

マクロブロックタイプを用いた MPEG2 圧縮動画像のカット点検出

青木真吾 森田啓義 荒俣吉壮

morita@is.uec.ac.jp
電気通信大学大学院情報システム学研究科
〒182-8585 調布市調布ヶ丘 1-5-1

あらまし 画像信号全体を復号することなく MPEG2 データから直接、瞬時ならびにディゾルブショットを検出する方法を提案する。提案方法は、MPEG データの I フレーム情報を利用した GOP 単位での粗い検出と、マクロブロックタイプを用いたフレーム単位での正確な検出という二つの手法を階層的に行うことにより、従来法に比べて、高い検出率と短い検出時間を可能にする。いくつかのビデオデータを用いた検出実験結果より、瞬時カット点検出では、通常の再生時間の 4 分の 1 程度の処理時間でもって、固定したしきい値を用いて 90~95 % 近い検出率を達成し、誤検出率も 5 % 程度と低く抑えられた。さらに、ディゾルブカット点検出に関しては、提案法は実験に用いた全てのデータに対して 80 % 以上の検出率を得、また検出性能は従来法より平均 20% 程度向上した。

キーワード カット点検出, MPEG2, マクロブロックタイプ

Cut Detection in MPEG2 Compressed Data Based on Macro Block Types

Shingo Aoki Hiroyoshi Morita Yoshitake Aramata

Grad. School of Inform. Systems, University of Electro-Communications
Chofugaoka 1-5-1, Chofu, Tokyo 182-8585

Abstract We propose a method to detect both of instantaneous and dissolved shots from MPEG2 data without decoding the entire volume of video signals. The proposed method consists of two detection algorithms in a hierarchical fashion. One algorithm coarsely detects shots in GOP unit by extracting I-frame information from the MPEG data stream. The other identifies the exact boundary between shots in frame unit using detailed information on types of macro blocks contained in the data stream. The proposed method perform faster and more efficient than traditional ones. Experimental results on some video data show that our method achieved 90 ~ 95% detection rate with about quoter of normal playback time for detecting instantaneous shots. Moreover, it suppressed the erroneous detection rate within 5%. As for dissolved shots, the proposed method attained more than 80% detection rate over all the video used in the experiment and the mean of detection ability is about 20% higher than the traditional ones.

Keywords Cut Detection, MPEG2, Macro Block Type

1 はじめに

近年、CPUの高速化、ハードディスクの大容量化が急速に進んで、高性能パーソナルコンピュータが廉価化するにつれ、一般ユーザが映像データを利用するハードやソフトが広く普及してきている。しかし、さまざまな映像データが大量に蓄積されてくると、映像データの必要な部分だけを取り出してつなぎ合わせるといった編集作業や、莫大な映像情報の中から必要なデータを効率良く選択し、その情報に素早くアクセスするビデオ検索技術が重要になる。

本稿では、ショット単位やシーン単位で映像にアクセスしたり、シーンの要約などのより高度な機能を実現することを目的に、通常の映像データに現れるカット点の大半を占める、瞬間的に映像の切り替わる瞬時カット点と、スポーツ映像で頻繁に用いられる、二つのショットが合成されながら切り替わるディゾルブカット点を、MPEG2圧縮データから自動的に検出する手法について考察する。

MPEG2(あるいはMPEG1)で圧縮された映像データからカット点を検出する方法はすでに様々な方法が提案されている[3]-[10]。それらの中でも、MPEG圧縮データ中のIフレームに含まれる輝度DC成分を2次元に配列した、いわゆる、DC画像を利用する方法[3]-[5],[8]は、完全に映像データを復元するための処理時間を省くことができ、しかも瞬時カット点検出の場合、一旦復元してからカット点を検出する方法[2]とほとんど変わらない精度で検出できるという点で注目に値する。反面、Iフレーム情報だけではカット点の正確なフレーム位置を同定するには不十分であること、映像に応じて検出に用いるパラメータをよく調整しないと、誤検出や未検出といった誤りが生じやすいという問題がある。

ディゾルブカット点検出に関しては、フレームごとの画素値の分散がディゾルブ部分においてU字形に変化するという特徴を利用して検出を行う方法[8,9]や、輝度値の差分を利用する方法[10]が知られているが、文献[5]で指摘されている通り、実際の映像では動きのあるシーンや局所的な変動によって理想的なU字型になることが少ないため未検出や誤検出が多く発生する恐れがある。

これらの問題は、MPEG符号によって圧縮されたデータから必要な情報だけを抽出することによって、映像データによっては解消することができる。一般に、映像データにカット点が生じると、

MPEG圧縮データに埋め込まれたマクロブロックタイプ情報には、MPEG符号器の動作に起因する特徴的なパターンが現れる。この特徴パターンを統計的に処理することにより、カット点の検出は画像再生にかかる時間を大幅に短縮することがができるだけでなく、高い検出率でカット点を検出できることが報告されている[4,6]。しかし一方で、映像に応じてパラメータをよく調整しないと、物体やカメラワークが急激に変化するシーンにおける誤検出や暗いシーンにおける未検出といった誤りが生じやすく、ディゾルブカット点検出では、検出性能は60%~70%前後であることが報告されている[6]。

本稿では、従来法のようにマクロブロック情報をBフレーム一枚ごとに処理するのではなく、隣接する2枚のBフレームのマクロブロック情報とカット点の関係に着目し、誤検出・未検出といった誤りの改善を図る。さらに、ディゾルブカット点検出に関しては、動きベクトル情報をマクロブロック情報に付加することによって、検出誤りが改善できることを示す。

以下、次節では、本稿の議論に最低限必要なMPEG2技術について述べる。第3節では、マクロブロック・タイプ情報を用いたカット点検出法を提案する。続いて第4節では、従来法と提案法の性能比較実験を行う。最後に、まとめを第5節で行う。

2 カット点とMBTとの対応関係について

MPEG2は、主にフレーム内符号化、予測符号化、双方向予測符号化という3種類の符号化手法から構成されており[1]、ここでは、便宜上、順に、I符号、P符号、B符号と呼ぶ。

これらの符号に対応して、映像データの各フレーム(1画面)には、Iフレーム、Pフレーム、Bフレームという3種類がある。どのフレームがどの種類になるかは周期的に決まる。1周期分のフレームはGOP(Group of Picture)と呼ばれ、市販のエンコーダなどでは、1GOPは12~15個のフレームから構成されることが多い。

I符号はフレーム内の情報のみを使用して符号化を行うが、P符号では、直前のIフレームもしくはPフレームを予測値として参照することによって、あらかじめ冗長な部分を取り除き、高い圧縮効果を得ている。B符号は、P符号と同様の予測を行うが、直前と直後のIフレームもしくはPフレームを参照することができるので、P符号より

さらに圧縮効果が期待できる。

ただし、B符号を用いる場合、表示順のままフレームを符号化すると、復号側で直後フレームの参照が実現できなくなるので、符号化順は図1に示すように並び換えられている。

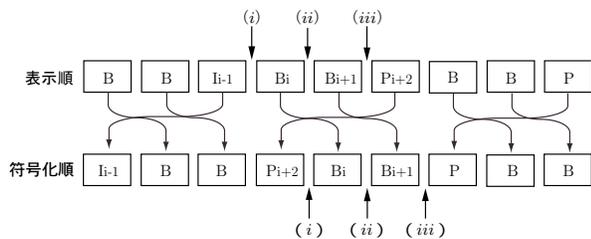


図 1: フレームの表示順と符号化順の違い

いずれの符号を用いる場合においても、基本的にはフレームをさらに細分化した 8×8 のブロック単位に離散コサイン変換が施され、変換係数の量子化・エントロピー符号化によってデータが圧縮される。それぞれのブロックにどの符号を適用するかは、縦・横 2×2 のブロックを一つにまとめたマクロブロック (MBと略す) を単位にして (図2参照)、符号化の際にもっとも圧縮効果が高くなるものが選ばれる。

なお、四つのブロックは輝度情報のみを表しており、MBにはそれらの色情報を表すために他に二つのブロックが含まれている。

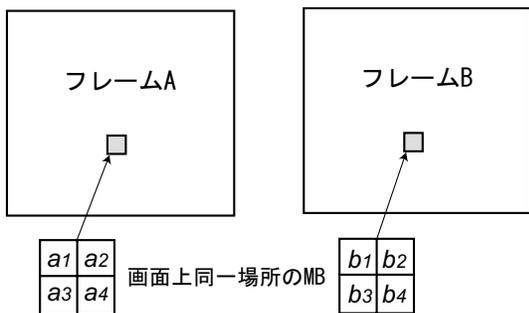


図 2: マクロブロックとブロックの関係

符号化について注意すべきことは、フレームと符号が1対1に対応しているのではなく、表1に示すように、フレームの種類によってマクロブロックごとに選択可能な符号が異なっている点である。したがって、例えば、Bフレームに含まれるブロックは、MPEG符号器において、I符号、P符号、B符号で同時に符号化される。正確に言うと、P符号を用いる際には、直前のIフレームもしくはPフ

レームを参照する順方向予測と、直後のそれを参照する逆方向予測がありうるので、合わせて、これら4種類の符号の中からもっとも圧縮効果が高いものが選ばれる。同様に、Pフレームのブロックに関しては、I符号と順方向予測のP符号から選択される。

各ブロックをどの符号を用いて符号化したかという情報はMB単位に符号化の際に圧縮データに埋め込まれている。この情報をマクロブロックタイプ (MBTと略す) と呼ぶ。

映像データにカット点が発生すると、符号器では予測がうまく働かず、PフレームやBフレームに対しても、I符号を選択する傾向が強くなる。圧縮データからMBTを抽出するのは比較的容易なので、MBTの分布をフレームごとに調べればカット点の発生を検出できると思われる。これが、本提案方式を含めMBTを用いるカット点検出法に共通する基本的な考え方である。

実際、瞬時カット点とディゾルブカット点が発生する場合に分けて、MBTの分布がどのようになるかについて以下で考察する。なお、本研究では、GOPの構成は、Iフレーム1枚、Pフレーム4枚、Bフレーム10枚、計15枚がIBBPBPBPBPBPBBの順で並んだ標準的なものに固定している。

2.1 瞬時カット点が発生する場合

瞬時カット点の発生する位置は1対の連続したBフレーム B_i, B_{i+1} に注目すると、以下の三つの場合に分類できる。

- (i) B_i の直前にカット点が存在する場合
- (ii) B_i と B_{i+1} の間にカット点が存在する場合
- (iii) B_{i+1} の直後にカット点が存在する場合

ここで、表示順、符号化順のいずれにおいても、 B_i の左隣のフレームと B_{i+1} の右隣のフレームはIもしくはPフレームであるが、どの場合でも扱いは同じなので、以下では、図1に示した表示順

表 1: 各フレームと適用可能な符号の関係

フレーム種類	I符号	P符号	B符号
Iフレーム			
Pフレーム			
Bフレーム			

(印は適用可能)

にフレームが $I_{i-1}, B_i, B_{i+1}, P_{i+2}$ と並んだ場合を例にとって説明する。

(i) では, B 符号を用いて B_i と B_{i+1} の各ブロックを符号化する際には, I_{i-1} と P_{i+2} が参照される。この場合, 映像は I_{i-1} と B_i の間で切り替わるので, I_{i-1} を参照しても, 予測がうまく働かず, P_{i+2} のみへの参照を符号器は採択する傾向にある。すなわち, B_i, B_{i+1} 共に逆方向予測を行うマクロブロック (BMB と略す) の数が増える (図 3)。(ii) の

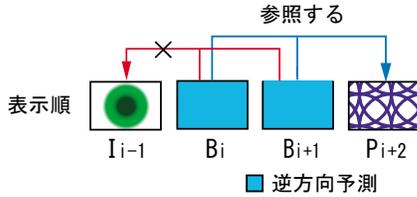


図 3: (i) の場合

場合は, B_i と B_{i+1} の間で映像が切り替わるため, 符号器では, B_i に対しては I_{i-1} への順方向予測を行うマクロブロック (FMB と略す), B_{i+1} に対しては P_{i+2} への BMB の数が増加すると考えられる (図 4)。(iii) の場合は, B_i, B_{i+1} は共に, I_{i+2} への FMB

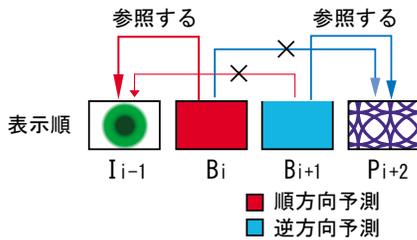


図 4: (ii) の場合

の数が大半を占める (図 5)。

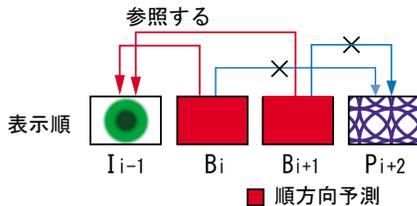


図 5: (iii) の場合

以上の考察より, B フレーム対の FMB と BMB 数の大小から, GOP 中のどこで瞬時カットが発生

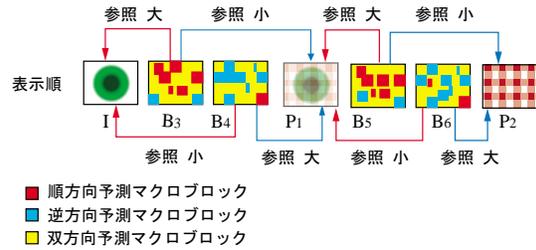


図 6: ディゾルブカットと MBT の関係

したかを検出できる,

2.2 ディゾルブカット点が発生する場合

画像が徐々に変化していくディゾルブカット部分では, MBT の分布は瞬時カットの場合とは異なる特徴をもつ変化をする。図 6 はディゾルブカット部分での MBT 分布の変化を例示したものである。図中の B_3 に注目すると, 参照できるフレームは, 左隣の I と二つ先にある P_1 であるが, I のほうが P_1 より時間的に近いので B_3 との相関が強く, 予測結果が P_1 を用いた場合に比べより正確に行えると考えられる。その結果, BMB に比べ, FMB の数が増える傾向にある。

一方, B_4 では, I より P_1 との相関が強いので, I への順方向予測より, P_1 への逆方向予測のほうが頻繁に行われるため, BMB が多く発生すると考えられる。

同様に, 対 (B_5, B_6) については, B_5 では P_1 への FMB が増え, また, B_6 では相関の強い P_2 への BMT が P_1 への FMT に比べてより多く現れる。

以上まとめると, ディゾルブ部分では B_3, B_5 で FMB が多く, B_4, B_6 で BMB が多くなると考えられる。すなわち, B フレームごとに FMB と BMB の頻度を比べると, その大小関係は, ディゾルブ変化が生じている部分において, フレーム単位で交互に反転するという現象 (ジグザグ現象とよぶ) が生じうる。この現象は, 文献 [6] の方法 (JYL 法) で始めて報告された。

3 提案カット点検出法

提案方式では, 最初に, I フレームの DC 画像情報を用いて GOP 単位でカット点を粗く検出し, 続いて, MBT を利用してフレーム単位で正確なカット点位置を検出するという階層的な手法を用いることによって, 検出処理を効率良く高速化している。最初の GOP 単位の検出と, 続くフレーム単位での検出には, 瞬時カット点とディゾルブカット点の場

合と異なる手法を用いているので、以下では二つに分けて説明する。

3.1 瞬時カット点検出

3.1.1 I法 [3, 5]

隣接する GOP それぞれの先頭に位置する二つの I フレームに含まれる輝度 DC 画像が大きく変化すればそれらの間でカット点が生じていると考えられる。輝度 DC 画像の変化は次のようにして定量的に評価できる。

まず、あるしきい値 T_δ と T_ν をあらかじめ固定しておき、図 2 中のフレーム A と B を、隣接する GOP にそれぞれ含まれる I フレーム I_A, I_B として考える。

さらに、それぞれのフレームにおいて画面上で同じ位置を占める MB 中のブロック $a_1 \sim a_4$ と $b_1 \sim b_4$ に対して離散コサイン変換を施した後の輝度 DC 成分をそれぞれ α_i, β_i ($i = 1, \dots, 4$) とおく。これらの差の絶対値の和

$$\delta_I = |\alpha_1 - \beta_1| + |\alpha_2 - \beta_2| + |\alpha_3 - \beta_3| + |\alpha_4 - \beta_4|$$

に対し、不等式

$$\delta_I > T_\delta$$

が成り立つ MB の数を ν_I とする。もし

$$\nu_I > T_\nu$$

ならば、両 I フレーム間のいずれかのフレームでカット点が発生したと判断する。

3.1.2 MBT 法

I 法でカット点を含むと判断された GOP に対して、MBT の分布を用いてフレーム単位での検出を行う。第 2.1 節で考察した (i) ~ (iii) の場合に対応して、B フレーム対 B_i, B_{i+1} の MBT から FMB の数 f_j と BMB の数 b_j ($j = i, i+1$) があるしきい値 T_μ を越えるかどうかを表 2 の判定条件と照合し、三つの場合のうちのどれが生じたかを判断する。

しきい値 T_μ の決め方は、画面の明るさと映像の動きの激しさに関係する。画面が暗いと、カット点が生じているフレームでも双方向予測を行う MB が半数近く残る場合があり、その結果、表 2 のどの条件にも該当しないという未検出誤りが発生しやすい。

表 2: 瞬時カット点の判定ルール

判定条件	判定結果
$\min\{b_i, b_{i+1}\} \geq T_\mu$	(i)
$f_i \geq T_\mu \wedge b_{i+1} \geq T_\mu$	(ii)
$\min\{f_i, f_{i+1}\} \geq T_\mu$	(iii)

提案法では、I 法で算出したブロックに対する δ_I の値をフレームごとに平均をとり、あらかじめ定めた一定値に乘じることによって、 T_μ の値を適応的に変化させ、未検出を減らしている。

しかし、この方法だけでは、画面が暗くかつ動きが激しいシーンでは、同じ GOP 内に判定条件をみたす B フレーム対がいくつも現れ、誤検出が発生する可能性がある。そこで、GOP 中の各 B フレーム対 (B_i, B_{i+1}) に対して、

$$\Delta = |f_i - b_i| + |f_{i+1} - b_{i+1}| \quad (1)$$

を計算し、最大値 Δ_{\max} をとる B フレーム対を求める。他のフレームのとり値と Δ_{\max} の比が一定のしきい値 T_Δ 以下であるとき、カット点が発生したと判断する (図 7 参照)。

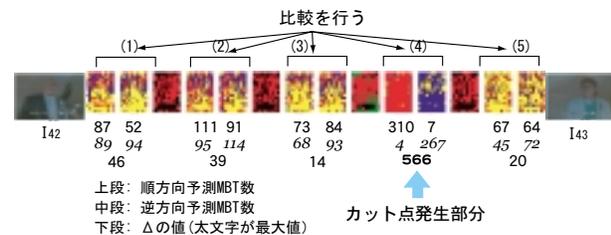


図 7: カット点発生部分の例

3.2 ディゾルブカット点検出

ディゾルブカット点ではジグザグ現象が発生するが、この現象は、動きの激しいシーンにおいても生じうる。ただし、ディゾルブカット点の大半は継続時間が 1GOP 程度なので、以下で述べる kI 法で、数 GOP に及ぶ動きの激しいシーン部分を除外し、つぎの修正 BYL 法で短時間のものを省くことで誤検出を抑える。

3.2.1 kI 法

ディゾルブが継続する間隔を覆う k 枚の I フレームに対して、輝度 (Y)、色差 (Cb)、色相 (Cr) それ

それぞれについて、フレーム間のヒストグラムの変化量(変動距離, ダイバージェンス)を求め、その結果からディゾルブカット点の検出を行う。まず、成分 $x = Y, Cb, Cr$ に対し、 j 番目の I フレーム ($j = 1, \dots, k$) の各 DC 成分値を均等に q 値に量子化する(実験では $q = 16$ に固定した)。このとき i 番目の量子化区間に値が落ちる相対頻度を $p_j^x(i)$ とおくと、変動距離 V_j^x とダイバージェンス D_j^x は以下の式で与えられる。

$$V_j^x = \sum_{i=1}^q |p_j^x(i) - p_{j+1}^x(i)|$$

$$D_j^x = \sum_{i=1}^q p_{j+1}^x(i) \log \frac{p_{j+1}^x(i)}{p_j^x(i)}.$$

これらを 3 成分全体で平均をとった値をそれぞれ V_j, D_j とおく。つぎに、

$$r_j = V_j D_j$$

を計算する。ディゾルブ分では、 k 枚の I フレームの中間部では、二つのシーンが重なるため r_i の値が際立って大きくなり、一方、シーンの変化が少ない両端部では、 r_i の値は極端に小さくなると考えられる。

ところで、 k のとるべき値はディゾルブの継続する時間によって変化するが、今回対象にした映像データではほとんどが 1 秒未満で切り替わっているので、 $k = 6$ として考え、

$$r_3 \geq T_1$$

が成り立ち、さらに、

$$(r_1 \leq T_2 \wedge r_2 \leq T_2) \vee (r_4 \leq T_2 \wedge r_5 \leq T_2)$$

が成り立つならば、中心位置を占める 3 番目の I フレームから始まる GOP 部分でディゾルブが発生したと判断する。

3.2.2 修正 BYL 法

kI 法でディゾルブがあると判断されたならば、さらにフレーム単位の検出を行いディゾルブであるかどうかを、2 節で述べたようにジグザグ現象が起きるかどうかで判断する。

BYL 法 [5] では B フレーム内の FMB 数 f と BMB 数 b から順方向マクロブロック比 ρ を

$$\rho = \frac{f}{f+b}$$

として定義し、 ρ の値の変化からディゾルブを検出している。

しかし、ディゾルブ変化は上下左右に物体が動くような変化ではなく、画像が浮き出てくるような変化なので、動きベクトルの大きさは比較的小さいと考えられる。

そこで本稿では、圧縮データから各 MB の動きベクトル情報を取り出し、ベクトルの大きさがしきい値 T_m より大きな MB はあらかじめ f と b の計算から除外した上で、順方向マクロブロック比 ρ' を求める修正 BYL 法を提案する。

さらに修正 BYL 法では、B フレームの対 (B_i, B_{i+1}) ごとに、 ρ'_i, ρ'_{i+1} を計算したら、B フレームの対ごとに

$$\Delta\rho = \rho'_{i+1} - \rho'_i$$

を計算し、しきい値 T_ρ より大きい $\Delta\rho$ の値が連続するならば、ディゾルブが発生したと判断する。

4 カット点検出実験

瞬時カット点検出、ディゾルブカット点検出それぞれについて、提案法の有効性を調べるために、従来法との比較実験を行った。従来法としては、瞬時カット点検出については、提案法と同様に MBT を利用している MJC 法 [4]、ディゾルブカット点検出に関しては、MBT 比のジグザグ現象の利用する JYL 法 [6] を用い、各方式のプログラムはすべて JAVA で書いた。

実験に使用した計算機は、CPU が Athlon (1.4GHz)、搭載メモリーは 512MB の Windows 2000 Pro が稼働する PC で、市販の MPEG2 エンコーダ基板を PCI 接続している。MPEG 2 圧縮データはすべて市販ビデオから作成し、解像度は 352×240 、ビットレートは平均 2Mbps、最大 4Mbps の VBR (Variable Bit Rate) である。

カット点検出の結果を評価するため、検出率 (RCD)、誤検出率 (RFD)、検出性能 (DP) の三つの基準を用いた。これらの値は、既知正解カット点数 (CSC)、正しく検出されたカット点数 (CD)、誤検出カット点数 (FD) から、

$$RCD = \frac{CD}{CSC} \times 100$$

$$RFD = \frac{FD}{CD + FD} \times 100$$

$$DP = \frac{CD - FD}{CSC} \times 100$$

として求められる。

実験結果については、瞬時カット点とディゾルブカット点で異なる映像データを用いているため以下で分けて述べる。

4.1 瞬時カット点検出実験結果

瞬時カット点検出では実験データとして表3に掲げるものを用いた。それぞれのデータのカット

表3: 瞬時カット点検出に用いた映像データ

映像の種類	時間	カット点数
(11) ドラマ	10分	204個
(12) サッカー	10分	78個
(13) ホラー	10分	113個
(14) アニメ1	60分	866個
(15) アクション映画	105分	2020個
(16) サスペンス映画	100分	746個
(17) ラブストーリー	113分	1229個
(18) アニメ映画	133分	1662個

点位置は、あらかじめ目視により正確な位置を確かめた。しきい値は提案法では全てのデータで固定した値を用いたが、MJC法では各データごとに検出性能が最もよくなる値を選んだ。

検出率、誤検出率、検出性能の結果を表4に示す。表中の太文字はより良い方の値を示している。

提案法とMJC法との比較すると、検出率では双方ともに90%以上の良好な結果が得られた。しかし、映像(I5)、(I8)に対しては、提案法は従来法と比べて検出率が低減している。

この原因は、二つの映像では、動きの激しいシーン中に瞬時カット点が発生している場面があり、提案したMBT法では、GOP内に複数のBフレーム対で Δ/Δ_{max} が T_{Δ} を越えるため、カット点とは判断されないことによる。

誤検出率は従来法に比べ提案法は大幅に改善されている。映像(I6)、(I7)はいずれも動きの少ないシーンで構成されているが、一般に、動きの少ない映像に対しては、MBT法は誤検出率を低く抑えるのに有効である。動きの激しい映像である(I2)、(I4)、(I5)、(I8)に対しても、MJC法に比べ誤検出率は1/2から1/3程度に減少している。

検出性能に関しては全ての映像データで提案法はMJC法を上回り、ほとんどのデータで90%を超える結果を得ている。動きの激しい(I4)、(I8)では検出率が低いため、検出性能が80%台に留まっている。動きの激しい映像でのカット点検出率を

より向上させることが今後の課題である。

検出時間に関しては提案法、MJC法共にデータの再生時間の約4分の1で検出することができた。今回の実験ではプログラム作成にJAVAを使用した。C言語に翻訳すれば、検出時間はさらに短縮できると考えられる。

4.2 ディゾルブカット点検出実験結果

ディゾルブカット点検出実験に用いた映像データを表5に掲げる。瞬時カット点に比べるとディゾルブカット点は使用される頻度が低く、データ数も必然的に限られてくる。そのため、比較的ディゾルブを多く使用しているスポーツの映像を中心にTV放送から採取した。50分間の全データ中、111個のディゾルブカット点が含まれている。

しきい値に関しては、JYL法、提案法ともに各データごとに検出性能の値が最もよくなるものを選び検出を行った。検出率、誤検出率、検出性能の結果を表6に示す。検出率では提案法は80%以

表5: ディゾルブカット点検出に用いた映像データ

映像の種類	時間	カット点数
(1) サッカー1	10分	18個
(2) サッカー2	10分	9個
(3) サッカー3	10分	57個
(4) 野球	10分	15個
(5) ドキュメント	10分	12個

上の結果を得ることができ、JYL法より良い結果を得ている。未検出となったものには、類似シーン間でのディゾルブが多く見られた。しかし異なるシーン間での場合などは正確な位置で検出されていることからディゾルブカット点検出に有効である。

誤検出率に関しては、大きな動きベクトルをもつMBを数えないという修正BYL法の考え方が有効に働いている。しかし、(D2)では逆に誤検出率がBYL法に比べて増加している。実際に誤検出が生じたのはスロー再生のシーンであり、動きベクトルの大きさは小さいため、新たな工夫が必要と思われる。この点に関しては今後の課題とする。なお、検出性能では、ほとんどの映像で80%近い結果を得ており、従来法より平均で20%程度向上している。

表 4: 瞬時カット点検出結果の比較 (上段: MJC 法/下段: 提案法)

(百分率表示)

映像データ	(I1)	(I2)	(I3)	(I4)	(I5)	(I6)	(I7)	(I8)
検出率 (RCD)	100.	92.31	95.58	93.76	93.37	95.58	96.50	93.86
	100.	96.15	99.12	91.68	94.01	99.06	97.97	89.71
誤検出率 (RFD)	0.48	10.00	3.57	10.96	9.24	3.52	5.49	11.86
	0.00	3.36	0.88	5.47	3.36	0.81	0.41	6.87
検出性能 (DP)	99.51	82.05	92.04	82.22	83.86	92.01	90.90	81.22
	100.	92.31	98.23	86.37	90.74	98.26	97.55	83.37

表 6: ディゾルブカット点検出結果の比較 (上段: JYL 法/下段: 提案法)

(百分率表示)

映像データ	(D1)	(D2)	(D3)	(D4)	(D5)
検出率 (RCD)	78.78	66.67	75.44	80.00	66.67
	100.	100.	84.21	80.00	83.33
誤検出率 (RFD)	22.22	0.00	10.41	7.69	38.46
	10.00	10.00	9.43	0.00	23.08
検出性能 (DP)	56.56	66.67	66.67	73.33	25.00
	88.89	88.89	75.44	80.00	58.33

5 まとめ

MPEG2 圧縮データから映像の瞬時カット点ならびにディゾルブカット点検出において、動きの激しいシーンにおける誤検出、暗いシーンにおける未検出という二つの誤りを改善するために、B フレーム対のマクロブロックタイプを利用する提案法の有効性が確かめられた。

今後の課題としては、動きのあるシーンなどでの誤検出を防ぐとともに、同系色でのディゾルブディゾルブカット点の検出精度を上げることに、さらに、他の漸次カット点が生じた場合と MBT の関係について考察することが挙げられる。

最後に、MPEG2 符号化方式には、圧縮率を上げるためにさまざまな画像処理技術が用いられている。これらを圧縮データに含まれている情報からの確に引き出すことによって、今後、カット点検出を含め、さまざまな応用の可能性が期待できる。

参考文献

[1] ISO/IEC 13818-2 Committee Draft (MPEG-2)
 [2] 大辻清太, 外村圭伸, 大庭有二, “突発的フィルタを用いた映像カット点検出法”, D-II, 信学論, vol. J77-D-II, No.3, pp.519-528, 1994.

[3] Y.Nakajima, “A Video Browsing Using fast scene change detection for an efficient networked video database access,” *IEICE Trans. Inf. & Sys.*, vol.E77-D.no.12, pp.1355-1364, Dec 1994.
 [4] Jianhao Meng, Yujen Juan, Shih-Fu Chang, “Scene Change Detection in a MPEG Compressed Video Sequence,” *Digital video Compression Algorithms and Technologies*, SPIE, vol. 2419, pp. 14-25, 1995.
 [5] 中島康之, 氏原清乃, 米山暁夫, “部分複合を用いた MPEG データからのカット点検出”, D-II, 信学論, vol.J81-D-II, No.7, pp.1564-1575, 1998.
 [6] Sung-Bae Jun, Kyoungro Yoon, Hee-Youn Lee, “Dissolve Transition Detection Algorithm Using Spatio-Temporal Distribution of MPEG Macro-Block Types,” ACM, 2000.
 [7] 新倉康巨, 阿久津明人, 谷口行信, 佐藤隆, 外村佳伸, 浜田洋, “MPEG 符号化映像ショットチェンジ検出のための動き補償解析ハイブリット法の提案”, D-II, 信学論, vol.J81-D-II, No.8, pp.1838-1846, 1998 年 8 月.
 [8] B.L.Yao and B.Liu, “Rapid scene analysis on compressed video,” *IEEE Trans. Circuits and Systems for Video Technology*, Dec. 1995.
 [9] A.M.Alattar, “Detecting and compressing dissolve regions in video sequences with a DVI multimedia imaging compression algorithm,” *Proc. IEEE Int. Sym. on Circuits and Systems*, pp.13-16, May 1993.
 [10] L. Gu, K. Tsui and D. Keightley, “Dissolve Detection in MPEG Compressed Video,” *IEEE Int. Conf. on Intelligent Processing Systems*, Beijing, China, Oct. 1997.