映像情報メディア学会 放送技術研究会 BCT2003-43, pp.1-4 (October 2003)
- [76] 黄炯韜, 金次保明: オブジェクト連動データ放送サービス用関連データ入力データベースの構築及び検証, 映像情報メディア学会 2003 年冬季大会 (December 2003)
- [77] Ysuaki Kanatsugu, Toshihiko Misu, Masaki Takahashi, Seiichi Gohshi: The Development of an Object-linked Broadcasting System, International Workshop on Advanced Image Technology 2004 IWAIT2004 (January 2004)
- [78] 青柳滋己, 佐藤孝治, 高田敏弘, 廣津登志夫, 尾内理紀夫: 音データへのリンク情報付加法の検討, 日本ソフトウェア科学会 第 14 回大会, pp.189-192 (September 1997)
- [79] 佐藤孝治, 高田敏弘, 青柳滋己, 廣津登志夫, 尾内理紀夫. WWW 上の映像ブラウジング機構の提案と実現, 日本ソフトウェア科学会 第 14 回大会, pp.193-196 (September 1997)
- [80] 高田敏弘, 青柳滋己, 佐藤孝治, 廣津登志夫, 尾内理紀夫: WWW における連続メディアへのリンク情報の附与, 画像電子学会 第 9 回メディア統合技術研究会 (October 1997)
- [81] 高田敏弘, 青柳滋己, 佐藤孝治, 廣津登志夫, 尾内理紀夫: Cmew: 連続メディアと WWW の統合, Proceedings of the Japan WWW Conference'97 (December 1997)
- [82] 廣津登志夫, 高田敏弘, 青柳滋己, 佐藤孝治, 尾内理紀夫: マルチメディアデータのための Plug-in Annotation Server の設計と実装, Proceedings of the Japan WWW Conference'97 (December 1997)
- [83] 高田敏弘, 有村浩一, 尾内理紀夫: 応答性/対話性を重視したビデオマークアップエディタ, 日本ソフトウェア科学会 インタラクティブシステムとソフトウェアに関するワークショップ WISS'97, pp.15-22 (December 1997)
- [84] 青柳滋己, 佐藤孝治, 高田敏弘, 廣津登志夫, 尾内理紀夫: リンク情報付き音データへのアクセス法の検討, 日本ソフトウェア科学会 インタラクティブシステムとソフトウェアに関するワークショップ WISS'97, pp.1-8 (December 1997)

- [85] 佐藤孝治, 高田敏弘, 青柳滋己, 廣津登志夫, 尾内理紀夫: WWWにおける映像データへのリンク情報付与のための枠組み, 日本ソフトウェア科学会 インタラクティブシステムとソフトウェアに関するワークショップ WISS'97, pp.23-30 (December 1997)
- [86] 廣津登志夫, 高田敏弘, 青柳滋己, 佐藤孝治, 菅原俊治, 尾内理紀夫: Multimedia Annotation ServerのためのPlug-in Toolkitの設計と実装, 日本ソフトウェア科学会 第1回プログラミングおよび応用のシステムに関するワークショップ SPA'98, (March 1998)
http://spa.jsst.or.jp/1998/019_hirotsu/index.html
- [87] Toshihiro Takada, Koji Sato, Shigemi Aoyagi, Toshio Hirotsu and Toshiharu Sugawara: Cmew: Integrating Continuous Media with the Web, Proceedings of the 3rd Annual Multimedia Technology and Applications Conference MTAC'98, pp.136-140 (September 1998)
- [88] 高田敏弘, 佐藤孝治, 青柳滋己, 廣津登志夫, 菅原俊治: Cmew: 連続メディアを起点とするインターネットリンク機構, 日本ソフトウェア科学会 第15回大会, pp.73-76 (September 1998)
- [89] 佐藤孝治, 高田敏弘, 青柳滋己, 廣津登志夫, 尾内理紀夫, 菅原俊治: 連続メディアとWWWの統合システム Cmewにおけるメディアデータ構造化のためのシナリオ制御機構, 日本ソフトウェア科学会 第2回プログラミングおよび応用のシステムに関するワークショップ SPA'99, (March 1999)
http://spa.jsst.or.jp/1999/006_sato.pdf
- [90] 佐藤孝治, 高田敏弘, 青柳滋己, 廣津登志夫, 菅原俊治, 尾内理紀夫: 連続メディアとWWWの統合システム Cmewにおけるシナリオ制御機構の実現, コンピュータソフトウェア, Volume 16, Number 3, pp. 47-56 (May 1999)
- [91] Shigemi Aoyagi, Toshihiro Takada, Koji Sato, Toshio Hirotsu and Toshiharu Sugawara: Cmew/A: An Access Method for Audio Data with Link Information, Proceedings of the Sixth International Conference on Distributed Multimedia Systems DMS'99, pp. 26-33 (July 1999)
- [92] Koji Sato, Toshihiro Takada, Shigemi Aoyagi, Toshio Hirotsu and Toshiharu Sugawara: Dynamic Multimedia Integration with the WWW, Proceedings of the 1999 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing PACRIM1999, pp. 448-451 (August 1999)
- [93] Toshio Hirotsu, Toshihiro Takada, Shigemi Aoyagi, Koji Sato and Toshiharu Sugawara: Cmew/U — A Multimedia Web Annotation Sharing System, Proceedings of the IEEE Region 10 Conference TENCON'99, pp. 356-359 (September 1999)
- [94] 青柳滋己, 高田敏弘, 佐藤孝治, 廣津登志夫, 菅原俊治, 尾内理紀夫: 音データへのリンク情報の埋め込み法, コンピュータソフトウェア, Volume 16, Number 6, pp.13-23 (December 1999)
- [95] 廣津登志夫, 高田敏弘, 青柳滋己, 佐藤孝治, 菅原俊治: Web アノテーション共有システム Cmew/Uの設計と実装, 情報処理学会論文誌, Volume 42, Number 10, pp. 2466-2475 (October 2001)

- [96] 藤川和利, 下條真司, 松浦敏雄, 西尾章治郎, 宮原秀夫: オブジェクト指向に基づくハイパーメディアシステム Harmony の構築, 電子情報通信学会論文誌 D-I, Volume J75-D-I, Number 11, pp.1015-1024 (November 1992)
- [97] 藤川和利, 下條真司, 松浦敏雄, 西尾章治郎, 宮原秀夫: 分散型ハイパーメディアシステム Harmony における情報間同期機構の実現, 電子情報通信学会論文誌 D-I, Volume J76-D-I, Number 9, pp.473-483 (September 1993)
- [98] 呉世雄, 島村栄, 椿野慎治, 藤川和利, 下條真司, 松浦敏雄, 宮原秀夫: 分散型マルチメディアシステム Symphony における品質保証機構, 情報処理学会 マルチメディア通信と分散処理研究会 DPS-70-7, pp.39-44 (May 1995)
- [99] 呉世雄, 藤川和利, 松浦敏雄, 下條真司, 宮原秀夫: 分散型マルチメディアシステム Symphony における QoS 指定を含んだシナリオ記述言語の設計, 電子情報通信学会論文誌 B-I, Volume J79-B-I, Number 5, pp.329-337 (May 1996)
- [100] W3C: W3C Synchronized Multimedia Home page:
<http://www.w3c.org/AudioVideo/>
- [101] RealNetworks: RealNetworks Inc. Home Page:
<http://www.realnetworks.com/>
- [102] Apple Computer: Apple - QuickTime Home page,
<http://www.apple.com/quicktime/index.html>
- [103] 姉齒康: QuickTime Guide Book, ビー・エヌ・エヌ (October 2002)
- [104] 有川正俊, 村尾真洋, 岡村耕二: 遠隔定点観測カメラを用いた拡張/減少ライブビデオ応用の開発・実験, 電子情報通信学会 マルチメディア・仮想環境基礎研究会 MVE98-96, pp.77-84 (February 1999)
- [105] 岡村耕二, 村尾真洋, 有川正俊: ネットワーク透過型仮想ビデオキャプチャデバイスの設計, 情報処理学会 マルチメディア通信と分散処理研究会 DPS-91-3, pp.13-18 (January 1999)
- [106] P. Bruce Berra, Forouzan Golshani, Rajiv Mehrotra, and Olivia R. Liu Sheng: Guest Editors' Introduction Multimedia Information Systems, IEEE Transactions on Knowledge and Data Engineering, Volume 5, Number 4, pp.545-550 (August 1993)
- [107] Wendy E. Mackay: EVA: An experimental video annotator for symbolic analysis of video data, ACM SIGCHI Bulletin, Volume 21, Number 2, pp.68-71 (October 1989)
- [108] Wendy E. Mackay and Glorianna Davenport: Virtual video editing in interactive multimedia applications, Communications of the ACM, Volume 32, Number 7, pp.802-810 (July 1989)
- [109] 阿倍博信, 吉良賢治, 高梨郁子, 永沼和智: 応用アプリケーションのためのビデオオブジェクト管理システムの構築, 情報処理学会 第 49 回全国大会 3W-2, pp.4-219-4-220 (September 1994)

- [110] Hironobu Abe and Koji Wakimoto: Architecture and Implementation of a Video Object Retrieval Environment, Second IASTED/ISMM International Conference DISTRIBUTED MULTIMEDIA SYSTEMS AND APPLICATIONS, pp.43-49 (August 1995)
- [111] Taiji Tsuchida, Hironobu Abe and Mikio Sasaki: HyperFrame: A hypermedia framework for integration of engineering applications, ACM 11th International Conference on System Documentation SIGDOC'93, pp.345-355 (October 1993)
- [112] 阿倍博信, 土田泰治, 佐々木幹郎: アプリケーション統合を目的としたハイパーメディアの構築と実システムへの適用, 情報処理学会 アドバンスト・データベース・システム・シンポジウム ADBS'93, pp.37-44 (December 1993)
- [113] V. Micheael Bove, Jr., Jonathan Dakss, Edmond Chalom and Stefan Agamanolis: Hyper-linked television research at the MIT Media Laboratory, IBM Systems Journal, Volume 39, Number 3-4, pp.470-478 (2000)
- [114] Nitin "Nick" Sawhney, David Balcom and Ian Smith: HyperCafe: Narrative and Aesthetic Properties of Hypertext, Hypertext'96, The Seventh ACM Conference on Hypertext, pp.1-10 (March 1996)
- [115] 神田準史郎, 阿倍博信, 脇本浩司: 動画ハイパーメディア制作支援のためのオブジェクト追跡方法, 情報処理学会 第52回全国大会 5G-10, pp.2-273-2-274 (March 1996)
- [116] 阿倍博信, 神田準史郎, 脇本浩司: 動画画像解析を利用した動画ハイパーメディアシステムの構築, 電子情報通信学会 パターン認識・理解研究会 PRU95-238, pp.21-26 (March 1996)
- [117] 神田準史郎, 脇本浩司, 阿倍博信, 田中聡: 動画ハイパーメディアシステムにおけるアンカー自動設定方法とその評価, 電子情報通信学会 パターン認識・メディア理解研究会 PRMU96-146, pp.41-46 (January 1997)
- [118] 脇本浩司, 神田準史郎, 阿倍博信, 田中聡: 動画画像解析を応用した対話型ビデオ制作システム, 画像センシング技術研究会 第3回画像センシングシンポジウム, pp.47-52 (June 1997)
- [119] Junshiro Kanda, Koji Wakimoto, Hironobu Abe and Satoshi Tanaka: Video Hypermedia authoring using automatic object tracking, IS & T/SPIE's 10th International Symposium Electronic Imaging: Science and Technology, pp.108-115 (January 1998)
- [120] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach and T. Berners-Lee : Hypertext Transfer Protocol – HTTP/1.1, RFC 2616 (June 1999)
- [121] Microsoft: Microsoft Windows Media Home page:
<http://www.microsoft.com/windowsmedia/>
- [122] 映像情報メディア学会 (編): デジタル放送ハンドブック, オーム社 (June 2003)
- [123] 外村佳伸: インタラクティブ映像技術の展望, 電子情報通信学会 マルチメディア・仮想環境基礎研究会 技術報告 MVE99-69, pp.35-41 (February 2000)

- [124] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson : RTP: A Transport Protocol for Real-Time Applications, RFC 3550 (July 2003)
- [125] 佐藤純, 橋本浩二, 高坂幸春, 柴田義孝 : メディア同期を考慮したマルチメディア転送プロトコル, 情報処理学会 マルチメディア通信と分散処理研究会 研究報告 DPS-88-13, pp.73-78 (April 1998)
- [126] 端山貴也, 清木康 : 協調型同期方式によるメディア間同期とその実現方式, 情報処理学会論文誌, Vol.39, No.3, pp.704-715 (March 1998)
- [127] 阿倍博信, 泉丙完, 山田淳, 松本佳宏, 宮木和徳: 拡張イベントモデルによる映像と付加情報の同期配信システム, 情報処理学会/電子情報通信学会 第2回情報科学技術フォーラム FIT2003 M-028, pp.4-95-4-96 (September 2003)
- [128] 電波産業会編: BS/広帯域CS デジタル放送運用規定技術資料 ARIB TR-B15 3.0版 (July 2004)
- [129] 横里純一, 安田哲, 清水明宏: 動画ハイパーメディアの初等教育用マルチメディアコンテンツへの適用に関する一考察, 情報処理学会 第61回全国大会 6M-1, pp.3-81-82 (October 2000)
- [130] 三菱電機株式会社: 高知学校インターネットI 目的適応型マルチキャスト通信ネットワークアーキテクチャの研究開発,
<http://www.kochi-isin.jp/results/h11/katsudou/05gakko.htm>

研究業績

論文

1. Hironobu Abe and Koji Wakimoto: Content-based Management of Video in a Multimedia Authoring Environment, Journal of Multimedia Tools And Applications: Special Issue on Highlighting Industrial Projects, Volume 2, Number 3, pp.199-214 (May 1996)
2. 阿倍博信, 横里純一, 脇本浩司, 泉丙完, 厚井裕司: マルチメディア教育を目的とした動画ハイパーメディアシステム: VisualSHOCK MOVIE, 画像電子学会誌 創立 30 周年記念 次世代マルチメディア・コンテンツ流通技術論文特集号, Volume 30, Number 5, pp.565-574 (September 2001)
3. 阿倍博信, 脇本浩司, 神田準史郎, 重野寛, 岡田謙一: 自動追跡とマニュアル編集の組み合わせによる動画ハイパーメディアオーサリングの効率化, 画像電子学会誌 2003 年度論文特集号, Volume 32, Number 5, pp.635-644 (September 2003)
4. 阿倍博信, 福田雅裕, 山田淳, 松本佳宏, 重野寛, 岡田謙一: インターネット映像配信サービスのための映像と付加情報の同期配信方式, 情報処理学会論文誌 新時代の分散処理とネットワーク (Web サービスと P2P) 特集, Volume 46, Number 2 (February 2005) [採録決定]

国際会議

1. Taiji Tsuchida, Hironobu Abe and Mikio Sasaki: HyperFrame: A hypermedia framework for integration of engineering applications, ACM 11th International Conference on System Documentation SIGDOC'93, pp.345-355, Waterloo Canada (October 1993)
2. Hisayuki Masui, Hironobu Abe and Yoshihisa Udagawa: Real Time Distributed Concurrent System with Non-Real Time Distributed Argument Support Interface (ARG VIEW) for Concurrent Engineering, IEEJ Japan Industry Applications Society Conference JIASC'94, pp.183-188, Matsuyama Japan (August 1994)
3. Hironobu Abe and Koji Wakimoto: Architecture and Implementation of a Video Object Retrieval Environment, Second IASTED/ISMM International Conference DISTRIBUTED MULTIMEDIA SYSTEMS AND APPLICATIONS, pp.43-49, Stanford USA (August 1995)
4. Junshiro Kanda, Koji Wakimoto, Hironobu Abe and Satoshi Tanaka: Video Hypermedia authoring using automatic object tracking, IS & T/SPIE's 10th International Symposium Electronic Imaging: Science and Technology, pp.108-115, San Jose USA (January 1998)

5. Takashi Kan, Hironobu Abe, Toyoshi Makino, Heikan Izumi and Norio Shiratori: HDTV-PC for Media Convergence, IEEE International Conference on Consumer Electronics ICCE2002, pp.24-25, Los Angeles USA (June 2002)
6. Hironobu Abe, Junshiro Kanda, Koji Wakimoto, Hiroshi Shigeno and Ken-ichi Okada: Video hypermedia authoring for video delivering service based on a combination of automatic tracking and manual editing, 5th International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS2004, in CD-ROM Proceedings, Lisbon Portugal (April 2004)

研究会・シンポジウム

1. 小高俊之, 阿倍博信, 横山光男, 松下温: 平均演算子を利用したあいまい要求処理, 情報処理学会 データベース・システム研究会 第76回研究会 90-DBS-76-2, 機械振興会館 (東京都港区) (March 1990)
2. 阿倍博信, 土田泰治, 坂下善彦: 非同期分散型議論支援インタフェース: ArgView, 情報処理学会 グループウェア研究グループ 第1回研究会 92-GW-1-7, pp.43-50, 慶應義塾大学理工学部 (神奈川県横浜市) (June 1992)
3. 阿倍博信, 土田泰治, 佐々木幹郎: アプリケーション統合を目的としたハイパーメディアの構築と実システムへの適用, 情報処理学会 アドバンスト・データベース・システム・シンポジウム ADBS'93, pp.37-44, 工学院大学 (東京都新宿区) (December 1993)
4. 阿倍博信, 神田準史郎, 脇本浩司: 動画像解析を利用した動画ハイパーメディアシステムの構築, 電子情報通信学会 パターン認識・理解研究会 PRU95-238, pp.21-26, 千葉大学工学部 (千葉県千葉市) (March 1996)
5. 神田準史郎, 脇本浩司, 阿倍博信, 田中聡: 動画ハイパーメディアシステムにおけるアンカー自動設定方法とその評価, 電子情報通信学会 パターン認識・メディア理解研究会 PRMU96-146, pp.41-46, 京都大学工学部 (京都府京都市) (January 1997)
6. 脇本浩司, 神田準史郎, 阿倍博信, 田中聡: 動画像解析を応用した対話型ビデオ制作システム, 画像センシング技術研究会 第3回画像センシングシンポジウム, pp.47-52, パシフィコ横浜 (神奈川県横浜市) (June 1997)
7. 阿倍博信, 泉丙完, 菅隆志, 佐香清二, 厚井裕司: 通信放送融合端末とプログラムガイド統合システム, 情報処理学会 放送コンピューティング研究グループ 第1回研究会 2001-BCCgr-1-1, pp.1-8, 情報処理学会 (東京都港区) (February 2002)
8. 横里純一, 阿倍博信, 加藤嘉明: 動画ハイパーメディアコンテンツストリーム配信方式の検討, 情報処理学会 マルチメディア, 分散, 協調とモバイルシンポジウム DICOMO2002, pp.149-152, 桂川シーサイドホテル (静岡県田方郡) (July 2002)
9. 阿倍博信, 福田雅裕, 虻川雅浩, 泉丙完: 衛星デジタルデータ放送を用いた IPv6 通信システムの構築, 情報処理学会 マルチメディア, 分散, 協調とモバイルシンポジウム DICOMO2003, pp.313-316, ニュー阿寒ホテル (北海道阿寒郡) (June 2003)

口頭発表

1. 野村恭彦, 阿倍博信, 小高俊之, 平岩真一, 横山光男, 松下温: あいまい要求のための検索システム, 情報処理学会 第 40 回全国大会 1H-3, pp.775-776, 早稲田大学理工学部 (東京都新宿区) (March 1990)
2. 平岩真一, 阿倍博信, 小高俊之, 野村恭彦, 横山光男, 松下温: 拡張演算子のあいまい要求処理への適用, 情報処理学会 第 40 回全国大会 1H-4, pp.777-778, 早稲田大学理工学部 (東京都新宿区) (March 1990)
3. 阿倍博信, 土田泰治, 坂下善彦: 電子会議システムのユーザインタフェースに関する検討, 情報処理学会 第 44 回全国大会 7J-2, pp.5-263-5-264, 明治大学理工学部 (神奈川県川崎市) (March 1992)
4. 阿倍博信, 土田泰治, 中岡佐知子, 水野忠則: 技術文書管理システムにおける複合ブラウジング方式の実現, 情報処理学会 第 46 回全国大会 8G-7, pp.4-247-4-248, 工学院大学 (東京都新宿区) (March 1993)
5. 佐々木幹郎, 阿倍博信, 土田泰治: ハイパーメディアリンク機構を有したアプリケーション統合システム「HyperFrame」, 情報処理学会 第 48 回全国大会 5E-4, pp.4-175-4-176, 東京理科大学 (千葉県野田市) (March 1994)
6. 吉良賢治, 高梨郁子, 永沼和智, 阿倍博信, 野口誠二, 重住恒雄: マルチメディアを応用した専門教育訓練システム -その狙いと全体構成-, 情報処理学会 第 49 回全国大会 1E-2, pp.6-217-6-218, 北海道大学教養部 (北海道札幌市) (September 1994)
7. 永沼和智, 吉良賢治, 高梨郁子, 阿倍博信, 野口誠二, 重住恒雄: マルチメディアを応用した専門教育訓練システム -シミュレータ実現方式-, 情報処理学会 第 49 回全国大会 1E-3, pp.6-219-6-220, 北海道大学教養部 (北海道札幌市) (September 1994)
8. 阿倍博信, 吉良賢治, 高梨郁子, 永沼和智: 応用アプリケーションのためのビデオオブジェクト管理システムの構築, 情報処理学会 第 49 回全国大会 3W-2, pp.4-219-4-220, 北海道大学教養部 (北海道札幌市) (September 1994)
9. 神田準史郎, 阿倍博信, 脇本浩司, 平田孝雄: ビデオオブジェクト管理システムにおけるハイパーリンク機能, 電子情報通信学会 1995 年総合大会 D-99, p.105, 福岡工業大学 (福岡県福岡市) (March 1995)
10. 神田準史郎, 阿倍博信, 脇本浩司: 動画ハイパーメディア制作支援のためのオブジェクト追跡方法, 情報処理学会 第 52 回全国大会 5G-10, pp.2-273-2-274, 電気通信大学 (東京都調布市) (March 1996)
11. 阿倍博信, 泉丙完, 佐香清二, 厚井裕司: 通信と放送の融合環境における電子プログラムガイド統合方式の提案, 情報処理学会 第 63 回全国大会 4Y-01, pp.3-233-3-234, 山口大学吉田キャンパス (山口県山口市) (September 2001)

12. 阿倍博信, 泉丙完, 菅隆志, 佐香清二, 厚井裕司: 通信放送融合端末のためのプログラムガイド統合配信システム, 情報処理学会 第 64 回全国大会 6Z-04, pp.3-207-3-208, 東京電機大学鳩山キャンパス (埼玉県比企郡) (March 2002)
13. 横里純一, 阿倍博信, 加藤嘉明, 田鍋潤一郎, 清水明宏: 動画ハイパーメディアコンテンツストリーム配信に関する一検討, 電子情報通信学会 2002 年総合大会 D-15-24, p.213, 早稲田大学理工学部 (東京都新宿区) (March 2002)
14. 虻川雅浩, 阿倍博信, 湯川真紀, 田中功一: 衛星 IP 通信システムにおける帯域・経路制御の一考察, 電子情報通信学会 2002 年総合大会 B-6-77, p.91, 早稲田大学理工学部 (東京都新宿区) (March 2002)
15. 阿倍博信, 濱谷英次: 映像アノテーションによる講義映像を用いたマルチメディア教材オーサリング, 教育システム情報学会 第 28 回全国大会 C6-1, pp.303-304, 茨城大学 水戸キャンパス (茨城県水戸市) (August 2003)
16. 阿倍博信, 泉丙完, 山田淳, 松本佳宏, 宮木和徳: 拡張イベントモデルによる映像と付加情報の同期配信システム, 情報処理学会/電子情報通信学会 第 2 回情報科学技術フォーラム FIT2003 M-028, pp.4-95-4-96, 札幌学院大学 (北海道江別市) (September 2003)
17. 阿倍博信, 佐香清二, 濱谷英次: 映像アノテーションによる講義映像を用いた自主学習教材の制作と配信, 教育システム情報学会 第 29 回全国大会 TB-5, pp.25-26, 香川大学 幸町キャンパス (香川県高松市) (August 2004)
18. 三尾武史, 平松隆宏, 阿倍博信, 中島宏一: 遠隔映像監視システム用蓄積サーバの検討, 電子情報通信学会 2004 年ソサイエティ大会 B-8-7, p.213, 徳島大学 常三島キャンパス (徳島県徳島市) (September 2004)

解説・他

1. 阿倍博信: InterNote: ハイパーメディアフレームワークの注釈の共同作業支援への拡張, 情報処理学会誌 文献紹介 90-41, Volume 31, Number 12, pp.1691-1692 (December 1990)
2. 澤本潤, 臼井澄夫, 居駒哲夫, 阿倍博信: マルチメディア情報流通技術の現状と当社の取組, 三菱電機技報 マルチメディア情報流通技術小特集, Volume 75, Number 2, pp.42-45 (February 2001)
3. 土田泰治, 阿倍博信, 小林義人: 電子会議支援方式, 三菱電機技報 特許と新案, Volume 77, Number 4, p.64 (April 2003)
4. 福田雅裕, 阿倍博信, 虻川雅浩, 長谷川英司, 柴田清己, 本田文雄: 高度デジタルデータ放送技術の研究開発, 画像ラボ, Volume 14, Number 10, pp.18-21 (October 2003)
5. 阿倍博信, 川畑幸保, 上野幾朗: セキュリティ映像配信技術, 三菱電機技報 物理セキュリティ特集, pp.51-54, Volume 78, Number 8 (August 2004)

特 許

1. Junshiro Kanda, Hironobu Abe and Koji Wakimoto: Representative image display method, representative image display apparatus, and motion image search apparatus employing the representative image display apparatus, United States Patent 5,894,333, April 13, 1999 (April 1999)
2. 佐々木幹郎, 土田泰治, 阿倍博信: 情報管理・検索装置, 日本国特許 第 2930855 号, 1999 年 5 月 21 日 (May 1999)
3. 阿倍博信, 土田泰治, 小林義人: 電子会議支援方式, 日本国特許 第 2990919 号, 1999 年 10 月 15 日 (October 1999)
4. Hironobu Abe, Junshiro Kanda and Koji Wakimoto: Moving image anchoring apparatus and hypermedia apparatus which estimate the movement of an anchor based on the movement of the object with which the anchor is associated, United States Patent 5,970,504. October 19, 1999 (October 1999)
5. Hironobu Abe, Junshiro Kanda, Koji Wakimoto and Satoshi Tanaka: Moving image anchoring apparatus which estimates the movement of an anchor based on the movement of the object with which the anchor is associated utilizing a pattern matching technique , United States Patent 6,144,972. November 7, 2000 (November 2000)
6. 平島保彦, 阿倍博信, 神田準史郎, 前田岳, 吉川肇: 動画処理装置、動画アンカー表示方法、動画アンカー判断方法, 日本国特許 第 3214258 号, 2001 年 7 月 27 日 (July 2001)
7. 神田準史郎, 阿倍博信, 脇本浩司: 代表画像表示方法、代表画像表示装置およびこの装置を用いた動画検索装置, 日本国特許 第 3432348 号, 2003 年 5 月 23 日 (May 2003)
8. 阿倍博信, 神田準史郎, 脇本浩司, 田中聡: 動画アンカー設定装置, 日本国特許 第 3511278 号, 2004 年 1 月 16 日 (January 2004)