

TV 受信端末におけるダイジェスト視聴システム

橋本 隆子^{†‡} 白田 由香利^{†‡} 真野 博子[‡] 飯沢 篤志^{†‡}

放送のデジタル化に伴い、映像情報の補足情報をインデックスとして付加し、そのインデックスを用いて映像情報を検索する方式が各種研究されている。インデックスを利用した映像情報検索方式の一つとして、ダイジェスト視聴に注目し、視聴者ごとに意味的に重要なシーンを判定してパーソナルなダイジェストを可能とするシステムを試作した。またこのダイジェスト視聴システムでは、視聴者が複合事象に対して自分の嗜好を交えた抽象度の高い用語を使って検索できるように抽象インデックスを定義できるようにした。本稿では、まずダイジェスト作成モデルを説明し、それを実現するための映像スキーマ定義言語を説明する。そして、野球番組を対象にして、視聴者の嗜好情報登録及び各種検索が可能なダイジェスト視聴システム及びその検索インタフェースの試作を行ったので、その機能についても述べる。

Prototype of Digest Viewing System for Television

Takako Hashimoto, ^{†‡} Yukari Shirota, ^{†‡} Hiroko Mano[‡] and Atsushi Iizawa^{†‡}

Beyond program contents, digital data broadcasting can deliver additional data as indexes attached to the contents. Using these indexes, users can browse and retrieve parts of the program. We have developed a system that can construct a digest of the program on the fly using indexes. The system can extract scenes and restructure them based on the information obtained from indexes and user preferences. Also, the system has a mechanism to translate the general terms users would use in describing the scenes they are looking for to those terms actually used in the program indexes. The paper firstly describes a formal model of the digest production and a program scheme definition language to implement the model. Also, we explain our prototype system of the digest viewing system with a user interface to put in user preferences and various retrieval requirements for baseball programs.

1. はじめに

近年、放送のデジタル化が世界的規模で急速に進展している。日本においても、2000年末頃にはBS(Broadcast Satellite)デジタル放送や地上波デジタル放送の開始が予定されている。我々は、現在の放送の発展型として、高度に構造化されたデジタル情報をサービスとして提供する「情報放送」の研究を行なっている¹⁾。情報放送では、これまでのアナログ放送の「配信された番組をそのまま見る」という視聴だけでなく、番組に付随して配信される各種の番組付加情報を利用して、「受信端末上でノンリニアに視聴する」という視聴形態もまた可能となる。

(株)次世代情報放送システム研究所では、番組付加

情報の1つとして日本放送協会(NHK)と協力し、番組インデックスの開発をした。番組インデックス方式は電波産業会(ARIB)に提案され、99年5月に標準化されている。番組インデックスとは、本格的なBSデジタル放送に向けて、EPG(Electronic Program Guide)をさらに発展させ、番組表だけでなく番組間の関係づけや、番組の構造など、様々な表現を可能とするメタデータ記述である²⁾³⁾。我々は番組インデックスを用いたサービス形態の一つとして本ダイジェスト視聴システムを研究し、試作実験により検証している⁴⁾⁵⁾。我々が想定するダイジェスト視聴は、一般家庭の受信端末において、放送局から送信された映像付加情報を利用して、視聴者自らが動的にダイジェスト映像を作成するシステムである。一般に映像の事象を厳密に表現した情報

[†] (株)次世代情報放送システム研究所 Information Broadcasting Laboratories, Inc.

(株)リコーより(株)次世代情報放送システム研究所に兼任出向中。

The authors are partly on loan from Software Research Center, Ricoh Company, Ltd. to Information Broadcasting Laboratories, Inc.

[‡] (株)リコー 画像システム事業本部 ソフトウェア研究所

Software Research Center, Imaging System Business Group, Ricoh Company, Ltd.

を付加しようとする、付加コストがかかり、かつ利用者の個人的視点に偏り汎用的でなくなるため、放送事業者が映像付加情報を付加、送信する場合には、低コストでかつ再利用可能な情報が作成されと考えられる。そこで、我々は「放送で配信するインデックスはアプリケーションビューに依存しない汎用インデックスのみ」と仮定した。我々の利用した番組インデックスは、GEL[®]というシステムで入力したものであるが、コンテキスト定義は複雑ではなく、これによるインデックス付加作業は容易であるという利点をもつ。GELで付加するインデックスは基本的かつ客観的なインデックスのみである。

ダイジェスト視聴システムには、以下のような要求事項が考えられる。

● パーソナル化

受信端末上で動的にダイジェストを作成する場合、個々の視聴者の嗜好を反映したパーソナルなダイジェスト映像が望まれる。特にスポーツ映像のように、人気が高く、視聴者の嗜好の差が激しい番組に対してダイジェストを作成する場合は、各視聴者の好きなチーム、選手に注目した、それぞれの視聴者にとって意味的に重要な映像が要求される。

● 抽象的な用語による検索 (抽象検索)

パーソナルなダイジェストを作成するためには、個々の視聴者ごとに意味的に重要なシーンを検索できなければならない。そのためには、基本的な番組インデックスを用いて、映像内容の意味付けを行う仕組みが必要となる。我々は基本的な番組インデックスから、視聴者になじみのある親しみやすい抽象的な用語による意味的なインデックスを生成し、検索する方式の検討を行った。以後、この抽象用語による検索を「抽象検索」と呼ぶ。

我々は、上記の要求事項に対応するために、視聴者の嗜好を反映することができるダイジェスト映像作成の汎用モデルを作成した⁷⁾。さらにこのモデル上で重要場面判定に必要なパラメータやインデックスを定義するために、映像スキーマ定義言語の開発を行い⁸⁾、野球番組を対象としたダイジェスト映像作成システム及び一般視聴者向け検索インタフェース(IF)の試作を行った。

本稿は次のような構成になっている。第2章では、我々が提案するダイジェスト作成モデルについて述べ、第3章では、そのモデル実現のための映像スキーマ定義言語について説明する。第4章では、野球映像を対象として試作したダイジェスト視聴システムについて紹介し、第5章でその評価を行う。第6章では、我々の

方式と関連研究との比較を行う。

2. ダイジェスト作成モデル

本節では、我々が提案するダイジェスト作成モデルを説明する。図1にモデル概要を示す。本モデルは、エンジン部分となるダイジェスト作成部と、視聴者向けインタフェースとなる検索IF、受信端末におけるデータベース(DB)から構成される。本ダイジェスト作成モデルは表1のような要素で構成される。表には誰が定義するのか、またその情報がパーソナルなものか汎用であるかの区別も示した。表中"パ"はパーソナルであることを示す。

表1 ダイジェスト作成モデルのタプル項目
Tab.1 Tuples of Digest Making Model

記号	名称	定義(作成者)	パーソナル
V	映像シーンの集合	放送局サイド	汎用
I	インデックスの集合	放送用:放送局	放送用:汎用
R	ルール関数の集合	放送局サイド	汎用
F	嗜好情報の集合	視聴者	パ
C	検索条件の集合	視聴者	パ
P	重要度判定パラメータの集合	放送局サイド	汎用
calc	重要度判定パラメータ計算関数	放送局サイド	汎用
digest	ダイジェスト映像作成関数	放送局サイド	汎用

以下では、重要場面がどのように選択されるかを、上記モデルと図1のモデル概要図を用いて順を追って説明する。

2.1 インデックス

まず放送局あるいは番組制作サイドは予め番組の種類ごとに基本的(非複合事象的)かつ客観的事象に対してインデックス体系を策定する。このインデックスを基本インデックスと呼ぶ。例えば、野球番組であればホームラン、ヒット、サッカー番組であればシュート、ゴールなどの事象が基本インデックスとなる。我々は上述したダイジェストのパーソナル化を実現するために、基本インデックスを用いて映像内容の意味を示す抽象インデックスを定義できるようにした。抽象インデックスは、既に振られている基本インデックスの状態遷移パターンで記述される。指定された状態遷移パターンがマッチした場合に、抽象インデックスが映像上に付加される。抽象インデックスには、(A)複合事象を示すインデックスと、(B)主観的なインデックスがある。図2に基本インデックスと抽象インデックスの区別を示した。例えば、野球番組における「逆転」や「同点」というインデックスは複合事象に対して定

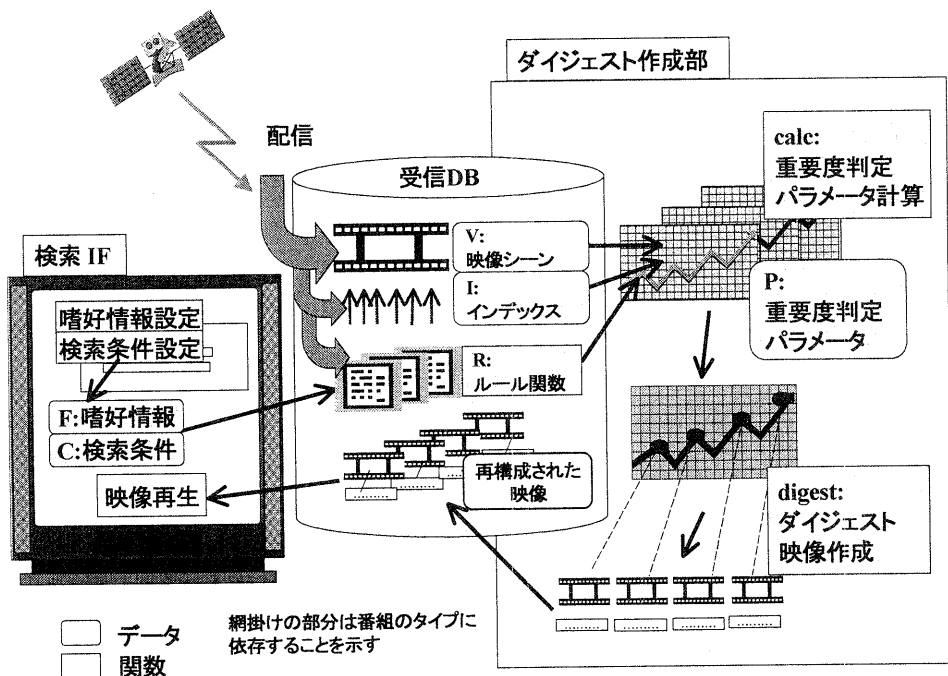


図1 ダイジェスト作成モデル
Fig.1 Digest Making Model

義される客観的インデックスである。一方、「新人投手が4番打者相手に頑張っているシーン」というインデックスは、「頑張っている」という内容をどのように定義するかは個人の主観によるので、主観的な複合事象である。放送局サイドで定義した抽象インデックスを実際に付加してから放送する場合、放送では抽象インデックスの定義だけ配信しておき、家庭においてダイジェスト作成システムを起動した時に実際に付加される、という2通りの運用形態がある。

2.2 ダイジェスト作成の概要

基本インデックス及び抽象インデックスから構成されるインデックスIが、映像シーンVとともに放送配信される。放送局サイドでは、予め、番組の種別毎に、重要場面を判定するための重要度判定パラメータPを定義する。重要度判定パラメータPの計算ルールRも上記のインデックスIを用いて定義しておく。ルールR自体は中立的立場で定義され汎用である。ルールRを計算する際、視聴者の嗜好情報Fを利用することによりパーソナルなダイジェストが作成される。番組放送の際は、放送局からインデックスIが付加された映像シーンV、及び重要度判定パラメータPの計算ルールRが配信され、家庭のTV型受信端末の受信DBに格納される。視聴者は家庭内のTV型受信端末上で動作する検索IFを通して、次の2種類の値を設定する。

(ア) 嗜好情報(F): 嗜好情報は、好きな選手、好きなチームなどの嗜好項目の種別ごとに設定される。嗜好項目種別ごとに、嗜好項目種別ごとに、嗜好項目 f_{ki} と、その重みのペア (f_{ki}, w_{ki}) を定義する。但し $0 \leq w_{ki} \leq 1$ 。添字 k は、嗜好項目種別の識別のため、添字 i は、同種の種別に対して複数設定できるのでその識別のためである。種別ごとの嗜好情報の集合を以下のように表わす。

$$F_k = \{(f_{ki}, w_{ki})\}$$

F_k の和集合が嗜好情報 F となる。

(イ) 検索条件(C): 各場面の長さ、最終的なダイジェストの長さなど。

2.3 重要場面判定処理

重要度判定パラメータ P の計算処理について説明する。基本インデックス $i_j \in I$ を其々一つだけ含むように映像を映像シーン $v_j \in V$ に分割する。各基本インデックスと映像シーンは1対1関係となる。基本インデックス i_j が付加された映像シーンを v_j と表記することとする。重要度判定パラメータ P の値は各基本インデックスの振られた位置に対して計算され付加される。ある基本インデックス i_j に対応する重要度判定パラメータは p_1, p_2, p_3 というように表記する。重要度判定パラメータ値を計算する関数がルール R で

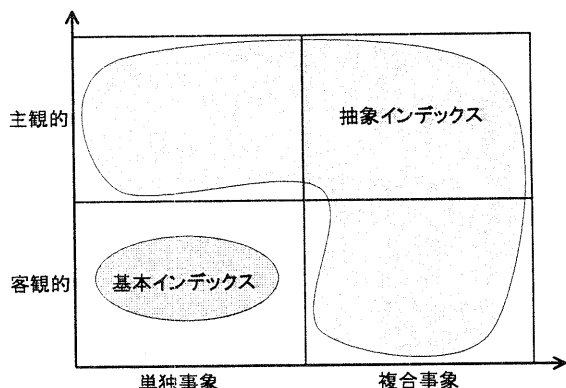


図2 基本インデックスと抽象インデックス
Fig. 2 Basic Index and Abstract Index

あり, その関数 $\forall \text{rule}_k \in R$ は次のように定義される.

$$\text{rule}_k: I \times P \rightarrow P$$

ルールは, あるインデックスが存在した場合, どのように P を計算するかを定義するものである.

ルールは, 基本インデックスだけでなく, 抽象インデックスに対しても定義できる. その場合も, 計算された値は, 抽象インデックスが付加された位置に対して計算され付加される. Rule 関数の内容は四則演算に限られない一般処理を想定している.

パーソナルなダイジェスト作成のため, 嗜好情報 F が参照される. 重要度判定パラメータ計算の際, 嗜好情報 F の項目 f_{ki} に関連したルール R があつた場合, その計算結果は, 重み w_{ki} を参照して調整される. どのように調整されるかは, 実装に依存する.

次にこれら複数の重要度判定パラメータ値 $\{p_i\}$ を使って, 各基本インデックスに対して一つの重要度値 pp を計算する. この重要度値計算関数が calc である. 関数 calc の内容は四則演算に限定されない一般手続きとする. 典型的な calc 処理内容例は, 以下のような重み付け合計である. この例ではインデックス i_j の位置に対して計算を行っている.

$$pp: \text{calc}(p_1, p_2, p_3, \dots) = \sum_i \text{weight}_i \times p_i$$

次に, この pp 値の絶対値の大きい順に基本インデックスをソートし, その基本インデックスを中心として映像シーンの切り出しを行なう. この重要場面判定及びシーンの切り出し処理を行なうのがダイジェスト映像作成関数 digest である. 関数 digest は, 選出された基本インデックスが付けられた時点を中心とし, 前後のどの部分までをその再生シーンとすべきかを計算する. 指定されたダイジェスト視聴時間を越えるまで, シーン切り出しを繰り返し行なう. 検索条件(C)はダイジェスト作成関数 digest の中で参照される. 関数 digest 自体は汎用に定義されるが, 検索条件(C)の情報を引数として渡すことにより, パーソナルなダイジェスト作成が行われる.

関数 calc 及び digest はダイジェスト視聴システムのプログラムの一部として実現される. そのプログラムも放送配信して家庭に配る, という運用形態もある.

3. 映像スキーマ定義言語

前章で述べたダイジェスト作成モデルを実現するために, 映像スキーマ定義言語を実装した. 本章では, その言語仕様と特長を説明する. 映像スキーマ定義言語は, 以下の4つの記述部分から構成される.

(1)インデックス名の変換定義

(2)階層構造定義

(3)抽象インデックス定義

(4)重要度判定パラメータ定義

それぞれの定義について以下で説明する. なお予約語は下線で示されている.

3.1 インデックス名変換定義

付加される基本インデックスの種類や名前, 属性名などはインデックス製作者ごとに異なると考えられる. このような基本インデックスの特徴から, インデックス名や属性を隠蔽するため, 本ダイジェスト作成システム内で参照するときの名称, 参照インデックス名を定義する(図3).

<参照インデックス定義> =

<INDEX> <参照インデックス名>

<REF> <基本インデックス名> </REF> </INDEX>

図3 参照インデックス定義の構文形式

Fig.3 Reference Index Definition

3.2 階層構造定義

ダイジェスト作成モデルにおいて階層構造はないが, 実装では以下の効率化のため階層構造を定義している.

- ・階層構造を用いることで, 抽象インデックスの検索処理効率向上が図れる.

- ・重要度判定パラメータの計算処理の定義が容易になる.

階層構造定義では基本インデックスを利用して, 映像を階層的に構造化し, 検索クラスとして定義できるようにしている(図4参照). 階層構造定義は<検索クラス定義>における<親検索クラス名指定>で行う. この定義されたクラスを検索クラスと呼ぶ.

<検索クラス定義>では<開始基本インデックスリスト>に指定されているインデックスから<終了基本インデックスリスト>に指定されている基本インデックスまでが一つの検索クラスインスタンスとみなされる. <終了基本インデックスリスト>が省略されている場合は, <開始基本インデックスリスト>に指定されているインデックスから, その次に同じインデックスが出現するまでが検索クラスインスタンスとなる. 例えば野球番組において, 「イニング開始」基本インデックスから次の「イニング開始」基本インデックスまでを一つのイニングのクラスインスタンスとする.

階層構造は重要度判定パラメータの計算ルール R の定義を実現するために利用される. また, 検索クラスにより検索範囲を限定することで検索効率向上も図っている. そのためにも検索クラスの長さは大き過ぎ

```

<検索クラス定義> = <CLASS> <検索クラス名>
<PART OF> <親検索クラス名指定> </PART OF>
<START> <開始基本インデックスリスト> </START>
[ <END> <終了基本インデックスリスト> </END> ]
[ <パラメータ定義> | <参照データ定義> ] </CLASS>
<参照データ定義> = <DATA> <参照データ名>
[ <REF> <設定箇所指定> [ | <設定箇所指定> ] </REF> ]
</DATA>
<設定箇所指定> = <参照箇所データ> [ <条件式> ]
<設定箇所データ> = <基本インデックス名. 属性名>
| <参照データ名> | <パラメータ名>
<インデックス名> = <参照インデックス名>
| <抽象インデックス名>
<条件式> = ( <参照データ> (=) | <|> | <|<=> | <=> )
( <参照データ> | <数値> )

```

図4 検索クラス定義の構文形式
Fig.4 Retrieval Class Definition

ないことが望ましい。例えば、野球番組では試合の構造が明確に細分化されていて、その構造単位は検索クラスとして適当である。一方サッカーの試合のように明確な構造が前半、後半の2個しかなく、検索クラスとして不適当なものもある。こうした場合には、放送局サイドの判断で予めさらに細分化した構造単位を決める必要がある。我々はサッカー番組に対しては、セットプレイの間をプレイ区間と呼び、構造単位としている⁹⁾。

参照データとは、検索クラス内の計算処理において参照されるデータであり、検索クラス内の状態変数に相当する。参照データを定義しているクラスの開始時に参照データが生成され、指定インデックスが出てく

```

<抽象インデックス定義> = <ABS INDEX> <抽象用語名>
<CLASS> <検索クラス名> </CLASS>
<PATTERN> <状態遷移パターン指定> </PATTERN>
[ <KEY> <キーインデックス指定> </KEY> ]
[ <EXP> <構文要素指定> </EXP> ]
[ <ADD> <付加インデックス位置> </ADD> </ABS INDEX> ]
<状態遷移パターン指定> =
<遷移パターン> ( [ | & ] <遷移パターン> )
<遷移パターン> = [ [ ] <パターン要素> ]
[ [ ] <パターン要素> ] [ ]
<パターン要素> = ( <インデックス名> [ + ] )
| ( <データ> ( = | > | < | <= | < ) <データ> ) | [ & ] )
<データ> = <参照データ名> | <パラメータ名> | <数値>
| ( <インデックス名> [ <数値> ] [ $ ] )

```

図5 抽象インデックス定義の構文形式
Fig.5 Abstract Index Definition

る度に、その値が再計算される。参照データは親検索クラスから子検索クラスに継承される。

3.3 抽象インデックス定義

次に抽象インデックス定義を説明する(図5参照)。

<抽象インデックス定義>は、抽象インデックスを生成するための記述である。<状態遷移パターン指定>は、ダイジェスト作成モデルにおける抽象インデックスを定義する事象列を記述する指定である。<検索クラス名>を指定することにより、状態遷移パターンマッチングを行う範囲を指定する。<キーインデックス指定>とは、<状態遷移パターン指定>で指定されているパターンをインデックス集合Iから検索するにあたって、検索の取り掛かりとなるキーインデックスを定義するものである。抽象検索処理においては、まずキーインデックスの位置を発見し、そこから前後に、指定された状態遷移パターンがないかどうか検索クラス範囲内で探していく。よってキーインデックスの指定により検索効率が向上する。

映像上での状態遷移パターン発見の効率化アプローチ手法として、グルー演算に対して順序関係フィルタを適応する研究がある¹⁰⁾。この研究では、あるキーワードが振られている区間集合間での演算の効率化を図っている。このアプローチはある区間のキーワード出現率が高い場合に有効である手法と言える。例えば、犬と人が遊んでいるシーンに対して犬が出てくる度にインデックスを振るような運用形態である。しかし、野球番組のインデックス付加においては、ホームラン、加点などの事象が起こった際、基本インデックスを振るのは1度限りである。そうしたインデックスの出現頻度が小さい場合、区間演算を効率化する手法よりも、我々のキーインデックスを中心に前後に検索する方式の方が効率的と言える。

抽象インデックスパターンが発見された時は、その抽象インデックスを指定位置に付加する。その位置は<付加インデックス位置>によって指定する。<付加インデックス位置>は、既に付加されているインデックス名を指定する。基本インデックスだけでなく、抽象インデックスを指定してもよいが、その場合、その抽象インデックスの<付加インデックス位置>を遡ると、必ず基本インデックスにたどり着くので、その基本インデックスの位置に対して、重要度値ppは計算される。<付加インデックス位置>の指定が省略された場合は、キーインデックスの場所となる。

映像上に付加された抽象インデックスは付加された後、通常の基本インデックスと同様に扱えるため、抽象インデックスを使って新たな抽象インデックスを定

義することができる。

<状態遷移パターン指定>においては、<パターン要素>を “,” または “_”, “~” で並列表記したものが指定できる。“,” による並列表記は、そのパターン要素が時系列に成立することを示す。指定されたパターン要素の間に他のインデックスが付加されていても良い。“_” による並列表記は、そのパターン要素が時系列に成立し、かつそのパターン要素間に他のインデックスが入ってはいけないということを示す。“~” による並列表記は、指定されたパターン要素の順序に制約がなく、どのような順序であってもそのパターン要素が成立すればマッチすることを示す。<パターン要素>には、<インデックス名>の他に、<参照データ名>、<パラメータ名>などを使った状況指定ができる。<インデックス名>の後に続いて記述される “+” は、そのインデックスが 1 回以上、<検索クラス>で指定された範囲内に付加されていることを示している。“^” と “\$” は、検索クラスの開始インデックスと終了インデックスを示す。

3.4 重要度判定パラメータ定義

本映像スキーマ定義言語における「重要度判定パラメータ」（ダイジェスト作成モデルにおける P）の定義構文を図 6 に示した。各重要度判定パラメータは、検索クラスに対して定義される。クラス生成時にパラメータのインスタンスも生成され、生成時にパラメータ値はゼロクリアされる。重要度判定パラメータは、直接的及び間接的に重要度値 pp の計算に使われる。

次に嗜好情報を用いてどのようなパーソナルなダイジェストを作成するのか、処理方式を説明する。嗜好情報を利用した重要度パラメータルール記述では以下のように、基本インデックス属性と嗜好項目種別の対応するペアを角括弧部分に示す。

```
Index[attr_name=$k] +=5
```

この例では、計算している重要度判定パラメータは p1 であると仮定する。映像スキーマ定義言語では嗜好項目の種別を “\$” 記号により示す。嗜好項目の種別として何があるのかは、番組ごとのダイジェスト視聴用検索 IF で定義されている。基本インデックスの属性 attr_name として、嗜好項目の種別 k の項目 f_{ki} が出現したら、[ルールで定義された加点分×重み w_{ki}] が p1 値に追加される。例えば嗜好項目の種別 “who” に対して、嗜好情報 (“清原”, 0.8) が設定されており、かつ、以下のルール及び基本インデックス i_m があるとす

rule: ホームラン[打者=\$who] +=5

i_m : ホームラン [打者="清原"]

$p1_m$ は嗜好情報が設定されていなければ、5 点加算され、設定されていれば、 $5 \times (1+0.8)=9$ という計算により 9 点加算される。

3.5 定義の評価順序

抽象インデックスの付加されるタイミング及びルールの実行順序などの評価が行われる順序について説明する。

ダイジェスト視聴は蓄積型視聴を前提としている。ダイジェスト作成開始時に TV 型受信端末の DB に格納されている情報(ダイジェスト作成モデルにおける V, I 及び R)に対して、抽象インデックス付加及びルール計算が行われる。

評価処理は、毎回、映像スキーマ定義ファイルの先頭から最後まで、前から順に一度だけ行うこととした。つまり、ファイルの中の、重要度判定パラメータ値の計算及び抽象インデックス付加処理は、その定義ファイル中での定義された順番に処理が行われる。一つの重要度判定パラメータ値計算ルール定義の中に、複数の<パラメータ要素>がある場合、書かれた順番に処理される。

抽象インデックスを使ったルール定義の場合は、必ずルール定義の前に抽象インデックスの定義を行う必要がある。抽象インデックスの定義が評価されると、その時点の状態は、全映像シーンに対してその抽象インデックスが付加された状態となる。その状態に対して、抽象インデックスを利用したルール計算処理が行われる。一度付加された抽象インデックスは付加されたまま DB に格納される。

本映像スキーマ定義言語の評価においては、後ろの定義によってパラメータの値が変わっても、それが溯って前の定義処理を再度引き起こすことはない、とした。それは「定義は、前から一度だけ評価する」としたからである。また、基本インデックスが付加された位置と同じ位置に、抽象インデックスが振られることがあるが、この場合、あくまでも抽象インデックスは

```
<パラメータ定義> = <PARAMETER> <パラメータ名>
<RULE> <パラメータ計算ルール指定> </RULE>
</PARAMETER>
<パラメータ計算ルール指定> =
  <パラメータ要素> <計算式>
  [ : <パラメータ要素> <計算式> ]
<パラメータ要素> = <インデックス名> [[<数値>] | [<$>]]
[ <条件式> ]
```

図 6 パラメータ定義の構文形式

Fig. 6 Parameter Definition

基本インデックスに振られるのではなく、基本インデックスの振られた位置に対して振られるので R の計算上の曖昧性はない。

重要度判定値 pp の計算は、各基本インデックスに対して、基本インデックス位置と同じ位置に振られた抽象インデックスに関する計算結果値も含めて、行われる。例えば、あるインデックス i_j に対して $p1$ 値を計算するルールの<パラメータ要素>が複数あった場合は、定義された順に各々の<パラメータ要素>を評価処理し、 $p1_j$ を加算していく。

pp の計算は、抽象インデックスではなく、基本インデックス集合に対して時系列に一度だけ前から順に計算をすればよい。再計算がないこと、かつ、定義ファイル中で定義された順に一度だけ評価を行うため、同じインデックス集合 I に対しては、毎回同じ計算結果となる。

問題は、インデックス間の相関がある場合、前から順に一度だけの処理で、定義したい内容を記述可能であるか、という記述能力に関することがあげられる。しかし、この問題は、冗長にパラメータ及びルールを定義することで解決できる。例えば、 $p1$ を再計算したい場合、その後方部分で $p1'$ を再度定義し、pp の計算に使えばよい。このような仕様にした理由は、(1)再計算をしないので、処理系が簡単に実現できること、(2)再計算ループの発見などをしなくてすむので、エラー処理が簡単になること、(3)プログラムに不慣れな利用者（放送局の番組制作者など）には、こうした簡単な処理ルールの方が理解し易いこと、があげられる。

実際にダイジェスト視聴が行われる場面を想定して説明する。例えば、野球番組のダイジェスト視聴にお

いて、視聴者が帰宅したのが 5 回裏の時点だとする。ここでダイジェスト作成が指示されると、システムは試合開始から 5 回裏までの情報に対して処理を行う。その後、別の家族が深夜に帰宅してダイジェスト作成を指示すると、再度試合開始から試合終了までの全情報に対して処理を行う。その際システムが差分情報だけに対して処理を行うのでは不十分である。例えば、検索クラス「試合」に対して抽象インデックスが振られていた場合、再度試合開始から見えないと、正しく抽象インデックスが発見できないことがある。よって毎回、試合開始からのデータを対象に処理を行う。

4. 野球映像用ダイジェスト視聴システム

第 3 章で述べたダイジェスト作成モデルに基づき、第 4 章の映像スキーマ定義言語を用いて、野球映像用のダイジェスト作成システムを試作した。その試作事例を説明する。

4.1 野球映像におけるルール関数：rule

まず野球番組に対してどのようにスキーマ定義を行ったか説明する。野球映像において、試合、インニング、打席、投球をクラス階層として定義した。そのインスタンス例を図 7 に示す。映像スキーマ定義言語では、<検索クラス定義>においてこの構造化が記述されている(図 8)。試合クラスにはインニングの表、裏を示すデータが定義されている。

次に P 及び R をどのように定義したか説明する。我々は野球における重要度判定パラメータ P として、以下の 3 つを定義した。

表 2 野球映像における代表的な重要度判定パラメータ
Tab.2 Important Level Analysis Parameters for Baseball Video Streams

P1	攻撃レベル
P2	投手レベル
P3	興奮レベル

この他にも重要度判定パラメータとして、「ホームチーム得点」などを定義しているが、pp の計算に直接的に関わるのはこの 3 つである。この 3 つの重要度判定パラメータを計算するルールについてそれぞれ説明する。p1, p2, p3 の各パラメータ値は、そのパラメータが定義された検索クラスの開始時にゼロクリアされる。図 8 の定義では検索クラスが打席なので、打席クラスの開始にゼロクリアされ、その後増加し、その打席クラスの終了時に再度 0 にリセットされる。現在、野球番組に対する重要度判定パラメータの計算ルールは、四則演算に限定している。

攻撃レベルは攻撃的に重要なレベルを示すパラメータである。ヒットやホームランなどのイベントで加算される。

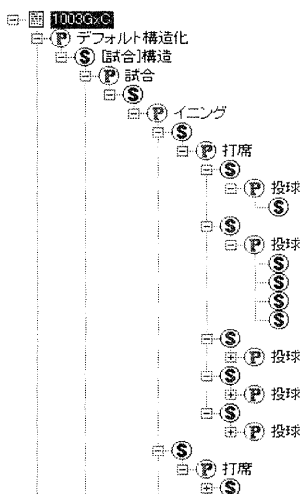


図 7 野球映像の階層定義によるインスタンス例

Fig.7 Instance Example by Class Definition for Baseball

投手レベルは投手及び守備の調子を示す。アウトやストライク、三振などを取ると加算されるようにした。興奮レベルは視聴者の期待及び興奮度を示す。打順が3、4、5番のクリーンナップの打席であったり、ランナーが3塁に出ていて得点のチャンスであるといっ

たような時に値が加算される。

p1 から p3 までのルール集合は、視聴者の嗜好項目と重みのペア (f_{ik}, w_{ik}) を反映して加算される。嗜好項目と重みは後述する検索 IF から入力される。野球用検索 IF では、嗜好項目の種別としてチームと選手の 2

種類を入力できるようにした。映像スキーマ言語中で定義された \$steam, \$who に関するルール、及び野球用検索 IF で入力された嗜好項目と重みをのペアを用いて、対応するルールがあった場合は、その重要度判定パラメータ値に加算する。

4.2 重要度判定パラメータ計算関数: calc

野球用の calc 関数は、上記 3 つの重要度判定パラメータの重み付け加算処理とした。calc ではまず、3 つの重要度パラメータに対して、どのような重みをかけるかについて決定を行なう。重み付けの判定基準及びその計算は、次の通りである。

① チームの平均得点率 s を次のように計算する。

$$s = \text{チーム得点} / \text{経過した回数}$$

例えば、試合が 9 回裏で終了した時点におけるスコアが 3 対 7 であった場合、 s 値は次のようになる。

$$\text{先攻チーム: } s = 3 / 9 = 0.33$$

$$\text{後攻チーム: } s = 7 / 9 = 0.77$$

② 上記 s から重み値を決定する。

平均得点率におけるしきい値 sb に対して $s > sb$ の時には、得点の入った打撃戦であると解釈し、該当チーム攻撃時の各パラメータの重み $(p1w, p2w, p3w)$ を、 $(1, 0, 1)$ として計算する。一

```
#検索クラス定義 (階層構造定義)
<CLASS> 試合 <START> game_start </START> <END> game_end </END>
<DATA> 表裏 <REF> inning_start.half </REF> </DATA>
<PARAMETER> ホームチーム得点 <RULE> 加点[表裏="裏"]+1 </RULE> </PARAMETER>
<PARAMETER> アウェイチーム得点 <RULE> 加点[表裏="表"]+1 </RULE> </PARAMETER>
</CLASS>
<CLASS> インニング <PART OF> 試合 </PART OF> <START> inning_start </START>
<DATA> インニング数 <REF> inning_start.no </REF> </DATA>
<SWITCH> 表裏
  <CASE> "表"
    <DATA> 攻撃チーム得点 <REF> アウェイチーム得点 </REF> </DATA>
    <DATA> 守備チーム得点 <REF> ホームチーム得点 </REF> </DATA>
  <CASE> "裏"
    <DATA> 攻撃チーム得点 <REF> ホームチーム得点 </REF> </DATA>
    <DATA> 守備チーム得点 <REF> アウェイチーム得点 </REF> </DATA>
</SWITCH>
</CLASS>
<CLASS> 打席 <PART OF> インニング </PART OF> <START> start_batting </START>
<DATA> 打者名 <REF> start_batting.batter_name </REF> </DATA>
</CLASS>
<CLASS> 投球 <PART OF> 打席 </PART OF> <START> pitching </START> </CLASS>
#抽象インデックス定義
<ABSINDEX> 先制 <CLASS> 試合 </CLASS>
  <PATTERN> 加点 </PATTERN> <KEY> 加点 </KEY> <ADD> 加点 </ADD>
</ABSINDEX>
<ABSINDEX> 同点 <CLASS> インニング </CLASS>
  <PATTERN> (攻撃チーム得点 < 守備チーム得点), 加点+, (攻撃チーム得点 = 守備チーム得点)
  </PATTERN> <KEY> 加点 </KEY> <ADD> 加点[$] </ADD>
</ABSINDEX>
<ABSINDEX> 逆転 <CLASS> インニング </CLASS>
  <PATTERN> (攻撃チーム得点 < 守備チーム得点), 加点+, (攻撃チーム得点 > 守備チーム得点)
  </PATTERN> <KEY> 加点[$] </KEY> <ADD> 加点[$] </ADD>
</ABSINDEX>
<ABSINDEX> 反撃開始 <CLASS> インニング </CLASS>
  <PATTERN> (攻撃チーム得点 < 守備チーム得点), 進塁, 加点 </PATTERN>
  <KEY> 進塁 </KEY> <ADD> 進塁 </ADD>
</ABSINDEX>
<ABSINDEX> 反撃及ばず <CLASS> インニング </CLASS>
  <PATTERN> 反撃開始, (攻撃チーム得点 < 守備チーム得点)_$ </PATTERN>
  <KEY> 反撃開始 </KEY> <ADD> 反撃開始 </ADD>
</ABSINDEX>
<ABSINDEX> サヨナラ勝ち <CLASS> インニング </CLASS>
  <PATTERN>
    ((インニング数 >= 9) & (表裏 = "裏") & (攻撃チーム得点 <= 守備チーム得点)),
    加点+, (攻撃チーム得点 > 守備チーム得点)
  </PATTERN>
  <KEY> インニング開始[(インニング数 >= 9) & (表裏 = "裏")] </KEY>
  <ADD> 加点[$] </ADD>
</ABSINDEX>
<ABSINDEX> 4番打者チャンスを逃す <CLASS> 打席 </CLASS>
  <PATTERN> ((打順 = 4) & ((二塁ランナー = ON) | (三塁ランナー = ON))),
  アウト,$
  </PATTERN> <KEY> 打席[打順 = 4] </KEY> <ADD> 打席 </ADD>
</ABSINDEX>
#パラメータ定義
<PARAMETER> 攻撃レベル <CLASS> 打席 </CLASS>
  <RULE> ヒット[打者=$who]+2: 加点[打者=$who]+2: ホームラン[打者=$who]+4:
  進塁[走者=$who]+1: エラー[選手=$who]+1: 先制[チーム=$who]+2:
  同点[チーム=$steam]+2: 逆転[チーム=$steam]+4
  </RULE> </PARAMETER>
<PARAMETER> 投手レベル <CLASS> 打席 </CLASS>
  <RULE> ストライク[投手=$who]+1: 三振[投手=$who]+2: アウト[投手=$who]+2
  </RULE> </PARAMETER>
<PARAMETER> 興奮レベル <CLASS> 打席 </CLASS>
  <RULE> 打席開始[打順=[3-5], 選手=$who]+2:
  打席開始[三塁ランナー = ON, 選手=$who]+2 </RULE> </PARAMETER>
```

図 8 野球映像におけるスキーマ定義
Fig.8 Schema Definition for Baseball

方, $s < sb$ の時は, 守備的な投手戦であると解釈し, 該当チーム攻撃時の($p1w, p2w, p3w$)を, (0,1,1)として計算する。我々は, 現在 sb の値を 0.3 に設定している。

③ 各基本インデックスの付加されている映像上の位置に対して, 重要度値 pp を計算する。

4.3 ダイジェスト映像作成関数 : digest

$digest$ のアルゴリズムを図 9 に示す。 $digest$ ではまず, 重要度値 pp に従い基本インデックス集合をソートする。ソート順に従い, 基本インデックス i_j を含む映像シーン集合 $\{V_{j_k}\}$ を取り出す。野球映像について嗜好錯誤した結果, i_j を含む $\{V_{j_k}\}$ は, 投球からその打席終了までの間とする方式が最適であると, 我々は判断した。関数 get_scene では, i_j の直前の投球開始インデックスから i_j 直後の打席開始インデックスあるいはイニング開始インデックスまでを切り出し範囲とした。

次に得られた $\{V_{j_k}\}$ の長さを求める。これまでにシステムが切り出した映像の長さの合計と, 求めた $\{V_{j_k}\}$ の長さを加算したものが, 視聴者が指定したダイジェスト総時間を超えないならば, $\{V_{j_k}\}$ はそのままダイジェスト映像シーンのリスト vd に加えられる。長過ぎて指定時間を超えてしまう場合は, 処理中の $\{V_{j_k}\}$ から特に大事な部分だけを切り出し, 指定時間以内に収まるようにする。その調整関数が cut である。現在 cut では, $\{V_{j_k}\}$ に対応する基本インデックス集合 $\{i_{j_k}\}$ のうち, 時間的に最も早い基本インデックスを求め, その基本インデックスから指定時間 $time$ 分を切り出している。

4.4 抽象インデックス

図 8 に示すスキーマ定義例では, 抽象インデックスとして「同点」, 「逆転」を定義している。この抽象インデックスを利用して, 攻撃レベルパラメータの計算ルールが記述されている。また, 「サヨナラ勝ち」と「反撃及ばず」, 「4 番打者チャンスを逃す」などの抽象インデックスも定義されている。「チャンスを逃す」という抽象用語の定義は主観によるが, ここでは, 一つの試合中, 「2 塁あるいは 3 塁に走者がいる打席において加点できなかった」という内容で定義している。

4.5 検索インタフェース

野球映像用ダイジェスト作成システムの試作にあたり, TV 視聴者向けの操作性の良い検索 IF を開発した¹⁾。本検索 IF では, 嗜好項目の入力及び選択を容易にするために, 嗜好項目の集合 $\{f_{i_k}\}$ を画面上に選択ボタンとして並べ, その重み w_{i_k} はスライダバーとして調節できるようにした。 $\{f_{i_k}\}$ は映像の種別に依存

digest(pp, digest_legth)

```
{
//重要度値 pp の絶対値の順で基本インデックス をソート
l = sort(l, pp);
//基本インデックスのソート順に従い, i_j を含む映像
//シーン{V_{j_k}}を取り出す
while(l) {
{V_{j_k}} = get_scene(i_j); //投球開始から打席終了まで
//各シーンが既に取り出し済みかどうかチェック
vplength=0;
foreach v in {V_{j_k}} {
if(v->flag) continue;
else vplength += length(v);
}
if (vlength + vplength < digestst_length) {
//希望ダイジェスト時間を超えない場合は
//そのまま vd に追加
add(vd, {V_{j_k}});
vlength += vplength;
foreach v in {V_{j_k}} {
if(v->flag) continue;
else v->flag=TRUE;
}
} else {
//最後の映像切り出し処理
//起点インデックス切り出し
p_k = key_index({V_{j_k}});
//起点インデックス p_k を中心とした映像切り出し
time = digestst_length - vlength;
add(vd, cut(p_k, {V_{j_k}}, time));
break;
}
}
// 得られた映像シーンのリストを時系列でソート
vd = sort_by_time(vd);
}
```

digestst_length: 視聴者が指定したダイジェスト総時間
vlength: 切り出した映像の合計時間, 初期値 0
vplength: 映像 $\{V_{j_k}\}$ の切り出し長さ
vd: 切り出したダイジェスト映像シーンのリスト
p_k: 切り出しの起点となる基本インデックス

図 9 digest 関数
Fig.9 Function digest

するので, その映像により予め選択可能な嗜好項目の種別を与える。野球の場合, 嗜好項目の種別として, チームと選手を選んだ。視聴者は嗜好情報入力画面を通して自分のお気に入りのチームと選手を登録する(図 10)。

検索条件としては, (1)ダイジェストの長さ, と(2)各カットの長さの割合がスライダバーにより指定できる。本検索 IF では, 図 11 に示すように, 設定された嗜好項目と検索条件を同じ画面に並べた。

4.6 他の番組への適応

野球以外の番組へ適応する場合, どの部分が汎用で, どの部分を開発すべきかを説明する。



図 10 嗜好情報設定画面

Fig.10 User Interface for Setting Preferences

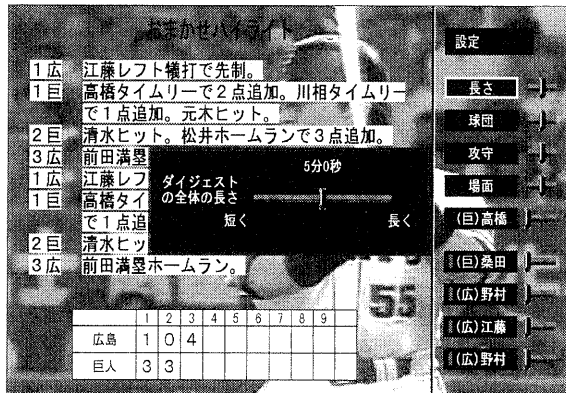


図 11 検索条件設定画面

Fig.11 User Interface for Setting Retrieval Conditions

映像スキーマ定義言語は汎用である。よって、他の種別の番組に対して適応することも可能である。但し、現バージョンの映像スキーマ定義言語では、ルール処理記述能力を、R の計算処理のみを対象として、四則演算に限定してある。現在、野球番組及びサッカー番組で検証した結果、四則演算で十分であると考えているが、必要であれば、一般的処理も記述できるように拡張する。その場合は一般的な処理内容の関数を予め各番組視聴システムで作成しておき、それら呼び出せるように仕様拡張すれば対応可能である。

ダイジェスト作成モデルにおいて、我々が野球用に特化して開発したモジュールは、calc 関数と digest 関数及び検索 IF である。これらのモジュールは C++ などの既存のプログラミング言語で開発可能である。例えばサッカー番組を対象とする場合は、現時点ではこれらのモジュールを新たに開発しなければならない。しかし第 5.2 節で述べたような calc 関数における重要度判定パラメータの重み付け加算処理を映像スキーマ定義言語で定義できるようにすれば、番組に依存した重要場面の判定処理をシステムから切り離すことが可能となる。また digest 関数における get_scene 関数

によるシーン切り出し処理の定義も映像スキーマ定義言語で記述できるようにすれば、汎用的なシーン切り出し処理を実現可能である。このような拡張を行うことにより、汎用性はさらに向上すると考えられる。

検索 IF は番組に依存する部分が非常に多いが、これに関しても、映像スキーマ定義言語で定義されている抽象インデックスやパラメータの名称などを利用した自動的な UIF コード生成モジュールの検討も行っている。

5. 試作システムの評価

本章では、試作したダイジェスト視聴システムの評価を行なう。評価の基準として明確なものはないので、我々は以下の方式により評価を行うとした。

- 入手できた 2 本の野球番組映像を用いる。
- ダイジェスト作成結果が正しいか否かは、ダイジェストで選択されたシーンが以下の WEB サイトで人間が選び出している場面の集合に含まれているかで判定した。

A. アソボウズ：<http://www.asobous.co.jp>

B. 日本テレビ劇空間プロ野球：<http://www.ntv.co.jp>

以下、それぞれ A サイト、B サイトとして参照する。上記サイトには人の手を介して選択されたダイジェスト説明が載っている。ここで説明されているシーンを正解シーンとする。1 回のダイジェスト作成の結果、検索されたシーン数と、上記 A, B サイトの正解シーン数を比べ、以下のような適合率及び再現率を其々求めた。

適合率=(検索結果 and 正解)の数 / 検索結果の数
再現率=(検索結果 and 正解)の数 / 正解数

- 嗜好情報によりパーソナルなダイジェストが作られていることを知る指標として、パーソナルなダイジェスト作成に対しても適合率、再現率を求めることとした。一般に、嗜好の度合いが高ければ高いほど中立的なシーンの比率が低くなるので、適合率、再現率共に低くなる。

以下に使用した 2 本の番組の内容を示す。

① 1998 年 10 月 3 日 巨人対広島戦

この試合は、1 回表、広島・江藤の先制犠打で始まり、続く 1 回裏、高橋、川相のタイムリーヒットで巨人がすぐに逆転。さらに 2 回裏、巨人・松井のスリーランホームランで 6 対 1 となる。6 回表に広島・江藤のホームランで 2 点差まで追い上げるが、8 回裏、巨人の追加点で勝負が決まる、という経過で行われたゲームである。両チームとも得点率 s が 0.3 を超えているため、我々のアルゴリズムでは打撃戦と判定される。

	1	2	3	4	5	6	7	8	9	
広島	1	0	0	0	0	3	0	0	0	4
巨人	3	3	0	0	0	0	0	1	x	7

② 1998年9月24日 中日対横浜

この試合は、前半までは、中日、横浜ともにピッチャーの好投により得点が押さえられている。6回の裏、ついに横浜の攻撃が開始され、4点追加。続く7回裏にも2点追加。8回表に中日がようやく1点をあげるが、反撃及ばずに7対1の大差で勝負がついた試合である。中日の得点率は0.11であり、横浜の得点率は0.78となるため、中日にとっては投手戦であり、横浜にとっては打撃戦であったと判定できる。

	1	2	3	4	5	6	7	8	9	
中日	0	0	0	0	0	0	0	1	0	1
横浜	1	0	0	0	0	4	2	0	x	7

上記①、②の各試合に対して以下の条件でダイジェスト作成を行った。

- ・ダイジェスト映像総時間：3分
- ・各シーンの最大長：30秒

(条件1):視聴者の嗜好を反映させない場合

(条件2):嗜好情報(選手x, 0.4)を設定した場合

(条件3):嗜好情報(選手x, 0.8)を設定した場合

選手xとして、各試合に対して2人を選んでダイジェスト作成を行った。②の試合は投手戦なので、xとして投手も選択した。「各シーンの最大長」は、切り出す映像シーン其々の最大の長さを定義している。我々のシステムでは、投球シーン(各打席の最後の投球開始から、打席終了まで)を基本として切り出しているため、ホームランやタイムリーヒットのシーンは長くなり、アウトや三振のシーンは短くなる。

表3 ダイジェスト視聴システム評価結果

Tab.3 Evaluation Results of a Digest Viewing System

試合	方式	A サイト	B サイト	パーソナル：シーン数	パーソナル：時間数		
		適合率	適合率				
		再現率	再現率				
①	条件 1: 嗜好無し	60%	60%	1	15sec		
		100%	100%				
	条件 2: (江藤,0.4)	60%	60%				
		100%	100%				
	条件 3: (江藤,0.8)	50%	50%			3	40sec
		80%	80%				
条件 2: (清原,0.4)	60%	60%	2	28sec			
	100%	100%					
条件 3: (清原,0.8)	40%	40%	4	60sec			
	70%	70%					
②	条件 1: 嗜好無し	70%		1	15sec		
		84%					
	条件 2: (石井,0.4)	60%					
		80%					
	条件 3: (石井,0.8)	40%				3	50sec
		60%					
条件 2: (野村,0.4)	53%		2	30sec			
	60%						
条件 3: (野村,0.8)	38%		4	75 sec			
	46%						

表3に、評価結果を示す。表中、アソボウズのサイトと日本テレビのサイトに其々に対して適合率及び、再現率を計算した。②の中日対横浜は、アソボウズのデータのみであったため、Aサイトのみ記載した。またパーソナル部分の比率が分かるように、中立的な正解以外の部分で選択されたシーン数及びその時間を記した。図12は①の試合における[条件1:嗜好無し]の映像解析結果を示している。図13は、検索結果である。図14は①の試合における[条件3:(江藤,0.8)]の結果を示している。

①の試合の[条件1]において、再現率はAサイト、Bサイトとも100%であった。また①の試合における[条件3:(江藤,0.8)]の場合、嗜好の度合いが高いため、中立的な立場で選択されるシーンよりも江藤のシーンが優先されている。そのためAサイト、Bサイトに対する適合率、再現率が低くなっている。

表3から、全体にわたり、嗜好の重みが大きくなる

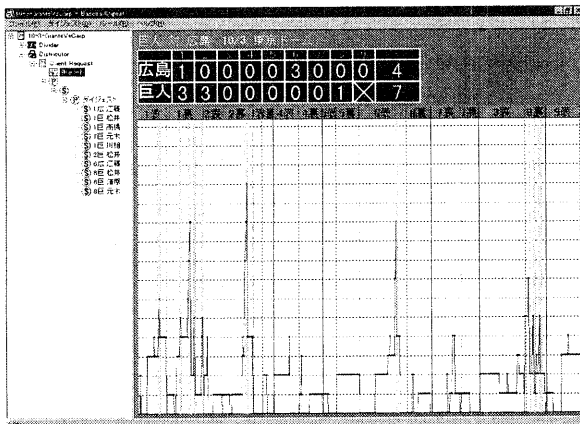


図12 解析結果
Fig.12 Result of Analysis

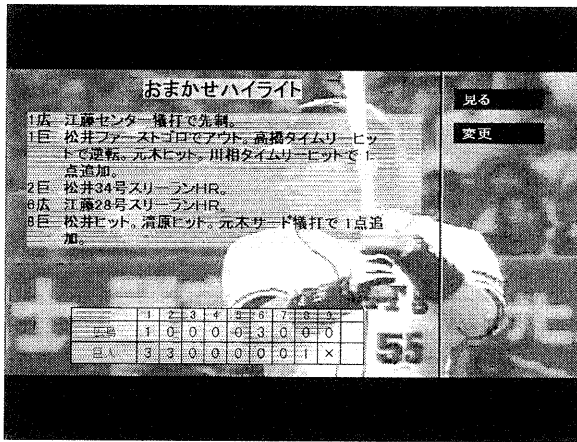


図 13 検索 IF の結果
Fig.13 Result of User Interface

につれて、適合率が減少、再現率も減少、パーソナルシーン数及び時間の増大、という傾向が見られる。これは嗜好情報の利用により、パーソナルなダイジェスト作成が正しく行われていることを意味する。また、嗜好無しの場合の適合率の高さから、現仕様で正しく重要場面を判定していることが分かる。

次に今回の評価に得られた今後の改良点を述べる。

現在の *digest* 関数の仕様では、重要度判定パラメータ値の高い順に映像の切り出しを行っているが、本来は、映像上の重要度が高い事象についてその因果関係も含めて切り出すべきである。実際上記の日本テレビの Web サイトでは、重要なシーンに関しては、その原因となる事象から詳細に解説を行っている。重要事象を詳細に説明するためには、その事象の前後関係が分かるようなシーンの切り出しが望まれる。例えば「2 回裏、松井のスリーラン HR」という事象に関して、その時点で出塁している打者は誰であるか、なぜ出塁したか、という因果関係の情報を視聴者に提示することにより「スリーラン HR」という重要な事象の状況の理解が促進される。しかし、現在の *digest* 関数

- | |
|--|
| 1 回表: 広島, 江藤センター犠打で先制. |
| 1 回裏: 巨人, 松井ファーストゴロでアウト. 高橋タイムリーヒットで逆転. 元木ヒット. |
| 2 回裏: 巨人, 松井 34 号スリーラン HR. |
| 4 回表: 広島, 江藤レフトフライでアウト. |
| 6 回表: 広島, 江藤 28 号スリーラン HR. |
| 8 回表: 広島, 江藤センターゴロでアウト. |
| 8 回裏: 巨人, 松井ヒット. |
| 9 回表: 広島, 江藤三振でアウト. |

図 14 検索結果(条件 3: (江藤,0.8))
Fig.14 Result of Retrieve [条件 3: (江藤,0.8)]

では必ずしも重要な事象の前後関係までは優先して取ってはいない。それは我々のアルゴリズムが、計算の結果得られた重要度値 *pp* を単純にソートするだけであり、「因果関係」に対する処理を行っていないからである。今後は、各事象間の関連について定義し、因果関係を持つ一連の映像を切り出す処理が必要と考える。

また現在、攻撃レベルと投手レベルのどちらを適用するかは得点率 *s* 値で判定しているが、②の試合のように序盤は投手が好調で、途中から崩れて打たれてしまう、といった場合に対応しきれていない。そのため条件 1 において、①の試合に比べ、再現率が低くなっている。試合のイニング単位、あるいは序盤、中盤、終盤における *s* 値を算出し、イニングごとに適用するレベルを決定するようにすれば、よりきめの細かいダイジェスト映像を作成できると考える。

6. 関連研究

本章では、番組のダイジェスト作成に関連する研究について述べる。

従来のダイジェスト作成方式には、どのようなインデックスを用いるかに関して、大きく以下の 2 つのアプローチ方法がある。

① 画像分析および、音声認識結果への自然言語理解の利用により、カメラカット、物体トラッキング、話者の変化、音声や BGM のタイミング、単語による文脈の変化などの情報を組み合わせ、対応する映像を切り出して知的にスキミングを行なう^{12)~16)}。

② 内容を見ながら、手作業で映像編集、あるいは内容に基づくインデックスを手作業で振り、そのインデックスを利用して検索し、ダイジェストを作成する^{17)~19)}。

上記①と②の違いは、①は画像分析及び音声認識技術によってインデックスを生成するのでインデックスが画像や音声の特徴量に強く関係するのに対し、②のインデックスは人間が判断して付けるので、直接的にその事象名を付加できる点である。例えば、①のアプローチ法でサッカー番組にインデックスを付加する場合、現在の画像分析の技術でゴールシーンを見つけることはできても、パスやドリブル、ヘディングなどの動作を区別することは困難である。最近では動画に対する特徴量抽出技術も向上しており部分的には実用レベルに達している²⁰⁾。しかし、我々は近い将来の実用を目指して検証実験を行っているため、実現が容易な②のアプローチを採用することにした。②の方式の問題は、インデックス付加作業にコストがかかる点であるが、

我々の方式では現実的なコストで実現可能である。

次に上記②のアプローチを使った他のダイジェスト作成方式との比較を行う。我々のダイジェスト方式は重要度判定パラメータとして適当な指標を定め、その値が高いところを重要場面としている。一方、我々の方式に対し、「試合の流れを表現する」ことを目的とし、シナリオテンプレートによるダイジェスト作成方式を提唱しているグループもある²¹⁾。指標計算による方法はシナリオテンプレート方式に比べて以下の点で優れていると我々は考える。

(1) 出現する全てのストーリーを予め予測してシナリオテンプレートを作成しておくことは困難である。特に試合構造の明確でないサッカーのようなスポーツの場合に対してストーリーテンプレートを記述することは困難と考える。

(2) 指標及びその計算ルールを定義するのに比較してストーリーは多数あるので、テンプレート作成コストが大きくなる。

我々はサッカーの試合に対しても指標による方式を検証したが、状況レベルパラメータを定義し、その値の変化の様子を抽象インデックスで定義することで試合の流れも表現できた⁹⁾。

他のダイジェスト作成方式として、イベントのエントロピーを定義し、イベント出現系列をマルコフ過程で定義した確率モデルを使って特異な場面を探す方式²²⁾があるが、特異な場面が意味的に重要であるとは限らないため、確率モデルだけではダイジェスト作成が不十分であると考えられる。

視聴者の望むシーンを取り出すための各種の手法も研究提案されている。多くは、映像上での代数演算を定義し、それらを組み合わせることで視聴者が求めるシーンを合成する、というものである¹⁷⁾。我々は野球番組に対するダイジェスト作成システムを試作した際、抽象インデックスの定義にこれらの代数演算の利用することを検討した。しかし、TV視聴者にとって、代数演算の記述は非常に困難であり、またそれらの演算だけでは記述が不十分な可能性もある。そこで我々は、抽象度の高い用語に対応する抽象インデックスを、既に振られている基本的インデックスの状態遷移パターンで記述することアプローチを取ることにした。

7. まとめ

本稿では、番組インデックスを利用したダイジェスト視聴を実現するため、ダイジェスト作成モデル及びそのシステムについて説明した。本システムの特長は視聴者が個人の嗜好にあったパーソナルなダイジェス

トを作成可能なことと、抽象インデックスを定義することで抽象用語による検索を可能とすることである。

抽象インデックスの利用において、我々のシステムを、受信端末上のDBと連携させることにより、より主観的な検索が可能になる。例えば、「今年限りかもしれない選手の活躍シーンを探せ」といった検索を行なう場合、「今年限りかもしれない」といった主観的判断は、DB上で自分用に定義する必要がある。今後は、こうしたDBとの連携による抽象検索の利用について検討していく予定である。

本稿では野球番組に対する事例を行ったが、サッカー番組に対しても検証実験を行っている。その結果、(A)サッカーの場合でも重要度判定ルールは四則演算のみで記述可能であること、(B)野球のような明確な試合構造が無くても独自に区切りイベントを定義することで適当な長さの区間を設定できること、(C)試合の流れも状況レベルパラメータを定義することで表現可能なことが検証できた⁹⁾。

今後は、ニュースなどの他のコンテンツにも我々のダイジェスト作成モデルの技術を適用し、我々の方式の検証を行っていきたいと考えている。TVML²³⁾の仮想キャラクターなども利用して映像検索結果を視聴者により効果的に提示する方法についても検討していく予定である。

謝辞 野球番組ダイジェスト視聴の検討およびこの発表にあたり野球映像を提供していただいた、日本テレビ放送網株式会社に深く感謝する。

参考文献

- 1) 権野善久, 山岸靖明, 木村武史: 情報放送への技術展望, 1998年映像情報メディア学会年次大会予稿集, pp.88-89(1998).
- 2) 社団法人電波産業会: デジタル放送に使用する番組配列情報, ARIB STD-B10 1.2版(1999).
- 3) 情報放送システム研究所編: 特集:「番組インデックスとメタデータ」, 機関誌 *ible*(アイブル) 第2号, <http://www.ibl.co.jp/ible/> (1998).
- 4) 橋本隆子, 白田由香利: 番組インデックスを利用したダイジェスト視聴方式の検討, 映像情報メディア学会技術報告, Vol. 23, No. 28, pp. 7-12(1999).
- 5) 橋本隆子, 白田由香利, 真野博子, 飯沢篤志: 番組インデックスを利用したダイジェスト作成方式の試作, 第10回データ工学ワークショップ(DEWS'99)予稿集CD-ROM(1999).
- 6) 久保木準一, 橋本隆子, 國枝孝之, 脇田由喜, 白田由香利, 木村武史: 番組制作のためのメタデータ付加工法～汎用イベントリスト(GEL), 映像情報メディア学会技術報告, Vol. 23, No. 28, pp. 1-6(1999).
- 7) 橋本隆子, 白田由香利, 真野博子, 飯沢篤志: TV受信端末におけるダイジェスト作成及び視聴システムの試作, 情報処理学会研究会報告, DBS-119, Vol.99, No.23, pp.133-138(1999).

- 8) 橋本隆子, 白田由香利, 真野博子, 飯沢篤志: 映像情報に対する抽象度の高い用語による検索手法, アドバンス・データベース・シンポジウム'99予稿集, pp.49-58(1999).
- 9) 橋本隆子, 白田由香利, 飯沢篤志: サッカー番組のダイジェスト作成システムの試作, 第11回データ工学ワークショップDEWS2000予稿集CD-ROM(1999).
- 10) 十河孝至, ブラダン スジツ, 田島敬史, 田中克己: ビデオ検索モデルとしてのGlue演算における効率化フィルタについて, アドバンス・データベース・シンポジウム'99予稿集, pp.29-38(1999).
- 11) 真野博子, 白田由香利, 橋本隆子, 飯沢篤志: TV受信端末におけるダイジェスト視聴用インタフェース, 映像情報メディア学会年次大会予稿集, pp.65-66(1999).
- 12) Smith, M. and Kanade, T.: Video Skimming for Quick Browsing based on Audio and Image Characterization, *Tech. Rep. CMU-CS-95-186*, School of Computer Science, Carnegie Mellon University(1995).
- 13) 金出武雄, 佐藤真一: Informedia:CMUデジタルビデオライブラリプロジェクト, 情報処理, Vol. 37, No. 9, pp.841-847(1996).
- 14) 有木康雄: DCT特徴のクラスタリングに基づくニュース映像のカット検出と記事切り出し, 電子情報通信学会論文誌, Vol. J80-D-II, No. 9, pp. 2421-2427(1997).
- 15) 有木康雄, 片山雅夫, 五十棲晋二: テロップ文字認識に基づくTVニュース記事の自動分類, 情報処理学会研究会報告, 98-DBS-116-28(1998).
- 16) 鎌原淳三, 香取啓志, 下條真司, 宮原秀夫, 西尾章次郎: 自動再構成を行うマルチメディアニュース推薦システム, 情報処理学会論文誌:データベース, Vol. 40, No.SIG3(TOD1), pp.124-133(1999).
- 17) Oomoto, E. and Tanaka, K.: OVID: Design and Implementation of a Video-Object Database System, *IEEE Trans. on Knowledge and Data Engineering*, Vol. 5, No. 4, pp. 629-641(1993).
- 18) 是津耕司, 上原邦昭, 田中克己: 映像の意味的構造の発見と動的Skimmingへの応用, 第10回データ工学ワークショップ(DEWS'98)CD-ROM(1998).
- 19) 牛尼剛聡, 渡邊豊英: イベントアクティビティ・モデルに基づくシーン検索における可変的な役割の利用, 情報処理学会論文誌:データベース, Vol. 40, No.SIG3(TOD1), pp. 114-123(1999).
- 20) 宮森 恒: 動きで映像を検索できる新技術, CRL(郵政省通信総合研究所)第97回研究発表会予稿集, pp. 29-32(1999).
- 21) 植田和憲, 鎌原淳三, 下條真司, 宮原秀夫: シナリオテンプレートによるストーリー性を重視したダイジェスト生成機構, 情報処理学会研究会報告, DBS-119-24, pp.139-144(1999).
- 22) 牛尼剛聡, 渡邊豊英: ビデオ・ブラウジングのための確率モデルに基づく概略表現生成手法, アドバンス・データベース・シンポジウム'99予稿集, pp. 39-48(1999).
- 23) 林正樹: 番組記述言語TVMLを使った情報の番組化, 情報処理学会研究会報告, 2000-DBS-120-13, pp. 91-98(2000).

(平成11年12月20日受付)

(平成12年 3月27日採録)

(担当編集委員 有川 正俊)



橋本 隆子 (正会員)

1985年御茶ノ水女子大学理学部化学科卒業。同年(株)リコー入社。現在同社ソフトウェア研究所勤務。データベース・システムの研究開発に従事。1996年4月より、(株)次世代情報放送システム研究所に兼任出向し、デジタル放送環境における受信端末上の映像検索の研究を行っている。情報処理学会会員。



白田 由香利 (正会員)

1989年東京大学大学院理学系研究科情報科学専攻博士課程修了。理学博士。同年(株)リコー入社。現在同社ソフトウェア研究所に勤務。データベース・システムの研究に従事。1997年、(株)次世代情報放送システム研究所へ兼任出向し、放送機構を用いた超大規模分散データベースシステムの研究開発に従事。情報処理学会, 電子情報通信学会, IEEE Computer Society, ACM 各会員。



真野 博子

1996年スタンフォード大学修士課程修了。同年(株)リコー入社。現在同社ソフトウェア研究所に勤務。データベース・システムの研究開発に従事。1998年より1999年3月まで、(株)次世代情報放送システム研究所へ兼任出向し、放送機構を用いた超大規模分散データベースシステムの研究開発に従事。ACM 会員。



飯沢 篤志 (正会員)

1982年東京大学大学院理学系研究科情報科学専門課程修士課程修了。1985年(株)リコー入社。現在同社ソフトウェア研究所に勤務。データベース・システムの研究に従事。1997年から(株)次世代情報放送システム研究所へ兼任出向し、放送機構を用いた超大規模分散データベースシステムの研究開発に従事。情報処理学会, 電子情報通信学会, ソフトウェア科学会各会員。