

イベントと状況変化の依存関係に基づいた 野球中継のインデキシング支援

広部 一弥 牛尼 剛聡 酒井 宏治
孫 魯英 渡邊 豊英

名古屋大学大学院工学研究科情報工学専攻
hirobe@watanabe.nuie.nagoya-u.ac.jp

我々は、野球中継を対象とした動画データベース・システム STRIKE を開発中である。STRIKE はイベント・アクション・モデルに基づいたシーン検索を実現するが、この検索を行うためには、動画の中で観測可能なイベントを動画フレームにインデックスとして対応付ける必要がある。我々は、STRIKE におけるインデキシング作業を支援するために、画像処理を利用して動画の中で発生したと考えられるイベント候補をデータベース作成者に提示する機構を開発している。本稿では、野球中継中のテロップから状況を判断し、テロップが不変である時区間において発生したと考えられるイベント候補を推定する手法を提案する。本手法では、状況の時間的前後関係を定義し、イベントが生じさせる状況の変化を表す状況遷移パターンを導入する。テロップから判断可能な状況の変化に一致する状況遷移パターンの組合せを計算することで、時区間中で発生したと考えられるイベント候補を推定可能である。

A Method for Assisting Users to Create Indexes on Movies Based on Dependency between Event and Change of Situation

Kazuya HIROBE Taketoshi USHIAMA Koji SAKAI
Luying SUN Toyohide WATANABE

Department of Information Engineering, Graduate School of Engineering, Nagoya
University

We are developing the movie database system STRIKE, which provide facilities for retrieving scenes in a movie of baseball game based on event-action model. STRIKE needs to create indexes on the frames in a movie with events. In this paper, we propose a method to deduce event sequences that occurred in a movie from changes of situation. For this deductions, we define a temporal order on situations, and introduce situation transition patterns that represent situation translations caused by event occurrences. Using situation translation patterns, event occurrences are deduced from the situations that are decided from telops in a movie.

1 まえがき

動画は、対象世界における活動の伝達、記録、分析等を目的として、さまざまな分野で利用されている。近年、計算機上で動画を効率的に管理、利用する要求が高まり、動画データベース・システムに関する研究が活発化している [1]。動画データベース・システムに要求される機能の一つにシーン検索がある [2, 3, 4, 5, 6]。動画の中で着目する実体や活動を捉える概念的レベルに応じて多種多様なシーンを考えることができるため、要求するシーンが利用者の視点に応じて異なる。我々は、野球中継を対象として、利用者の視点に基づくシーン検索が可能な動画検索システム STRIKE を開発中である [7]。STRIKE は動画中の実体の特性の変化と実体間の相互作用をイベントとしてモデル化し、動画内容をイベント系列として表現するイベント-アクション・モデルに基づくシーン検索を実現する。

イベント-アクション・モデルでは、データベース作成者が一つの野球の試合に対して数千個のイベントをインデックスとして付加する必要があるため、インデキシングの労力を軽減することが期待される。そこで、我々は STRIKE におけるインデキシング作業を支援する機構を開発中である。我々のインデキシング支援機構では、画像処理を利用して動画中の時区間で発生する可能性のあるイベント候補を推定し、データベース作成者がイベント候補から適切なものを選択することでインデキシングを可能とすることを目指している。本稿では、画像処理により抽出された情報を利用して動画中で発生したイベント候補を推定する手法を提案する。

イベントは、動画中の人物や物体の活動に基づいて発生する。しかし、一般に画像処理を用いて抽出可能な情報は抽象化レベルが低いため、こうした活動を自動抽出するのは困難である [8]。野球中継では、投手の投球するカットなどで試合の状況を表すテロップが現れる。この表示は、画像処理による抽出が比較的容易である [9, 10, 11]。本稿では、テロップが状況を表すことを利用して、テロップから観測可能な状況の変化から、イベント候補を推定する手法を提案する。なお、本稿ではテロップは正しく認識可能であることを前提とする。

2 イベント-アクション・モデル

イベント-アクション・モデルでは実体の特性の変化と実体間の相互作用をイベントとして表現し、動画内容をイベントの系列としてモデル化する。イベ

ント系列を構成するイベントを動画フレームにインデックスとして対応付けることにより、利用者はイベント系列の部分系列を指定することで、シーン検索が可能となる。

実体とは、識別可能であり、かつ任意の時刻においてその状態を考えることができる対象世界上のモノである。動画に表現される実体には物理実体と概念実体に大別できる。物理実体は、対象世界の物理物体、および編集時に付加されるテロップ等の記号である。物理実体は、動画のフレーム内の特定の閉領域に対応付け可能である。一方、物理実体以外の実体を概念実体と呼ぶ。概念実体の状態は、他の実体の状態に基づいて一意に決定される。例えば、野球の試合におけるそれぞれの選手は物理実体であり、「ストライク・カウント」は概念実体である。実体は文字列として表現され、同一の文字列は同一の実体とみなす。動画中にはさまざまな実体を観測可能であるが、対象とする実体のドメインを規定するために実体の型を用いる。実体の型は2項 (tn, EN) である。ここで、 tn は型名であり、 EN は実体の集合である。野球の試合における実体の例を表1に示す。

実体の特性の変化および実体の特性を変化させる実体間の相互作用をイベントと呼ぶ。イベントは2項 $(id, [en_1, \dots, en_n])$ として表現する。ここで、 id は識別子であり、 en_i ($1 \leq i \leq n$) は実体である。識別子はそれぞれのイベントを区別し、引数 $[en_1, \dots, en_n]$ はイベントに関係する実体を表す。イベントのドメインを規定するためにイベント型を用いる。イベント型は3項 $(tn, [tn_1, \dots, tn_n], EV)$ である。ここで、 tn は型名であり、 tn_i ($1 \leq i \leq n$) は実体型の型名であり、 EV はイベントの集合である。 EV の要素を型 tn のインスタンスと呼ぶ。 $[tn_1, \dots, tn_n]$ はインスタンスの引数の型を指定する。すなわち、 EV の要素の i 番目の引数は実体型 tn_i のインスタンスでなければならない。野球の試合におけるイベントの例を表2に示す。

3 状況とテロップ

3.1 状況

イベント-アクション・モデルでは、動画中の実体の特性の変化に着目するため、任意の時刻における実体の状態を陽に表現していない。そこで、任意の時刻の実体の状態を表現するために状況の概念を導入する。状況は実体間の2項関係の集合 $\{en_{11}:en_{12}, en_{22}:en_{22}, \dots, en_{n1}:en_{n2}\}$ である。こ

表 1: 野球の試合における実体の例

型名	実体
選手	"イチロー", "吉田"等
塁	"本塁", "1 塁", "2 塁", "3 塁"
塁間	"本-1 塁間", "1-2 塁間", "2-3 塁間", "3- 本塁間"
ボール	"ボール"
ボールの状態	"投手", "捕手", "野手", "投球されている", "フライ", "ゴロ"
グラウンド	"フェア・グラウンド", "ファール・グラウンド"
観客席	"内野席", "外野席"
試合	"オリックス対ヤクルト"等
カウント	"ストライク・カウント", "ボール・カウント", "アウト・カウント", "イニング"
得点	"先攻チームの得点", "後攻チームの得点"
数値	"1", "2"等

表 2: 野球の試合におけるイベントの例

イベントの型	説明
(throw,[選手, ボール])	選手がボールを投げる
(catch,[選手, ボール])	選手がボールを捕給する
(drop,[選手, ボール])	選手がボールを落とす
(touch-to,[選手, ボール, 選手])	選手がボールで選手に触れる
(touch,[選手, ボール])	選手がボールに当たる
(reach,[選手, 塁])	選手が塁に触れる
(bat,[選手, ボール])	選手がボールを打つ
(is-batter,[選手])	選手が打者になる
(is-out,[選手])	選手がアウトになる
(is-safe,[選手])	選手がセーフになる
(bound-in,[ボール, フィールド])	ボールが地面と接触する
(enter,[ボール, スタンド])	ボールがスタンドに入る
(strike,[試合])	ストライク
(ball,[試合])	ボール
(foul,[試合])	ファウル
(four-ball,[試合])	四球
(change,[試合])	チェンジになる
(playball,[試合])	プレイボール
(interrupt,[試合])	中断開始
(play,[試合])	試合再開
(game-set,[試合])	ゲームセット
(is-set-as,[カウント, 数値])	カウントを数値にする

表 3: 野球の試合における状況の例

特性	値
"イニング数"	: "3"
"表/裏"	: "表"
"先攻チームの得点"	: "1"
"後攻チームの得点"	: "2"
"アウト・カウント"	: "2"
"ストライク・カウント"	: "1"
"ボール・カウント"	: "0"
"打者"	: "イチロー"
"本-1 塁間"	: "なし"
"1 塁"	: "なし"
"1-2 塁間"	: "なし"
"2 塁"	: "田口"
"2-3 塁間"	: "なし"
"3 塁"	: "なし"
"3- 本塁間"	: "なし"
"ボール"	: "投球されている"

状況は、それを構成する要素によって、他の状況と区別される。2つの状況の属性とその値が同一ならば、その2つの状況は同一である。状況の属性に関するイベントが状況を変化させ、状況が同一である時間区間中には、その状況の属性に関するイベントは発生していない(図1)。

3.2 テロップ

テロップは、編集者が動画像に付加した記号である。動画像中のテロップは以下の2つの目的のために利用される。

1. 対象世界上のある時刻における状況の提示
2. 付加情報の提示

本研究では、画像が表現する世界の活動内容に興味があるので、前者のテロップを対象とする。具体的には以下のテロップを対象とする。

- 先攻チームの得点, 後攻チームの得点
- イニング数, 表/裏

ここで、 $en_{ij}(1 \leq i \leq n, 1 \leq j \leq 2)$ は実体である。状況 s の要素 ($en_1:en_2$) に対して、 en_1 を状況 s の属性と呼び、 en_2 を状況 s における属性 en_1 の値と呼ぶ。野球の試合における状況の例を表3に示す。表中で、属性 " x 塁" および属性 " $x - (x + 1)$ 塁間" は、状況 s における攻撃側の選手の位置を表す。状況の属性と属性値のドメインを規定するために、状況の型を導入する。状況の型は、実体と実体の型の2項関係の集合であり $\{ en_1:ty_1, \dots, en_n:ty_n \}$ と表記する。ここで $en_i(1 \leq i \leq n, 1 \leq j \leq 2)$ は実体であり、 $ty_i(1 \leq i \leq n, 1 \leq j \leq 2)$ は実体の型である。野球の試合における状況の型の例を表4に示す。表3に示した状況は、表4で示した状態の型のインスタンスになっている。

表 4: 野球の試合における状況の型の例

属性	値の型
"イニング数"	: 数値
"表/裏"	: 表/裏
"先攻チームの得点"	: 数値
"後攻チームの得点"	: 数値
"アウト・カウント"	: 数値
"ストライク・カウント"	: 数値
"ボール・カウント"	: 数値
"打者"	: 選手
"投手"	: 選手
"本-1 塁間"	: 選手
"1 塁"	: 選手
"1-2 塁間"	: 選手
"2 塁"	: 選手
"2-3 塁間"	: 選手
"3 塁"	: 選手
"3- 本塁間"	: 選手
"ボール"	: ボールの状態

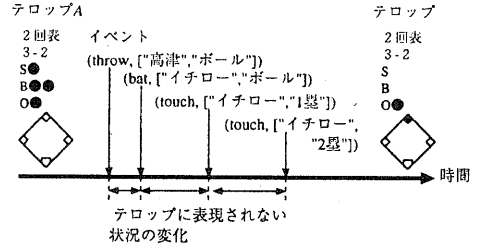


図 2: テロップに表示されない状況の変化

「打者がボールを打つ」が発生した後、「打者が1塁までの区間にいる」という状況になり、イベント「打者が1塁に着く」が発生し「打者が1塁から2塁までの区間にいる」という状況をになり、イベント「打者が2塁に着く」が発生し最終的に「打者が2塁にいる」という状況になると考えられる。しかし、テロップによって提示されるのは「2塁に走者がいる」という状況だけである。したがって、テロップが変化するときには状況を変化させる複数のイベントが発生している可能性がある(図2)。

4 出来事と状況の関係

状況には、時間的な前後関係が存在する。また、状況に応じて発生する発生したイベントが限定され、イベントによって次の状況が決定される。我々はこれを知識として状況の変化から発生したイベントを推定する。

状況は時間的な前後関係を考えることができる。例えば、一つの野球の試合において状況 { イニング:0, 表/裏:表, アウト:2, ストライク:1 } は、状況 { イニング:0, 表/裏:表, アウト:0, ストライク:0 } よりも時間的に後に出現する。これは、2つの状況において属性"イニング数"と"表/裏"の値が同一であるときには、"アウト・カウント"の値"2"は値"0"よりも時間的に後であるからである。このように、状況の属性の値には、変化のパターンが存在し、それらによって、状況の変化のパターンを規定することができる。各状況に他の状況との時間的前後関係を実現するために、属性の値間の時間的な前後関係と、属性間の依存関係を導入する。属性の値に関する時間的な前後関係 V_{\leq} を表5に示す。状況の属性間に依存関係が存在する場合がある。例えば、属性"ストライク・カウント"は属性"アウト・カウント"に規定される。このとき、"ストライク・カウント"は"アウト・カウ

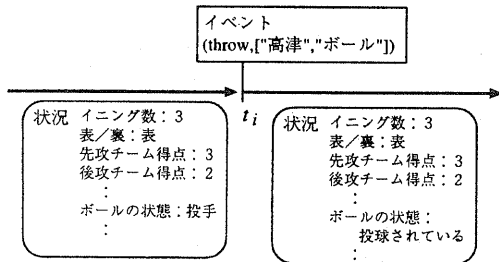


図 1: イベントと状況の変化の例

- アウト・カウント, ストライク・カウント, ボール・カウント
- 各塁の走者の有無

なお、テロップは状況のすべての属性を提示しているとは限らない。

テロップは対象世界の活動に基づいて編集者が付加するため、編集者が状況の変化を認識した後にテロップが変化する。したがって、状況が変化した時刻とテロップが変化した時刻は同一ではない。テロップが変化したとき、前回のテロップの変化から今回の変化までの時区間で状況が変化していることがわかる。状況が変化するのはイベントが発生したときであるから、テロップの変化には含まれた時区間のどこかで、イベントが発生していると考えることができる。

テロップが同一である時区間において、テロップに反映されない状況の変化が存在する場合がある。例えば、(「ボールを打つ」, 「1塁に着く」, 「2塁に着く」)というイベント系列が存在する場合、イベント

ント”に依存している。野球の試合における属性間の依存関係の例を図4に示す。属性の値の時間的前後関係と属性間の依存関係に基づいて、状況間の前後関係 $S_1 \prec S_2$ を以下のように定義する。

[定義] 状況の前後関係

状況の集合 S の要素 s_1, s_2 について、以下の2つの条件のいずれかを満足するとき、 $s_1 \prec s_2$ であるとする。

1. すべての属性 a に対して、 $s_1.a \leq s_2.a$ である。
2. 他の属性と依存関係をもたないすべての属性 a に対して、 $s_1.a \leq s_2.a$ である。さらに、依存する属性をもつ属性 a に対して、 a が依存するすべての属性 a' について、 $s_1.a' = s_2.a'$ かつ $s_1.a \leq s_2.a$ であるような a が存在する。

□

2つの状況が時間的前後関係にないとき、この2つの状況は矛盾関係にあるという。図3に野球の試合における状況の前後関係の例を示す。

動画像中で、状況 s_1 が状況 s_2 に変化した場合、 s_1 である時区間で、状況 s_1 を s_2 に変化させるイベントが発生していると考えられる。例えば、状況 { イニング:2, 表/裏:表, アウト:1, ストライク:1, ボール:1, 先攻チームの得点:3, 後攻チームの得点:2 } は、イベント「打者が空振りする」によって状況 { イニング:2, 表/裏:表, アウト:1, ストライク:2, ボール:1, 先攻チームの得点:3, 後攻チームの得点:2 } に変化する。

対象世界上では状況の変化には直接関与していないイベントが発生している場合がある。これは、状況を変化させるイベントの発生が、他のイベントに依存している場合である。したがって、状況を変化させるのは単一のイベントではなく、イベント系列であると考えることができる。

イベント系列 es が状況 s_1 を状況 s_2 に変化させるとき3項 (es, s_1, s_2) を状況遷移パターンと呼ぶ。ここで s_1 を事前状況と呼び、 s_2 を事後状況と呼ぶ。状況遷移パターンの例を表6に示す。

5 イベント候補の推定

状況遷移パターンを利用して、テロップから観測される状況の変化をもたらししたイベント候補を推定する。

いま、テロップによって観察される状況が A から B に変化したとする。このとき、状況 A を事前条件とする状況パターンの事後状況 A_1 をとしたとき、以

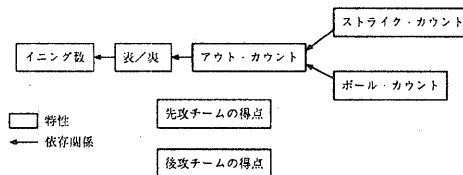


図4: 属性間の優先順序

下の基準でイベント候補を決定する。計算し、

- 1) A_1 が B と一致するなら、状況遷移パターンに含まれるイベントの系列を候補とする。
- 2) A_1 が B と一致せず、 A_1 が B よりも時間的に先攻しているか A_1 が B と矛盾関係にあるならば、その状況遷移パターンに含まれるイベント系列は候補ではない。
- 3) 1), 2) 以外なら、状況 A_1 を事前条件とする状況遷移パターンについて考える必要がある。

状況 A と状況 B には含まれたイベントの系列の集合を求める処理手順を以下に示す。STP は状況遷移パターンの集合であり STP_i はその要素を表す。

```
EventListSet ELS := null;
```

```
Status A,B;
```

```
PatternSet STP;
```

```
deduceSub(EventList EL, Status st) {
  for (i:=1 ; i<=n ; i++) {
    if (st に STP_i が適応できる) {
      ELの最後に STP_i のイベント系列を追加 ;
      st := STP_i 内の事後状況 ;
      if (st == B) {
        ELSにELを追加 ;
        return;
      }else if (st は B よりも
                時間的に後である) {
        return;
      }else if (st と B が矛盾関係にある) {
        return;
      }else {
        guessSub(EL, st);
      }
    }
  }
}
```

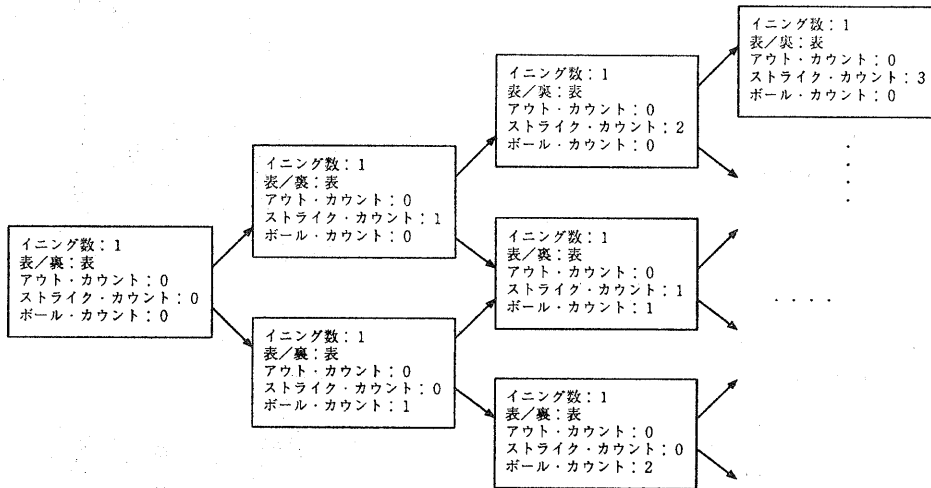


図 3: 状況内の前後関係

表 5: 状況の属性の取り得る値と順序

状況の属性	取り得る値と順序
イニング数	<"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", ...>
表/裏	<"表", "裏">
アウト・カウント	<"0", "1", "2", "3">
ストライク・カウント	<"0", "1", "2", "3">
ボール・カウント	<"0", "1", "2", "3", "4">
先攻チームの得点	<"0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10", ...>
後攻チームの得点	<"0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10", ...>

```

}
}
}
deduce() {
  EventList EL = null;
  deduceSub(EL,A);
  return ELS;
}

```

図 5 に処理の例を示す。図中の実線矩形で囲まれたものが状況である。事前条件から事後条件への状況遷移は、イベント系列を付記した矢印で示される。図中のバツ印を付けられた状況は、テロップ B が示す状況より時間的に先攻するか、テロップ B が示す状況と矛盾関係にある。テロップ A が示す状況からテロップ

B が示す状況への状況遷移パターンを作成し、その経路に含まれるイベント系列が候補の集合である。状況 { イニング:2, 表/裏: 表, アウト:1, ストライク:1, ボール:2, 先攻チームの得点:3, 後攻チームの得点:2 } と、状況 { イニング:2, 表/裏: 表, アウト:1, ストライク:2, ボール:2, 先攻チームの得点:3, 後攻チームの得点:2 } の間に起きたと考えられるイベント系列が表す直観的な意味は、「打者がストライクをとられる」、「打者が打った球がファール・グラウンドに落ちる」、「打者が打った球がファール・スタンドに入る」である。それ以外のイベント系列が変化させる状況は、状況 { イニング:2, 表/裏: 表, アウト:1, ストライク:2, ボール:2, 先攻チームの得点:3, 後攻チームの得点:2 } と比べて時間的に矛盾するか、時間的に後である。

表 6: 状態遷移パターンの例

イベント系列	事前状況	事後状況
(throw, {投手, ボール})	ボール: 投手	ボールの状態: 投球されている
(catch, {捕手, ボール}) (strike, {試合}) (is-set-as, {ストライク・カウント, ストライク・カウント+1})	ボール: 投球されている	ストライク・カウント: ストライク・カウント+1 ボール: 捕手
(catch, {捕手, ボール}) (ball, {試合}) (is-set-as, {ボール・カウント, ボール・カウント+1})	ボール: 投球されている	ボール・カウント: ボール・カウント+1 ボール: 捕手
(bat, {打者, ボール}),	ボール: 投球されている	本-1 塁間: 打者 打者: 無し ボール: フライ
(bound-in, {ボール, フェールグラウンド}) (is-set-as, {ストライク・カウント, {0,1,2}})	ボール: フライ	ストライク・カウント: ストライク・カウント+1 ボール: 投手

6 おわりに

本稿では、STRIKE のインデキシングを支援するために、テロップに基づいてイベント候補を推定する手法を示した。本手法では、状況間の時間的な前後関係を定義し、時間的な前後関係を有する状況とそこで発生するイベント系列を状況遷移パターンとして表現する。テロップから観測可能な状況の変化に対して、その変化を満足する状況遷移パターンの組合せを導出することにより、テロップが不変な時区間で発生したと考えられるイベント候補を推定する。

野球中継では、テロップが不変である時区間でカット切り替えが発生することがある。野球中継中に出現するカットはカメラアングルから数種類に分類可能である。状況に応じて、出現するカットが限定可能であることからカットを自動的に分類することで、カット単位でのイベント候補の推定を行う手法を開発中である。

参考文献

- [1] A. Elmagarmid, et al.: "Video Database Systems", Kuwer Academic Publishers (1997).
- [2] E. Oomoto and K. Tanaka: "OVID: Design and Implementation of a Video-Object Database System", *IEEE Trans. on Knowledge and Data Engineering*, Vol. 5, No. 4, pp. 626-643 (1993).
- [3] 田淵仁浩, 村岡洋一: "動画データベース中の系列データを指定する条件の不完全さを許容できる問い合わせ処理と MeSOD モデル", *信学論 (D-I)*, Vol. J76-D-I, No. 6, pp. 288-299 (1993).
- [4] Y. F. Day, S. Dağtaş, M. Iino, A. Khokhar and A. Ghafo: "Object-Oriented Conceptual Modeling of Video Data", *Proc. of International Conference on Multimedia Computing and Systems*, pp. 98-105 (1995).
- [5] J. S. Li, T. Ozsu and D. Szafron: "Modeling of Video Spatial Relationships in an Object

Database Management System", *Proc. of International Workshop Multimedia DBMS*, pp. 124-133 (1996).

- [6] H. Arisawa, et al.: "Data Model and Architecture of Multimedia Database for Engineering Applications", *IEICE Trans. Inf. & Syst.*, Vol. E78-D, No. 11, pp. 1362-1368.
- [7] 牛尾剛聡, 広部一弥, 渡邊豊英: "イベント-アクションの概念に基づいた動画のシーン検索手法", 第55回情報処理学会全国大会講演論文集(3), pp. 44-45 (1997).
- [8] 美濃導彦: "知的映像メディア検索技術の動向", *人工知能学会誌*, Vol. 11, No. 1, pp. 4-9 (1996).
- [9] 館山公一, 川嶋稔夫, 青木由直: "動作スポッティングによるシーン検索", *信学技報 CVIM97-106*, pp. 115-122 (1997).
- [10] 佐藤隆他: "MPEG2 映像からのカット点とテロップの高効率検出法", *信学技報 PRMU96*, pp. 47-54 (1996).
- [11] 桑野秀豪, 倉掛正治, 小高和己: "映像データ検索のためのテロップ文字抽出法", *信学技報 PRMU96*, pp. 39-46 (1996).

