

メタデータの管理に基づくビデオデータベースの構成

黎 亜和[†] 田 中 譲[†]

本論文では、ビデオ映像データと、これに付加したメタデータをともに格納・管理することにより、ビデオの内容に関して検索条件を課し、これを満たす映像を検索できるビデオデータベースのアーキテクチャを提案する。ビデオの内容検索を実現するために、リファレンスフレームという概念を導入し、これに基づくビデオ映像の内容記述法を提案する。また、複数の映像オブジェクト間の空間関係に関する検索条件をビジュアルに直接操作可能なビジュアル検索要求言語を提案し、その実現について述べる。

A Video Database Architecture Based on the Management of Meta-data

YAHE LI[†] and YUZURU TANAKA[†]

This paper describes a video database architecture in which the content-based retrieval of video data can be processed using meta-data. In this architecture the video data was stored together with the meta-data attached to them. In order to implement the content-based retrieval of video, we introduce a concept of reference frames, and propose a new approach to represent video content, based on the reference frames. Furthermore, this paper describes the implementation of a visual query composition tool that helps us to specify retrieval conditions based on spatial relationships among video objects.

1. はじめに

近年、ビデオを計算機上で扱うことは珍しいことではなくなってきたが、これらのビデオリソースをデータベース化することによって、検索を可能にするにはいまだきわめて困難な状況にある¹⁾。ビデオデータは連続領域と時間軸を有し、映像中に現れるオブジェクトはいずれも分節化されていないという特性を持っているからである。ビデオデータに対して同一の枠組みで表現、蓄積、検索を行う手法はいまだ確立していない。

ビデオデータの管理とビデオデータのデータベース化に関して、現在までに提案されたアプローチは、構造化モデリングアプローチ^{2),3)}と階層化アプローチ⁴⁾の2種類に分類できる。前者では、ビデオを基本的な撮影単位であるカットに分割するのみである。これに対し後者では、カットをいくつかまとめて意味的に連続したシーンにまとめる。ビデオの内容検索を実現するためには、各ビデオ中に現れる意味のある映像オブ

ジェクトを分節化し、分節化した映像オブジェクトの属性値に対する索引を用意することが新たな手法として考えられる。既存の画像処理技術では映像中の映像オブジェクトを自動的に分節化することはそもそも困難である。映像のデータベース化に対する従来のアプローチでは、映像オブジェクトごとの分節化をしないで、カットといった単位に対して、それらの属性記述キーワードを付与し、これらの属性値に対する索引を用いて特定のカットを検索している。

ビデオ映像中に現れる意味のある映像オブジェクトに関する分節化情報とその属性値を、対象とするすべての映像オブジェクトに関してまとめた情報をここではメタデータと呼ぶ。本論文では、ビデオ映像とそのメタデータを、ファイルとデータベースを用いて格納し管理するビデオデータベースを提案する。メタデータを用いることにより、映像中に現れる意味のある映像オブジェクト名称や位置や大きさなどの属性に関して検索条件を指示することが可能になる。このようなビデオの内容検索を実現するために、リファレンスフレームという概念を導入し、これに基づくビデオ映像の内容記述法を提案する。さらに、複数の映像オブジェクト間の空間関係に関する検索条件をビジュアルに直

[†] 北海道大学知識メディア・ラボラトリー

Meme Media Laboratory, Hokkaido University

接操作可能なビジュアル検索要求言語を実現する。

本論文の構成は以下のとおりである。2章では、メタデータの管理・処理に基づくビデオデータベースシステムの構成を示す。3章ではリファレンスフレームの概念を導入し、これに基づくビデオ映像の内容記述法について説明する。4章では複数の映像オブジェクト間の空間関係をビジュアルに指定するビジュアル検索要求の実現法を述べる。5章では、提案した手法に基づき、RDBMSとIntelligentPadシステム⁵⁾を用いて実現したビデオデータベースシステムについて報告する。6章ではシステム実装について評価する。最後に、7章においてまとめる。

2. システムのアーキテクチャ

ビデオのメタデータは、フレーム番号や持続時間などを含むビデオの表現属性データと、ビデオで表現されている内容に関して記述した内容属性データの2種類に分けられる。このようなメタデータを用いることにより、ビデオ映像に対して内容に基づく検索を行うことができる。

ビデオ映像をそのままデータベースの操作対象とすることは困難であるが、メタデータの方はDBMSシステムに格納して、検索処理の対象とすることができる。我々は、メタデータを格納・管理し、その検索処理に基づいて、間接的にビデオ映像を管理・検索するビデオデータベースシステムのアーキテクチャを提案する。図1にこのアーキテクチャの構成を示す。以下にこのシステムアーキテクチャの特徴と機能について説明する。

- (1) DBMSシステムによりビデオのメタデータを管理し、ビデオデータは原データのBLOB(Binary Large Object)の形でアクセスする。ビデオのメタデータに対する検索処理の結果を用いて、ビデオ映像をアクセスする。これは我々が考案したシステムアーキテクチャの基本的な特徴である。
- (2) EditProcessorはビデオ映像ソースからビデオの属性データを抽出し、作成・編集のオーサリング過程を支援する。連続したビデオフレームシーケンスから、画像処理技術を用いて自動的にカット変わりを検出したり、特定の被写体が写っているフレームシーケンスを検出する方法に関しては、多くの研究があるが^{6),7)}、映像オブジェクトの分節化支援の機能は持たない。我々のシステムでは、ビデオデータベース作成者がツールを用いて半自動的にビデオ映像の分節化と分節化されたオブジェクトの属性データのオーサリングを行うことができる。

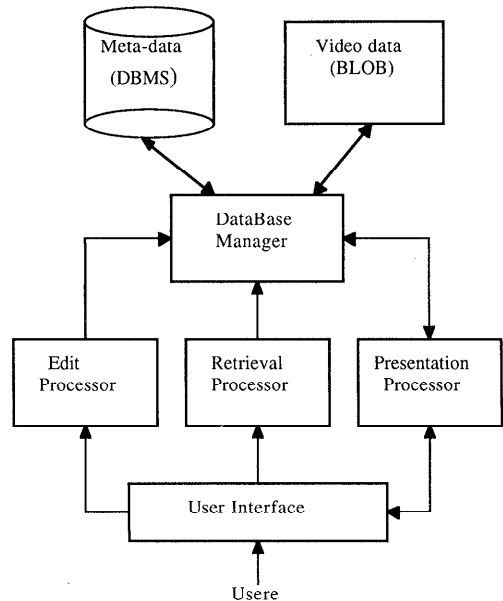


図1 システム・アーキテクチャ
Fig. 1 The architecture of our system.

- (3) Retrieval Processorはビデオデータの内容検索を扱う。まずはユーザからの検索要求をDatabaseManagerに送る。DatabaseManagerからの検索結果はPresentation Processorが受け、ユーザに提示する。ユーザは検索結果を映像として見ながら、さらに問合せを進めることができる。
- (4) Presentation Processorはビデオ表現に関して機能する。検索されたビデオクリップを映像として提示するため、検索結果のビデオクリップのOIDや開始フレーム番号および持続時間などのパラメータを用いて、該当するビデオクリップをファイルから呼び出し、指定された開始フレームから終了フレームまでをビデオ映像として再生する。
- (5) DatabaseManagerはビデオのメタデータとビデオ映像データの管理と検索処理を行う。

3. ビデオ映像の内容記述

従来の研究では、主な手法として、構造化モデリングアプローチが採用されてきた。構造化モデリングアプローチでは、ビデオクリップを基本単位のカットに分割し、個々のカットの内容をキーワードの集合の形で記述する。しかし、カット中の映像オブジェクトの位置や大きさや位置関係に関して条件を課し、該当するフレームシーケンスを検索することは困難だと考えられる。たとえば、「Aが出ているカット」というような検索は可能であるが、「AとBが重なっている場面」

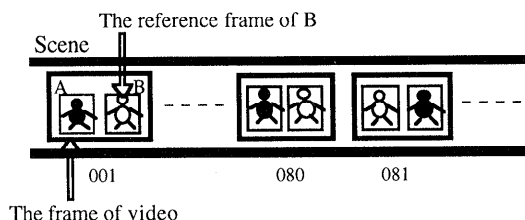


図 2 リファレンスフレームと映像フレーム

Fig. 2 The reference frame and the video frame.

というような検索は不可能である。

探したいビデオ映像やその中の場面を示す際に、ユーザがよく用いる条件は「A 対 B の試合」とか「A が B の右に立っている場面」などのような、映像オブジェクトの指定や、映像オブジェクト間の位置関係の指定である。ビデオ映像オブジェクトを分節化し、参照可能にするために、図 2 に示したように、映像中の特定の映像オブジェクトを覆う最小の矩形を考える。本論文ではこの矩形をリファレンスフレームと呼ぶ。内容に基づくビデオの検索を実現する際、検索条件はカットの属性に関して規定可能であるのみでなく、リファレンスフレームの属性に関して規定可能である。以下に、リファレンスフレームに基づくビデオ映像の内容記述法を提案する。

この内容記述法はシーンとリファレンスフレームという 2 種類の分節単位により、ビデオ映像の内容情報を構造化する。リファレンスフレームの属性は映像オブジェクトの内容属性（名称や色など）と映像オブジェクトの空間属性（ビデオ画面における映像オブジェクトを覆う最小の矩形の位置とサイズ）からなる。また、映像オブジェクトの位置とサイズの変化がほとんどない映像フレームシーケンスの区間に対して、リファレンスフレームの開始と終了映像フレームナンバを、リファレンスフレームの属性として求める。これらの属性値を用いて、リファレンスフレームを検索することにより、間接的には、それが覆う映像オブジェクトが出現している映像フレームシーケンスを検索することができる。

シーンとは、ある主題を表現する映像フレームシーケンスと定義する。各シーンはいくつかのリファレンスフレームを含むことができる。シーンの属性は、シーンが表現している事象やシーンの開始フレームナンバとシーンの長さなどである。

4. ビジュアルな検索要求

映像オブジェクト間の空間関係を検索条件として指定する場合のように、ビデオの内容に関して SQL 言

語を用いて検索要求を記述するには、SQL 言語に関する知識が必要である。一般のユーザ向けには、簡単かつビジュアルな検索要求表現を提供することを考えなければならない。本章では映像オブジェクトの名前と映像オブジェクト間の空間関係を、検索条件としてビジュアルに記述することができるビジュアル検索要求について説明する。

4.1 ビジュアル検索要求システムの考案

ビジュアルな検索要求システムに関する研究はおおむね 3 つのアプローチに分けられる。第 1 のアプローチはアイコンを用いて、検索要求を表現し、カラーの分布あるいはテキストチャの種類を規定することにより、検索する方法である^{8)~10)}。第 2 のアプローチでは、略図を用いて映像オブジェクトの概略形状を規定し、形状の面積や真円度や主軸オリエンテーションなどの特徴量によって、ビデオデータとのマッチングをとる^{10)~12)}。第 3 のアプローチは、主に記号射影というテクニックを用いて、映像オブジェクトの射影間の位置関係を 2D スtring にコード化し、その記号列マッチングにより検索をする^{10),12)}。第 1 と第 2 のアプローチでは各々カラー分布やテキストチャが同じ映像オブジェクトや、概略形状が似ている映像オブジェクトを区別することができない。第 3 のアプローチでは映像オブジェクトはすでに分節化されていると仮定し、分節化手法については何にも触れていない。

我々のシステムでは、リファレンスフレームを介して間接的にビデオ映像の内容を記述することにより、映像オブジェクトの位置とサイズを属性として検索条件の記述に利用する。さらに、これらの属性に対する関係演算を用いて、映像オブジェクト間の空間関係を検索条件として表現する。我々は、以下のような特徴を持つビジュアル検索要求表現とその処理法を提案する。

- (1) 検索要求指定において、映像オブジェクトを指定する表現には、ビットマップイメージやピクチャなどのビジュアルオブジェクトを用いる。映像オブジェクト間の相対位置関係と映像オブジェクトのサイズは、ビジュアルオブジェクトの大きさや配置をユーザが決めることにより指定される。
- (2) システムは (1) で述べた操作によりユーザによって入力されたビジュアル検索要求を分析し、映像オブジェクトの名称、位置、サイズなどの属性を取り出し、映像オブジェクト間の空間関係をあらかじめ想定されている標準的空間関係パターンの 1 つと同定する。これによって、SQL

表 1 従来の手法と我々の手法の比較
Table 1 The comparison between proposed approaches and our approach.

	検索条件の指定	検索用の特徴量	比較
第 1 のアプローチ	アイコン	カラーの分布, テクスチャーの種類	同一の特徴量を持つものを区別できない
第 2 のアプローチ	略図	形状の面積, 真円度, 主軸オリエンテーション	同一の特徴量を持つものを区別できない
第 3 のアプローチ	アイコンで指定した対象オブジェクト間の位置関係	対象オブジェクトの射影間の位置関係をコード化した 2D スtring	特徴量をオーサリングする処理がより複雑, 分節化に関して何にも触れていない
提案された手法	ピクチャで指定した対象オブジェクト間の位置関係	2D 空間において, 対象オブジェクトの位置と大きさ	特徴量をオーサリングする処理が簡単, 分節化の機能を提供

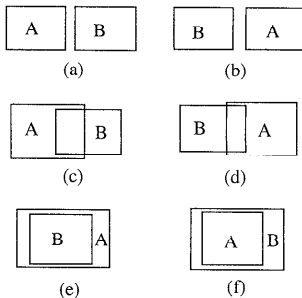


図 3 左右関係のパターン

Fig. 3 The six patterns of left-right relationships.

検索要求を生成する。従来の 3 つの手法と提案された手法の比較を表 1 に示す。

4.2 空間関係の表現

開発したシステムは、検索要求を指定するためのビジュアルオブジェクトを用意している。これらの映像オブジェクトをビジュアルに表現するオブジェクトに対する操作により、映像オブジェクトとその間の相対位置関係を表現する。2 つの映像オブジェクトの左右関係を表現する際、図 3 に示したように、いくつかのパターンが考えられる。これらを指定するのに、図 3(a) のパターンをもとに、映像オブジェクト A, B に対し、リサイズ、移動、改名などの操作をユーザに行わせ、図 3 の (b), (c), (d), (e), (f) に示したようなパターンを指定させる。このシステムの例ではこれらの空間関係中 3 種類の X 方向の A と B の相対位置関係のみを取り扱っている。ユーザが指定できる空間関係は、ビジュアル検索要求を SQL 検索条件に変換するアルゴリズムにおいてあらかじめ想定している標準パターンの範囲に限定されている。これを拡張することは可能である。

4.3 SQL 検索要求の自動生成

開発したシステムでは、SQL 検索要求言語を用いてビデオのメタデータの内容に関して検索条件を記述

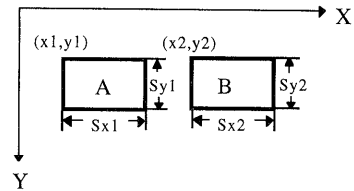


図 4 2次元座標システム

Fig. 4 The coordination system of 2D.

し、これを満たすビデオ映像を検索する。ビジュアル検索要求を SQL 検索条件に変換する作業は自動的に行われる。以下にこの変換アルゴリズムについて述べる。説明を簡略化するために映像オブジェクトの数を 2 つと仮定する。

手順 1 ユーザのビジュアル映像オブジェクトに対する操作により、指定された映像オブジェクトの名前や 2 次元空間上の位置とサイズなどのパラメータを得る。
手順 2 図 4 に示した 2 つの映像オブジェクト A と B の位置とサイズの x, y 値より、参照映像オブジェクトを以下のように選ぶ。

if $x1 \leq x2$ then

A を参照映像オブジェクトとする

else

B を参照映像オブジェクトとする

end if

手順 3 手順 2 で選ばれた参照映像オブジェクトが A であったとする (B の場合も同様である)。X 方向の A と B の相対位置関係は、以下のように 3 種類の場合に分類される。

(1) $(x1 + Sx1) < x2$

(2) $x1 < x2 \leq (x1 + Sx1) < (x2 + Sx2)$

(3) $x1 \leq x2 < (x2 + Sx2) \leq (x1 + Sx1)$

これらの場合とのマッチングをとることにより、X 軸に関して A と B の相対的な位置関係を確定することができる。

手順4 Y軸に関するAとBの相対位置関係は、以下のように6種類の場合に分類される。

- (1) $(y1 + Sy1) < y2$
- (2) $y1 < y2 \leq (y1 + Sy1) < (y2 + Sy2)$
- (3) $y1 \leq y2 < (y2 + Sy2) \leq (y1 + Sy1)$
- (4) $y2 \leq y1 < (y1 + Sy1) \leq (y2 + Sy2)$
- (5) $y2 < y1 \leq (y2 + Sy2) < (y1 + Sy1)$
- (6) $(y2 + Sy2) < y1$

これらの場合とのマッチングをとることにより、Y軸の上にAとBの相対位置関係を確定することができる。

手順5 XとY軸に関するAとBの相対位置関係に対してマッチングをとることにより確定した相対位置関係の表現式と、映像オブジェクトの名前を用いて、SQL検索要求の条件式を生成し、これをもとに、SQL検索要求を生成する。

5. ビデオデータベースの実現

5.1 システムの実現環境

ビデオのメタデータの管理と検索は、RDBMSを用いて容易に実現できる。

システムのインタフェースの実現と、ビデオ映像の編集や再生などの処理には、著者らの研究室で研究開発されたIntelligentPadというプラットフォームを用いる。IntelligentPadでは、画像、映像、音声などからなるマルチメディアドキュメントがパッドとして表現される。これらのパッドを複製することもできる。紙に紙を貼るかのようにパッドにパッドを貼ることにより、各パッドが持つ機能を組み合わせて1つに合成することができる。

5.2 ビデオデータベースの構造

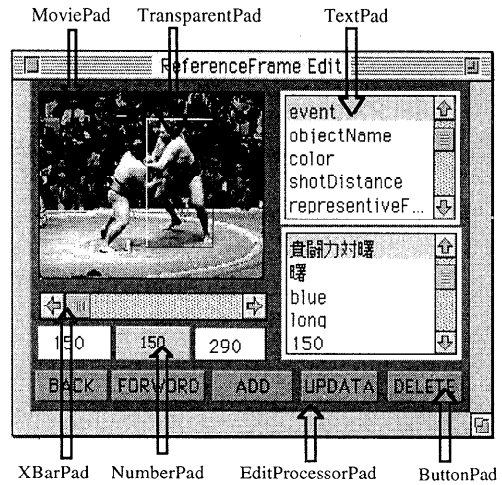
ビデオ映像にはQuickTimeムービー¹³⁾を用いる。QuickTimeを用いたことによって、映像フレーム単位での管理と操作が実現できる。これらのビデオ映像に関するメタデータは、以下に示した2つのテーブルを用いてRDBMSにより蓄積・管理する。

Scene (id, event, startFrame, duration)
ReferenceFrame (id, objectName, color, locationX, locationY, sizeX, sizeY, sceneId, startFrame, endFrame)

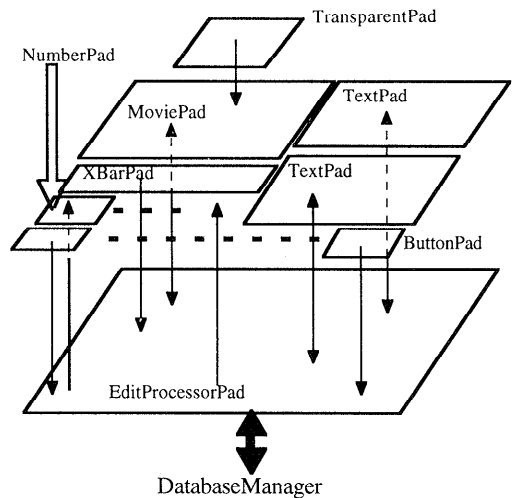
リファレンスフレームのテーブルはビデオ映像の表現や内容などを記述する属性を持った関係として定義する。シーンのテーブルはいくつかのリファレンスフレームを含むシーンの属性を持っている。

5.3 ビデオ映像の属性の抽出

ソースビデオ映像をビデオデータベースで管理する



(a) ユーザ支援ツール



(b) (a) のパッドの貼合せ構造

図5 ユーザ支援ツール

Fig.5 The support tool for user.

ためには、ビデオの内容属性などの値をユーザが対話的にシステムに入力するための支援ツールが必要である。図5(a)に示したのはリファレンスフレームの位置と大きさをビデオ上で指定するために、IntelligentPadシステムを用いて、合成パッドとして実現したユーザ支援ツールである。図5(b)に図5(a)のパッドの機能合成の構造とデータの流れを示す。一番下のEditProcessPadはシステムのEdit Processor機能を実現するために開発した専用パッドである。EditProcessPadの左の上に1枚のMoviePadを貼り付けている。このパッドは、Presentation Processorの機能を備えており、編集したいビデオ映像の選択やビデオ映像の再生などの機能を持っている。ムービーパッドの

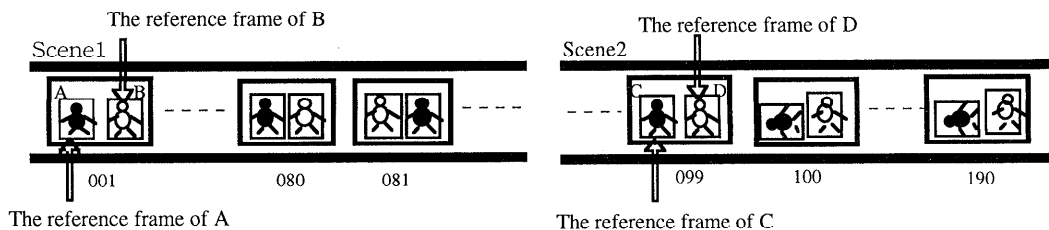


図 6 ビデオ映像オブジェクトとそのリファレンスフレーム

Fig. 6 The video projects and reference frames.

下のスクロールバー XBarPad を操作することにより、ビデオ再生速度をコントロールし、目的とする映像フレームを選ぶことができる。スクロールバーの下には、NumberPad で、選択したビデオ映像の開始フレームナンバと、現在のフレームナンバ、および終了フレームナンバが示されている。

EditProcessPad の右上のパッドは、リファレンスフレームの属性リストの表示と新属性の追加を指示・入力するための TextPad である。映像オブジェクトの位置とサイズを属性値として抽出するためには、ムービパッドの上に映っている映像オブジェクトの位置とサイズに合わせて透明なパッドを貼ればよい。このパッドをマニュアル操作で適切な大きさにして適切な位置に貼り付けると、パッド操作により、発生するパッドイベントによって、映像オブジェクトの位置とサイズが自動的にデータベースに入力される。

図 5 (a) に示したビデオ映像を例としてビデオ映像上でリファレンスフレームを設定する手順を以下に説明する。

手順 1 選択したビデオ映像を再生しながら、参照したい映像オブジェクトが映っていて、この映像オブジェクトの位置とサイズの変化がほとんどない映像フレームシーケンスの区間中の 1 つ映像フレームを選ぶ。例では、曙が貴闘力の右にいるという場面中の貴闘力と曙を別々のリファレンスフレームとする。

手順 2 貴闘力をリファレンスフレームとする場合、まず、ムービパッドの上に貴闘力の位置とサイズに合わせて、透明なパッドを貼り付ける。次に、TextPad により、貴闘力という名前を入力し、手順 1 で選んだ映像フレームシーケンスの開始フレームと終了フレームのナンバを求めて記述する。

手順 3 システムは、手順 2 で入力したデータと、貼り付けた透明パッドから得られる貴闘力の位置とサイズと、この場面を含むビデオ映像シーンの ID (物理的なムービファイル ID) を組にして、ReferenceFrame テーブルに 1 つのレコードを作成する。

手順 4 曙をリファレンスフレームとする場合、手順

2, 3 と同じ手続きで、同じテーブルに 1 つのレコードを作成する。

5.4 ビデオデータの検索

提案したビデオ映像の内容記述法によって、リファレンスフレームの属性を RDBMS で管理し、SQL 言語を用いて、ビデオ映像の内容検索を行う。

図 6 にビデオ映像オブジェクトとそのリファレンスフレームの例を示す。ビデオ映像 Scene1 中の映像オブジェクト A, B と、Scene2 中の映像オブジェクト C, D をリファレンスフレームにあらかじめ指定しておく。その結果を表 2 の ReferenceFrame と Scene テーブルに示す。たとえば、映像オブジェクト A に対し、ビデオ映像上でリファレンスフレームを設定・定義すると、5.3 節に説明した手順でリファレンスフレームの属性値からなる 1 つのレコードが生成され、ReferenceFrame テーブルに格納される。このような属性を持つテーブルからなる関係 DB に対し、SQL 検索言語を用いてビデオの内容検索を行う例を以下に示す。ここで、ReferenceFrame テーブル中の映像オブジェクト C を若乃花とし、映像オブジェクト D を武双山とする。

```
SELECT scene.id, ref1.startFrame,
       ref1.endFrame
FROM   scene in Scene; ref1, ref2 in
       ReferenceFrame
WHERE  scene.id= ref1.sceneId= ref2.sceneId
       and ref1.startFrame= ref2.startFrame
       and ref1.endFrame= ref2.endFrame
       and ref1.objectName= "若乃花"
       and ref2.objectName= "武双山"
       and ref1.locationX+ref1.sizeX
       < ref2.locationX
       and ref1.sizeX > ref1.sizeY
```

システムはこの SQL 検索要求を用いて、DB を検索する。Scene テーブルと ReferenceFrame テーブルを用いることにより、検索条件を満たすフレームシーケンスを含むシーンの ID、開始および終了フレーム

表 2 ReferenceFrame と Scene 関する関係

Table 2 The two relations on referenceframes and video scenes.

ReferenceFrame								Scene					
id	object Name	locationX	locationY	sizeX	sizeY	sceneId	start Frame	end Frame	id	event	start Frame	duration	
001	A	xa1	ya1	Sxa1	Sya1	scene1	001	080	scene1	e1	001	200	
002	A	xa2	ya2	Sxa2	Sya2	scene1	081	200	scene1	e2	001	190	
003	B	xb1	yb1	Sxb1	Syb1	scene1	001	080					
004	B	xb2	yb2	Sxb2	Syb2	scene1	081	200					
005	C	xc1	yc1	Sxc1	Syc1	scene2	001	099					
006	C	xc2	yc2	Sxc2	Syc2	scene2	100	190					
007	D	xd1	yd1	Sxd1	Syd1	scene2	001	099					
008	D	xd2	yd2	Sxd2	Syd2	scene2	100	190					

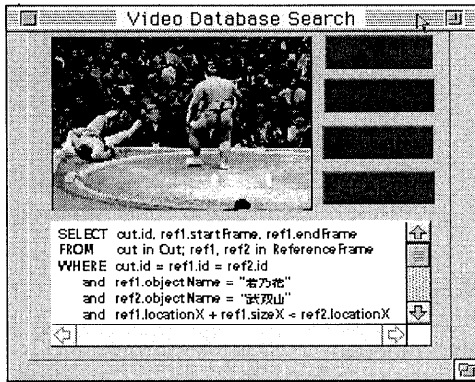


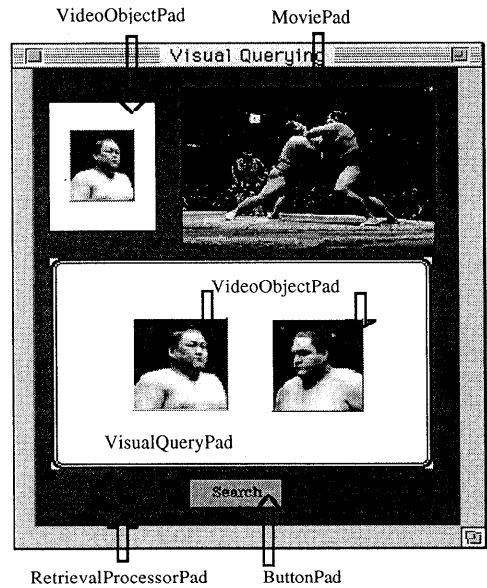
図 7 検索結果例

Fig. 7 An example retrieval.

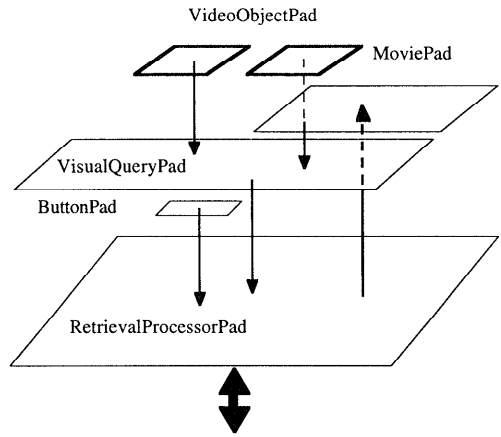
ナンバが検索結果として返される. これを用いて, 検索条件を満たす映像フレームシーケンスのビデオ映像を再生・提示することができる. 図 7 にこの検索結果を示す. 「若乃花と武双山の対戦場面で, 若乃花が武双山の左側に位置し, かつ武双山が若乃花を敗り, 若乃花が倒れている場面のビデオクリップを検索したい」というような検索要求に対し, SQL 検索言語を用いて DB を検索することが可能となる. 人物のような内容オブジェクトの高さと横幅の比較によって, 若乃花が倒れているという検索条件を指定できるが, このような物理的な値で検索条件を指定することは映像オブジェクトの物理的な属性と関連しているから, 想定した分野によって, その有効性が異なると思われる.

5.5 ビジュアルな検索要求のインタフェース

ビジュアルな検索要求のインタフェースを実現するためにも, IntelligentPad システムを用いることができる. 図 8 はビジュアル検索要求のインタフェースとそのパッドの貼合せ構造とデータの流れを示す. 図 8 (a) の左上の部分は映像オブジェクトを表現する VideoObjectPad を示している. ユーザはビジュアルな検索要求を生成する際, この VideoObjectPad を複



(a) ビジュアル検索要求インタフェース



(b) (a) のパッドの貼合せ構造

図 8 ビジュアル検索要求インタフェース

Fig. 8 The interface of the visual query specification.

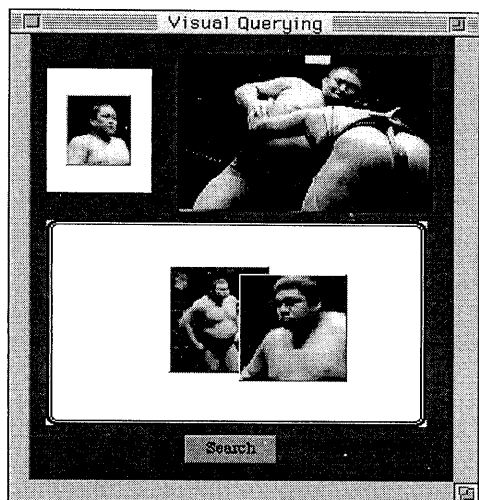


図 9 ビジュアル検索要求の検索結果

Fig. 9 A retrieval result for the visual query.

表 3 実験データ
Table 3 The data for test.

	シーン	リファレンスフレーム
数	80	166
平均長さ(秒)	16	7
レート(fps)	15	15
映像フレーム数	19200	17430

製し、図 8(a) に示した VisualQueryPad 上に貼り合わせる。図 8(a) のパッドには 2 つの映像オブジェクトが置かれており、「これらが左右に別れて存在する」という空間関係を示したビジュアル検索要求を示す。このビジュアル検索要求の意味は「曙対貴闘力の対戦で、曙が貴闘力の右側にいる場面が映っている」ビデオ映像クリップを探せという意味になる。映像オブジェクトを選ぶには、複製した VideoObjectPad をマウスクリックし、出てきたプルダウンメニューから、名前を選択すればよい。映像オブジェクト間の空間関係は、VideoObjectPad の移動やリサイズ操作により、容易に指定できる。図 8(a) の右上の部分は検索結果のビデオ映像クリップを再生する MoviePad である。

図 9 のビジュアル検索要求では、「貴乃花が武双山の左側において、貴乃花と武双山の身体が重なっている場面」を要求している。検索結果は右上の MoviePad に示すとおりである。

6. 評価

システムの実装には、相撲の 4 日間の試合の映像を用いて、デジタル化を行った。2 人の力士が立ち上がった後、勝負がつくまでのシーンのみを対象として

編集した。得られたシーン数、リファレンスフレーム数、平均長、映像フレーム数などを表 3 に示す。全体の映像フレーム数に対し、何らかのリファレンスフレームを定義できた映像フレームの割合は 91% である。

得られたすべてのシーンにリファレンスフレームを設定・定義するには、約 6 時間を要した。オーサリングの手間を減らすために、リファレンスフレームの開始フレームと終了フレームのナンバを指定することにより、この間の 1 つの映像フレームに対してのみリファレンスの設定を行えばよいように、システムを作成した。この方法は、映像オブジェクトの動きがあまり激しくないビデオデータに対しては、有効な方法と考えられる。動きが激しいオブジェクトに対しては、個々のフレームごとないしは短いフレームシーケンスごとに、リファレンスフレームを定義することが必要となる。本システムでは、このような場合への応用は想定していない。もちろん、このような場合にも労苦を惜しまなければ、本手法は適用できる。動作の激しいオブジェクトであっても、それに注目が向けられている場合には、映像中での動きは緩やかになるカメラワークが選ばれるので、このように想定しても妥当であると考えている。

相撲の 1 日分の取組みは 20 組である。1 つの取組みが表 3 では 1 シーンとして扱われている。各シーン中で 2 つの映像オブジェクトを対象とする場合、平均 1 日分の映像に定義できるリファレンスフレームの総数は 83 になる。1 年間のリファレンスフレームの総数は 7470 しかない。これは ReferenceFrame テーブルのレコード数でもある。10 年のデータを集めたとしても、妥当なサイズのデータベースに納まると考えられる。QuickTime ムービー形式のビデオ映像データファイルのサイズは、映像フレームサイズを 160 × 120 としたとき、長さ 1 分間のムービーでは約 22 MB になる。10 年間の相撲の試合のシーンは約 100 GB になるが、マルチメディアデータベースではこれも妥当なサイズの範囲である。

7. おわりに

本論文では、ビデオデータと、ビデオデータに付加したメタデータとともにファイルと RDBMS に格納することにより、メタデータをもとにビデオデータに関して内容検索できることを示し、ビデオデータベースのアーキテクチャを提案した。

ビデオデータの内容情報を構造化するため、リファレンスフレームという概念を導入し、これに基づくビデオ映像の内容記述法を提案した。この方法は、リ

ファレンスフレームを検索の基本単位として、ビデオデータの内容情報を構造化することによって、映像オブジェクトの名称や映像オブジェクト間の空間関係を検索条件としてビデオ映像の内容検索を行うことが可能になった。

提案したアーキテクチャと内容記述法をもとに、関係DBMSとIntelligentPadシステムを用いて、ビデオデータベースが実現できることを示した。さらに、映像オブジェクト間の空間関係をビジュアルに指定するビジュアル検索要求の実現についても述べた。

参 考 文 献

- 1) 有澤 博, 由井 仁, 富井尚志: 映像データベースシステムの構成—方式, *Proc. Advanced Database System Symposium '93*, pp.181-190 (1993).
- 2) Rubin, B. and Davenport, G.: Structure Content Modeling for Cinematic Information, *SIGCHI Bull*, Vol.21, No.2, pp.78-79 (1989).
- 3) Tonomura, Y.: Video Handling Based on Structured Information for Hypermedia Systems, *Proc. Int'l Conf. on Multimedia Information Systems*, Singapore, pp.333-344 (1991).
- 4) Auguierre-Smith, T.G. and Pincever, N.C.: Parsing Movies in Context, *Proc. USENIX, USENIX Assoc.*, Berkeley, CA, pp.157-168 (1991).
- 5) Tanaka, Y.: A Synthetic Dynamic-media System, *Proc. Int'l Conf. on Multimedia Information Systems*, Singapore, pp.299-310 (1991).
- 6) Nagasaka, A. and Tanaka, Y.: Automatic Video Indexing and Full-video Search for Object Appearances, *Visual Database Systems*, Vol.2, pp.113-127 Elsevier Science Publishers, Amsterdam (1992).
- 7) 井上誠喜: 画像合成のための対象物抽出法, 電子情報通信学会論文誌 (D-II), Vol.J74-D-II, No.10, pp.1411-1418 (1991).
- 8) Swain, M.J. and Gallard, G.H.: Color Indexing, *Int'l J. Computer Vision*, Vol.7, No.1, pp.11-31 (1991).
- 9) Binaghi, E., Gagliardi, I. and Schettini, R.: Indexing and Fuzzy Logic-based Retrieval of Color Images, *IFIP Trans. A-7, Visual Database Systems II*, Kunth, E. and Wegner, L. (Eds.), pp.79-92 (1992).
- 10) Flickner, M., Sawhnay, H., et al.: Query by Image and Video Content: The QBIC System,

IEEE Computer, pp.23-32 (1995).

- 11) Hirata, K. and Kato, T.: Query by Visual Example: Content-based Image Retrieval, *Advances in Database Technology-EDBT'92, Lecture Notes on Computer Science*, Vol.580, pp.56-71, Springer-Verlag (1992).
- 12) Del Bimbo, A., Pala, P. and Santini, S.: Visual Image Retrieval by Elastic Deformation of Object Shapes, *Proc. IEEE VL '94*, pp.216-233 (1994).
- 13) Apple Computer: QuickTime Developer's Guide (1993).

(平成 9 年 6 月 4 日受付)

(平成 10 年 1 月 16 日採録)



藜 亜和 (学生会員)

1982年中国中南工業大学計算機科学科卒業。同年中国非鉄金属鉱産地質研究院に勤務。主任技師。1993～1994年北海道大学客員研究員。現在、同大学院工学研究科博士後期課程在学中。データベース、マルチメディアデータベース等に興味を持つ。中国計算機学会会員。



田中 譲 (正会員)

1972年京都大学電気工学科卒業、1974年同大学院電子工学専攻修士課程修了。工学博士。1974年北海道大学工学部助手。1977年同講師、1985年同助教授を経て、1990年同教授、現在に至る。1996年北海道大学知識メディアラボラトリー長。この間、1985年10月より1年間、IBM社T. J. ワトソン研究所客員研究員。データベース理論、データベース・マシン、並列処理アーキテクチャ、メディア・アーキテクチャ等の研究に従事。「コンピュータ・アーキテクチャ」(雨宮氏との共著、オーム社)等の著書あり。1994年にIntelligentPadの開発に関して日経BP技術賞大賞受賞。1995年より3年間文部省科研費特別推進研究遂行。ソフトウェア学会、人工知能学会、米国IEEE各会員。