

## カラービデオ映像における自動索引付け法と物体探索法†

長坂 晃朗\*\* 田中 譲††

本論文では、ビデオデータベースを実現するための基本アルゴリズムとして、ビデオ情報の自動索引付け手法と、特定物体のフルビデオ検索法を提案する。これらは、フルテキストデータベースにおける自動索引付けと全文文字列検索に対応する検索機能をビデオデータベースに提供する。ビデオ情報の自動索引付けは、再生中のビデオ情報から実時間でカット変わりを自動検出することを基本とする。各カットの最初のフレームや最初の短時間の映像は、そのカットに特徴的な情報であり、これらの画像や動画像を表示する icon や micon のリストは、ビデオ情報の索引として用いることができる。リスト中の各 micon をマウスで指定し、対応するカットのフレーム番号を RS 232 C を介して VTR や LD に伝えることによって、そのカットを自動再生させることができる。フルビデオ検索は、検索者が指定した物体が現れているフレームをビデオ情報中からすべて探し出すことを目的とする。物体の指定は、その物体を表す画像を与えることで行う。ビデオ情報に登場する物体は、その現れる場面によって異なる大きさ、異なる形状、異なる向きを示すことが多いが、本論文で述べるアルゴリズムでは、そのような場面の物体をも見つけることができる。

## 1. はじめに

近年、マルチメディア情報の量が増大し、それらの情報の重要性が高まっている。このことは、録画テープのようなビデオ情報において特に顕著である。しかし、多くの場合、録画テープには、索引や目次といった将来の検索を手助けするいかなる情報も与えられていない。ビデオ情報の利用者が多量のビデオライブラリ中から自分の見たい場面を迅速に探し出すには、ビデオ情報のデータベース化が必要である。

ビデオ情報のデータベース化には、フルテキストデータベース (Full-text Database) の構成法が参考になる。ビデオ情報とフルテキスト情報は、ともに逐次的な情報であり、それらの情報の内容や構成に関する情報がなければ効率的な検索が行えない。しかし、両情報とも、そうした二次情報を持たない一次情報である。

フルテキストデータベースでは、検索手法として、自動索引付け (Automatic Indexing) と、全文を対象とした文字列検索 (Full-text String Search) の二つの手法が重要な役割を果たしている。ビデオ情報とフルテキスト情報の類似性を考慮すると、これらに対応する手法として、ビデオ情報に対して (1) ビデオ情報の索引付け (Automatic Video Indexing) と (2) 全ビデオ情報を対象とした物体検索 (Full-video Object

Search: 以下、フルビデオ検索と呼ぶ) の二つの処理機能を用意することが重要であると考えられる。

本稿では、ビデオ情報に対する上記二つの手法の実現方法について考察する。(1)のビデオ情報の索引付けについては、すでに著者らによってカット変わりの自動検出による索引付け手法<sup>1),2)</sup>が提案されている。カット変わりの自動検出手法は、NTTの外村<sup>3)</sup>からも発表されている。本稿では、カット変わりの検出手法について、著者らが提案した手法と、他に考えられる、外村の手法を含めたいくつかの手法とを比較し、著者らの手法の優位性を実証する。外村の手法は、2章で述べる度数分布法にあたり、著者らが提案した手法は同様に分割  $\chi^2$  検定法に対応する。(2)の物体検索については、ビデオ情報をあらかじめ人間が解析して場面の記述を与えておき、その記述を利用して物体の現れる場面を探す方法も考えられる。この実現のために、動画像中の移動体分離の自動化手法を開発し、場面記述の付与を対話的に行う提案<sup>4)</sup>がされている。しかし、場面記述の付与は、人手によって行うしかなく、大量のビデオ情報を対象にすることができない。

フルテキストデータベースにおいては、検索しようとする文や段落、文書の索引として、それらのテキスト中から抽出した単語や句のリストが用いられる。同様に、ビデオデータベースにおいては、フレームや短時間のビデオ映像を、検索する場面の索引語として使うことができる。特に、切れ目なく連続したカットと呼ばれる映像区分に映像を分割すれば、各カットのフレームは、そのカットを特徴付ける索引語となる。ビデオ情報の索引付けは、カット変わりを検出してカッ

† Automatic Video Indexing and Full-Video Search for Object Appearances by AKIO NAGASAKA and YUZURU TANAKA (Electrical Engineering Department, Faculty of Engineering, Hokkaido University).

†† 北海道大学工学部電気工学科

\* 現在 (株)日立製作所中央研究所

Now, Central Research Laboratory, Hitachi, Ltd.

トを抽出し、各カットに対して、これを代表するビデオフレームや micon (motion icon) を与える作業である。2章で述べるアルゴリズムは、このカット変わりの自動検出を行う。このアルゴリズムに基づき実現されたカット変わり自動検出システムは、再生中のビデオ情報から実時間でカット変わりを検出することができ、ズームやパンのようなカメラの動きがあってもカット変わりとは認識しない。

フルビデオ検索は、指定された物体が映っているすべてのフレームをビデオ情報中から選び出す検索手段である。3章で述べるアルゴリズムは、この処理を行う。試作した例では、探したい物体の画像のひとつを指定することにより、その物体が映っているフレームを見つけ出す。このとき、各フレームにおける物体の向きや大きさや形状は変化してもよい。

本稿の構成は、次のようになっている。2章では、計算機によるカット変わりの検出法として、いくつかの手法を比較検討する。3章では、フルビデオ検索の技法を述べ、その性能を評価する。最後に、4章でまとめを述べる。

2. 自動ビデオ索引付けのための評価関数

ここで述べるカット変わり検出アルゴリズムでは、ビデオ情報からフレームを周期的にサンプリングし、隣接するフレームのすべての対のうち、互いに著しく異なる画像特徴を持つ対を探し出す。カット変わりは、概ね、見つかったフレーム対の間にある。2枚のフレーム間の相違具合を調べるには、2枚のフレーム  $f_1, f_2$  を引数としてその相違度を数値で返す、評価関数  $E(f_1, f_2)$  を用いる。この評価関数の選定にあたっては、以下の条件を考慮する。

- (1) 実時間で計算が行える程度に簡単なこと。
- (2) ズームやパンといったカメラの動きの影響を受けづらること。

本章では、七つの異なる評価関数を考え、その中から、上の2条件を満たした、カット変わりの自動検出に最適な関数を導く。また、評価関数の7候補すべてについて、シャープ社製のパソコンX68000 (M68000, 10 MHz) とその専用フレームメモリを用いて実験を行った。サ

プリングしたフレームの有効領域は  $480 \times 480$  画素で、評価関数の計算では、そのうち間引いて  $120 \times 112$  画素を使用した。各画素は、濃淡 64 階調もしくは 64 色種をとる。1枚のフレームは、NTSC における1フレーム分の更新時間である 1/30 秒でフレームメモリに取り込まれる。評価関数のプログラムには、アセンブリ言語を使用した。

2.1 種々の評価関数とその性能の評価

実験を行った七つの評価関数とその性能比較を表1にまとめておく。表中の動き強度は、パンやズームの影響の受けづらさである。これらの評価関数は、濃淡情報のみを用いる三つの関数と、色情報を用いる四つの関数との2群に分類される。

まず面輝度法は、2枚のフレームの全体的な明るさの違いを評価値として与える。具体的には、2枚のフレームのそれぞれについて、その画像上のすべての画素の濃度レベルの総和をとり、その総和の差を相違度とする。評価式中、 $gray(f, x, y)$  はフレーム  $f$  上の  $(x, y)$  座標の画素の濃度レベルを与える関数である。この評価関数は、特にハードウェア上での実現が容易で、各フレームのビデオ信号の輝度成分を単純に積分するだけでよい。

TM法は、テンプレートマッチングの技法を用いた評価方法で、2枚のフレームのそれぞれ同じ座標の画

表1 カット変わりの自動検出を行うための評価関数7候補の比較  
Table 1 Comparison of 7 candidate evaluation functions for automatic cut-change detection.

評価方法	評価式	取扱画像	実時間処理	動き強度	評価精度
面輝度法	$ \sum_{x=1}^{120} \sum_{y=1}^{112} gray(f_1, x, y) - \sum_{x=1}^{120} \sum_{y=1}^{112} gray(f_2, x, y) $	濃淡	◎	○	×
TM法	$\sum_{x=1}^{120} \sum_{y=1}^{112}  gray(f_1, x, y) - gray(f_2, x, y) $	濃淡	◎		×
度数分布法	$\sum_{j=0}^{63}  H(f_1, j) - H(f_2, j) $	濃淡	◎	◎	
CTM法	$\sum_{x=1}^{120} \sum_{y=1}^{112} ColorDifference(f_1, f_2, x, y)$	色彩	◎		×
色度数分布法	$\sum_{j=0}^{63}  H_c(f_1, j) - H_c(f_2, j) $	色彩	◎	◎	
$\chi^2$ 検定法	$\sum_{j=0}^{63} \frac{\{H_c(f_1, j) - H_c(f_2, j)\}^2}{H_c(f_1, j)}$	色彩	◎	◎	
分割 $\chi^2$ 検定法	$SumOfMin_{8of16, r=1to16} \sum_{j=0}^{63} \frac{\{H_{cr}(f_1, r, j) - H_{cr}(f_2, r, j)\}^2}{H_{cr}(f_1, r, j)}$	色彩	◎	◎	◎

素と  
体と  
ムそ  
のヒ  
る評  
レー  
在数  
のヒ  
トグ  
C  
る。  
の濃  
Col  
ぞオ  
は、  
れ  
き、  
で  
bit

素どうしを逐一比較していき、最終的に二画像間の全体としての差異を求める。度数分布法は、2枚のフレームそれぞれに濃度ヒストグラムを求め、それら二つのヒストグラム間の相違度をフレーム間の相違度とする評価方法である。濃度ヒストグラム  $H(f, i)$  は、フレーム  $f$  において、各濃度レベル  $i$  を持つ画素の存在数をそれぞれ求めることで得られる。64階調濃度のヒストグラム  $H(f, i)$  に替えて、64色種の色ヒストグラム  $H_c(f, i)$  を用いると色度数分布法になる。

CTM 法は、TM 法を色彩画像に応用したものである。その評価式は、TM 法の評価式中における画素間の濃度差を、その色差で置き換えて得られる。関数 *ColorDifference* は、2枚のフレーム  $f_1, f_2$  のそれぞれ座標  $(x, y)$  にある画素間の色差を与える。色差は、比べる色を  $c_1, c_2, red, green, blue$  をそれぞれ RGB の各成分の濃度レベルを与える関数としたとき、

$$|red(c_1) - red(c_2)| + |green(c_1) - green(c_2)| + |blue(c_1) - blue(c_2)|$$

で定義する。この方法に限り、実験には RGB 各5 bit, 32,768色を用いる。 $\chi^2$  検定法は、二つのヒスト

グラムの比較を、度数分布法のような各度数の単純な差ではなく、カイ二乗検定を用いて行う。各度数の差  $H_c(f_1, i) - H_c(f_2, i)$  を2乗することによって、二つのヒストグラムが類似しているときとそうでないときの相違度の差が鮮明になる。 $H_c(f_2, i)$  で割ることにより、度数の値によって異なる各度数差の重みを均等化することができる。

図1に、サンプルビデオ映像に対する評価関数の値の時間変化を示す。横軸が経過時間、各カット変わりには番号を添えてある。ここに挙げたすべての評価実験で同じビデオ映像を用い、同じカット変わりには同じ番号を与えた。(a)の面輝度法では、5番と6番のカット変わり以外は検出に失敗していることがわかる。(b)のTM法と(d)のCTM法では、3番から6番にかけての区間でカット変わりの判定を困難にするほどの高い値が続いている。この区間はパンやズームが行われている場面であり、明らかにTM法とCTM法がカメラの動きに敏感すぎる事がわかる。CTM法のグラフで、TM法のときに比べ、評価値が全体的に高くなってピーク値が相対的に低くなっているのは、色差の表現が不十分だったためと思われる。

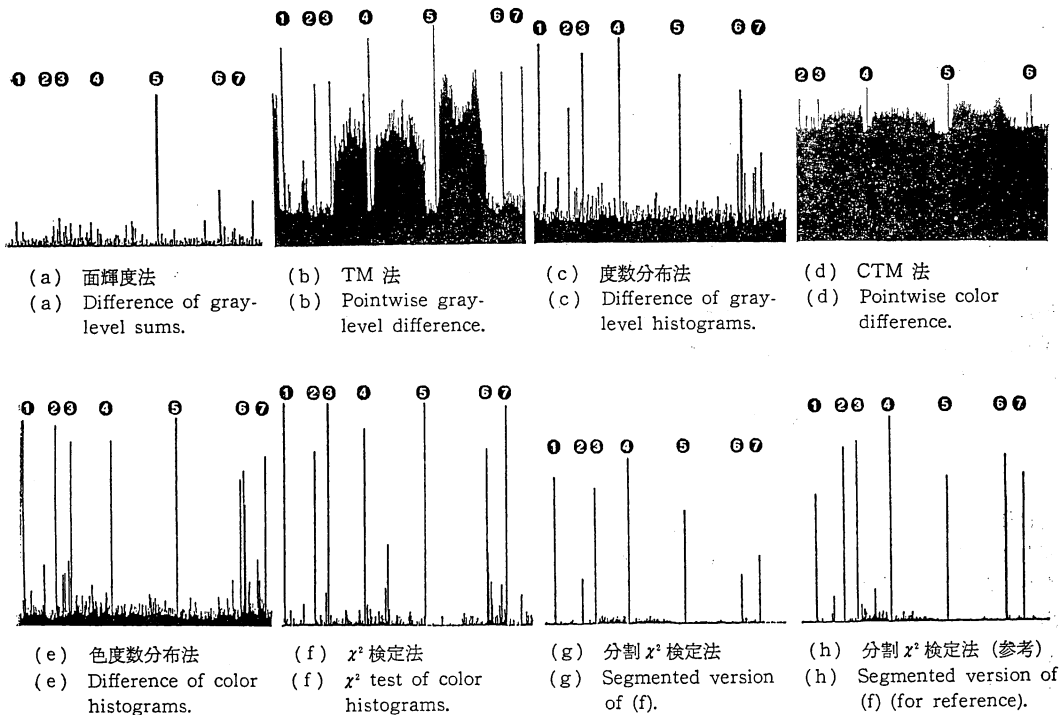


図1 カット変わりの自動検出を行う評価関数8候補による評価値の時間変化  
Fig. 1 Transitions of 8 candidate evaluation functions for automatic cut-change detection.

表 2 番組全体のカット変わり検出結果  
Table 2 Results of cut-change detection for a whole video program.

	国会中継	同 8 倍速	ドキュメント
総 カ ッ ト 数	188(60分)	188(60分)	227(44分)
正 検 出	188	188	198
誤検出	8	9	19
検出漏れ	0(0)	0(0)	29(25)

注) 括弧内は特殊効果がかかっているときの内数。

(c)の度数分布法と(e)の色度数分布法では、3番から6番のカット変わりにかけての区間でも低い評価値をとっている。色を用いたほうがカット変わりのピークが鋭く現れていることがわかる。(f)の $\chi^2$ 検定法は、(a)~(e)のいずれよりもカット変わりを示すピークがよりくっきりと現れている。

(f)の $\chi^2$ 検定法のグラフでは、カット変わり以外のところにも何本かのピークが現れている。これは、撮影のフラッシュやカメラ前の人の通過などの瞬間的なノイズの影響である。このような瞬間的なノイズが、フレームの半分以上に影響を及ぼすことは少ない。こうした部分的なノイズの影響は、以下の方法で除去できる。

まずフレームを $4 \times 4 = 16$ 個の同じ大きさの矩形領域に分割し、各領域に1から16の番号を付ける。2枚のフレームのそれぞれ同じ番号 $r$ の分割領域の色ヒストグラム $H_r(f, r, i)$ を比較し、計16個の評価値を得る。これらの評価値のうち、値の大きいものから順に八つを捨て、小さいもの八つの評価値の総和をとって(表の式の中では $SumOfMin$ と記す)最終評価値とする。これが分割 $\chi^2$ 検定法であり、この総和は明らかに部分ノイズの影響を受けづらい。この方法を用いた評価結果を(g)に示す。部分ノイズ対策の成果を示すため、16領域すべての評価値の総和を最終評価値とした実験結果を(h)に示す。(g)のグラフからは、明らかにカット変わり以外のピークが消えている。

この分割 $\chi^2$ 検定法を用い、1本の番組の始めから終わりまで連続してカット変わり検出を行った結果を表2に示す。素材には、国会中継とドキュメント番組を選んだ。国会中継では、すべてのカット変わりが検出されている。8倍速再生の映像に対しても同等の結果が得られ、索引付け時間短縮の可能性を示した。ドキュメント番組の検出率が低いのは、デゾルプやワイプといった徐々に移行するカット変わりが認識できなかったためである。これはズームやパンによる誤判定

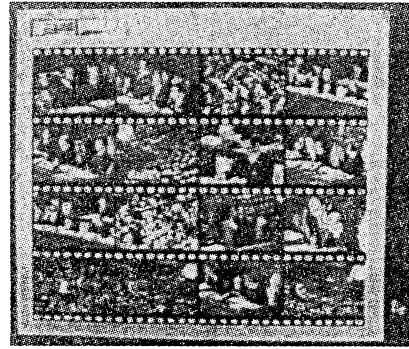


図 2 16個の micon を表示する Scene Browser  
Fig. 2 A scene browser with 16 micons.

を抑えた結果であり、今後の検討課題である。

## 2.2 Scene Browser

前述の評価関数を使うことによって、ビデオ情報をカット単位に分割することができる。各カットの始めから数秒間のビデオ映像を繰り返し再生する micon は、そのカットを代表する情報であり、一覧にして表示することによってビデオ情報の索引として利用することができる。この一覧表を Scene Browser と名付ける。図2に16個の micon を同時表示する Scene Browser を示す。各 micon は、対応するカットの録画テープ上のフレーム番号を記憶しており、任意の micon を利用者がマウスで指定することによって、フレーム番号が RS 232C を介して VTR あるいは LD に伝えられ、そのカットを自動再生させることができる。

試作した Scene Browser では、各カット間にリンクを与えることもでき、そのリンクに従ってカット間を自由に行き来することができる。Scene Browser には、インタラクティブビデオ<sup>5),6)</sup>のオーサリングツールとしての可能性もある。

## 3. 指定物体のフルビデオ検索

### 3.1 検索アルゴリズムの指針

フルビデオ検索は、検索者が指定した物体画像(参照画像)が示す物体のイメージを含んでいるすべてのフレームをビデオ情報の中から探し出すことを目的とする。2枚の画像が与えられたとき、一方の画像中に他方の画像が示す物体が含まれているかどうかの判定には、テンプレートマッチングをはじめ、いくつかの方法が知られている。しかし、フルビデオ検索で用いるには、次の二つの条件を満足する方法である必要がある。

- (1) 実時間ですべての検索処理が完了できること。
- (2) 探したい物体が、その形状や大きさや向きを変えて映っているフレームも探し出されること。

特に(2)を満たすためには、フレーム中の物体を、物体全体としての特徴で比較しながら探すのではなく、物体イメージ中の局所的な特徴の統計分布を比較することにより検索を行う必要がある。

こうした局所の特徴を用いた3種類の検索アルゴリズムについて実験を行った。局所の特徴として、それぞれ次の三つを用いた。

- (1) 参照画像の色ヒストグラム。
- (2) 参照画像に特徴的な色の集合。
- (3) 参照画像上で隣接する特徴的な色対の集合。

まず参照画像と対象のフレームをそれぞれ格子状に分割し、分割によってできた小領域(セルと呼ぶ)ごとに色ヒストグラムをとる。このとき、両画像のセルの大きさは同じにする。(1)を用いた検索手法では、フレームの各セルの色ヒストグラムを調べ、参照画像の各セルの色ヒストグラムの中に類似したものがあれば、そのセルを物体の画像の一部分とみなす。このようなセルがフレーム中に多数存在していれば、そのフレームに物体があると結論する。

(2)と(3)の検索手法とも、まずセルごとに、ヒストグラムの度数が一定数以上ある色(そのセルのセル色と呼ぶ)をすべて選ぶ。(2)では、参照画像中に最も多く現れているセル色をいくつか選び(最頻セル色)、それらの各々について、対象フレーム中でその色をセル色として持つセルの数を調べる。その数が、すべての最頻セル色で一定数以上あれば、そのフレームに物体があると判断する。(3)では、(2)のアルゴリズムにおける最頻セル色にかえて、参照画像中で隣り合うセルの双方からセル色を選んで色の対(色対)を作り、参照画像中に最も多く現れている色対の各々について、フレーム中で同様な色対を成す隣接セルの対の数を調べる。

アルゴリズム(1)は、各セルのヒストグラムが物体の拡大縮小で大きく変化するので、物体が大きさを変えて映っているフレームを検索できない。アルゴリズム(2)は、物体の変化の影響を受けづらいが、物体の色が物体を含まない画像領域に現れることも多く正確な判定ができない。アルゴリズム(3)では、色対の種類は単色の種類の数の2乗の数だけあるので、アルゴリズム(2)と比べて、物体を含まない画像領域に物体を特徴付ける色対が現れる確率が小さくなる。隣接の

方向を考慮しないので、物体の向きの変化の影響は受けない。次節では、アルゴリズム(3)の詳細を示す。

検索アルゴリズムは、次の三つのステップに分けられる。(1)前処理として、ビデオ情報の各フレームごとに、色ビットマップと呼ぶ $30 \times 30$ のビットマップを作成する。(2)参照画像から、特徴的な隣接色対を16個抽出する。(3)それら16個の色対を、色ビットマップを使ってすべてのフレームから探す。

### 3.2 ビデオ情報の前処理

まず最初に、ビデオ情報からフレームを周期的にサンプリングし、各フレームごとに以下の手順で得られる $30 \times 30$ の大きさの色ビットマップを作る。後述する検索処理には、これら色ビットマップだけをを用い、対応するフレームは一切参照しない。そのため、フレームのサンプリング時間に依存しない高速な検索が可能になる。

- (1) フレームを $30 \times 30$ 個のセルに分割する。このとき、フレームの有効領域は $480 \times 480$ 画素、各画素あたり512種類の色をとる。
- (2) フレームごとに、すべてのセルの色ヒストグラムを計算する。
- (3) 各セルにおいて、ヒストグラム中の度数が設定閾値10より大きい色をすべて抽出する。こうして選ばれた色がそのセルのセル色となる。
- (4) 512色それぞれについて、セルの並びに対応する $30 \times 30$ の大きさのビットマップを作成する。各色のビットマップは、その色をセル色に持つセルに1を立て、それ以外は0にリセットする。

### 3.3 参照画像に特徴的な色対の抽出

次に、物体の参照画像中で隣接関係を持つ色の対のうち、その物体に特徴的な色対を抽出する。その抽出アルゴリズムは次のようになる。

- (1) 参照画像を $16 \times 16$ 画素の大きさの矩形領域に分割し、その各分割領域ごとに色頻度ヒストグラムを求める。これらの領域を参照セルと呼ぶ。
- (2) 参照セルごとに、そのヒストグラム中の度数が設定閾値10より大きい色、すなわちセル色をすべて抽出する。
- (3) 物体画像上で隣り合う参照セルのすべての対について、一方の参照セルのセル色と他方の参照セルのセル色から各々一色ずつ選んで対を作る。これを、二つの参照セルの持つセル色の分だけ行い、可能性のあるすべての組合せをリストに登録する。また、各参照セルごとに、そのセル

内のセル色どうしの組合せもリストに登録する。  
 (4) 最も特徴的な色対を(3)で得られたリスト中から選択する。ここでは、参照画像中に最も多く現れている色対を、多い順に16個選ぶ。これら16個の色対を参照色対と呼ぶ。

3.4 参照色対を用いたフルビデオ検索

最後に、色ビットマップを介して、各フレーム中に参照色対が揃っているかどうかを調べる。もし、16個の参照色対のほとんどすべてが、そのフレーム上にそれぞれ一定数以上存在していれば、このフレームに物体があると結論する。そのアルゴリズムを以下に示す。

(1) 参照色対のフレーム上での分布状況を知るために、以下で定義する30×30の大きさのビットマ

ップを作成する。このビットマップにおける(i, j)座標のビットは、色ビットマップと同様、フ

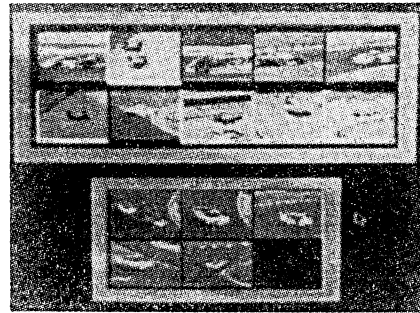
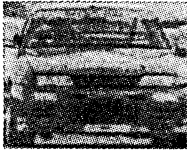
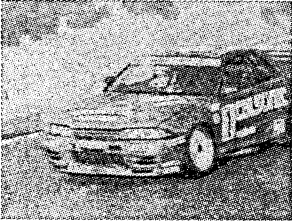


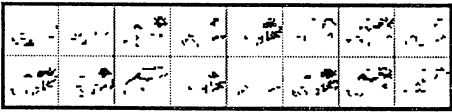
図3 フルビデオ検索の結果  
 Fig. 3 The result of a full-video search.




(a) 検索物体の参照画像



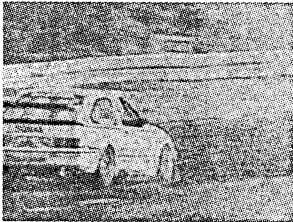
(b) 検索物体を含んだフレーム



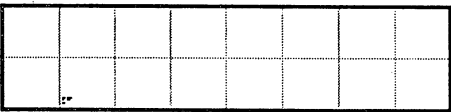
(c) (b)に対する16枚の色対ビットマップ



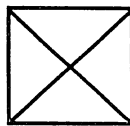
(d) (b)に対する検索結果ビットマップ



(e) 別の物体を含んだフレーム



(f) (e)に対する16枚の色対ビットマップ



(e)に対する検索結果ビットマップは生成されない。

図4 検索対象物体の参照画像と、その物体の出現検出

Fig. 4 A sample image of a search object and the searching of two frames for this object.  
 (a) An image of a car specified as a search object, (b) A frame containing the car, (c) 16 color-pair bitmaps for (b), (d) A search result bitmap for (b), (e) A frame containing a different car, (f) 16 color-pair bitmaps for (e).

レームにおける同じ位置のセルに対応する。もし、あるセルが色対を構成する2色のうちのいずれかをセル色として持ち、色対の他方の色を、そのセル自身、あるいはその隣接する8セルのうちのいずれかのセルがセル色として持っていれば、そのセルに対応するビットマップのビットをセットする。

こうして得られたビットマップを色対ビットマップと呼ぶ。参照色対の数だけ作られる16枚の色対ビットマップのうち、セットされたビットの数が8個に満たないものは、そのセットビットが検索物体の画像とは無関係に現れたノイズである可能性が高いので無効にする。このとき、有効なビットマップの数が14枚以上であれば(2)に進む。そうでなければ検索対象物体はこのフレームには現れていないと判断し、次のフレームの処理に進む。

- (2) 有効な色対ビットマップすべての論理和をとり、検索結果ビットマップと呼ぶ1枚の30×30ビットマップを作成する。このビットマップ上でのセットビット領域は、このフレームにおける検索対象物体のおおよその出現位置を与える。

### 3.5 試作例とその評価

フルビデオ検索アルゴリズムを、先述のX68000とその専用フレームメモリからなるシステムを使って試作した。上記のアルゴリズム説明の中で記した各設定閾値は、このシステムで最も優れた検索性能が得られたときの実験値である。

フルビデオ検索の実行例を図3に示す。上側のウィンドウが、10個のmiconを表示するScene Browserである。検索物体を指定するには、まず検索物体が表示されているmiconをマウスでクリックし、そのmiconを静止画表示状態にしてから、そのmicon画像中の検索対象物体をマウスを使って矩形枠で囲む。この矩形領域を参照画像として、参照色対を抽出して検索を行う。検索結果は、下側のウィンドウに表示される。検索対象物体が登場しているカットのmiconが列挙されている。

図4には、参照画像と2枚のフレームを示す。各フレームごとに16枚の色対ビットマップと検索結果ビットマップとを示す。左の画像が見つけるべき車を含んでいるフレームで、右が別の車を含んでいるフレームである。右側の例では、セットされたビットを持つ色対ビットマップの数が1個しかなく、本アルゴリ

表3 番組全体の物体検索結果

Table 3 Results of a full-video search of some video programs.

	A	B	C	D
総出現フレーム数	52	81	44	49
正 検 出	50	75	44	49
誤検出 過剰検出	9	93	68	38
検出漏れ	2	6	0	0

注) 各番組とも全20分、2秒ごと全600frameを検索。

ムは、このフレームには検索対象物体が存在しないと判断する。

表3に、実際のTV番組を使ってフルビデオ検索を行った結果を示す。AとBはクイズ番組で二人の司会者をそれぞれ検索した例であり、CとDはレース中継で車を検索物にした例である。いずれも不必要な検出は多いが、検索物体の現れているフレームを高い確率で探し出している。不要な検出が特に多いBは、黒無地のTシャツを着用した男性を検索物体とした例であり、黒も肌色も検索物体と無関係に現れることが多いために物体を一意に特定できなかったと思われる。また、検出漏れは、同一の物体でも異なる光源下では異なる色を示すことから生じている。色温度補正を用いて光源に依存しない色を定義するなどの措置が必要である。

## 4. おわりに

本稿では、ビデオ情報に対する自動索引付けのアルゴリズムを提案し、カットごとに取り出したサンプル映像をmiconにして表示するScene Browserの試作例を示した。また、指定した物体が登場しているフレームをビデオ情報からすべて検索する、フルビデオ検索手法を提案した。両アルゴリズムとも、専用ハードウェアを用いずに、パーソナルコンピュータを使って実時間処理することが可能であり、しかもカメラの動き等による画像変化やノイズの影響を受けにくい。問題点としては、色情報だけを用いて処理を行うため、過剰に検出することが多い点がある。また、徐々に移行するカット変わりの検出と、参照物体と異なる光源下の物体の検索が現在のところできていない。今後は、実時間処理性能を損なわずに、これらの問題点の解決を目指していくつもりである。

## 参 考 文 献

- 1) 長坂, 田中: ビデオ作品の場面変わりの自動検出法, 第40回情報処理学会全国大会論文集, pp.

642-643 (1990).

- 2) 長坂, 田中: ビデオ映像のカット変わり検出法, 画像ラボ, Vol. 1, No. 9, pp. 35-37, 日本工業出版 (1990).
- 3) Tonomura, Y.: Video Handling Based on Structured Information for Hypermedia Systems, *International Conference on Multimedia Information Systems '91*, pp. 333-344, McGraw-Hill, Singapore (1991).
- 4) 上田: インタラクティブな動画像編集方式の提案, 電子情報通信学会技術報告, IE 90-6 (1990).
- 5) Mackay, W.E. and Davenport, G.: Virtual Video Editing in Interactive Multimedia Applications, *Comm. ACM*, Vol. 32, No. 7, pp. 802-810 (1989).
- 6) Ripley, G.: DVI-A Digital Multimedia Technology, *Comm. ACM*, Vol. 32, No. 7, pp. 811-822 (1989).

(平成3年4月1日受付)

(平成4年1月17日採録)



長坂 晃朗 (正会員)

昭和42年生。平成元年北海道大学工学部電気工学科卒業。平成3年同大学院修士課程修了。同年(株)日立製作所中央研究所入社。大学院在学時より動画像処理の研究に従事。



田中 譲 (正会員)

昭和25年生。昭和47年京都大学電気工学科卒業。昭和49年京都大学電子工学専攻修士課程修了。工学博士。現在、北海道大学電気工学科教授。データベースマシン、データベース理論、メディア・ベース、論理型プログラミング等の研究に従事。主たる著書、「コンピュータ・アーキテクチャ」(オーム社、共著)。IEEE、ソフトウェア科学会、人工知能学会各会員。