

尤度比検定による MPEG 圧縮動画像からのカット検出

金子 敏充

堀 修

(株) 東芝 研究開発センター 情報・通信システム研究所

あらまし本稿では、MPEG により圧縮されている動画像を対象に、動画像解析の最も基本的な処理であるカット検出を尤度比検定を用いて行う手法を提案する。ここでは MPEG ビットストリームを復号せずに、直接得られる情報を利用するアプローチをとっており、一度復号してから処理を行うカット検出手法よりも処理速度が速いという特徴がある。利用可能な様々な種類の情報の中から、特に動きベクトル情報に着目した場合の尤度関数を具体的に求め、様々なジャンルからなる映像データに対してカット検出実験を行った結果、従来のようなフレーム内のマクロブロックの予測方式の割合だけを利用したカット検出手法よりも検出性能が改善できることが示された。

キーワード: ビデオインデクシング, シーンチェンジ, 尤度比検定, 動きベクトル

Cut Detection Method for MPEG Compressed Video using Likelihood Ratio Test

Toshimitsu KANEKO and Osamu Hori

R&D Center, Toshiba Corporation

Abstract A cut detection technique using likelihood ratio test for MPEG compressed video sequences is proposed. Automatic cut detection techniques are essential techniques to analyze video contents. Previous techniques define similarities of two successive frames and determines the cut points by comparing similarities with a threshold value. But the threshold values are determined heuristically. The new technique estimates likelihood functions from video samples and determine cut points by test the hypothesis that two frames belong to a same shot. This technique make it possible that the optimum threshold value is calculated analytically. The experimental results show that the new technique gives smallest probability of false negative for fixed probability of false positive in comparison with the previous techniques.

keywords: video indexing, scene change, likelihood ratio test, motion vector

1 まえがき

動画像において1台のカメラで連続的に撮影された映像区間をショットと呼び、ショットのつなぎ目をカットと呼んでいる。カットを検出する手法は、動画像の構造解析における最も基本的な処理として研究されてきた。

これまで研究は、どれも2フレーム間の類似度、または非類似度を求め、しきい値との比較によりカットを検出するというものである。2フレーム間の対応する画素ごとの輝度の差分を基にした手法[1]-[4]や、フレームごとの輝度または色のヒストグラムを基にした手法[5][6]などが良く知られている。

このようなアプローチは、どのような形式で表現された動画像に対しても画素値に変換してから適用できるため、応用範囲が広いという利点がある。しかし、動画像はほとんどの場合情報圧縮された形で扱われる。圧縮された動画像データからカット検出を行う場合には、復号を行わずに直接処理するほうが処理速度の点で有利である。実際、以前筆者らが開発した手法[7]では7~8倍の高速化を達成した。

最近では、動画像の圧縮方式としてMPEGに注目し、この方式に特化したカット検出を行う手法の研究が行われるようになった。このような研究として、IピクチャのDCT係数の直流成分のように簡易復号が可能な画像だけを部分的に復号し、従来のような画素値を用いたカット検出法に帰着させる手法が提案されている[8][9]。しかしこの手法では、Iピクチャは通常数フレームから十数フレームに1度の割合でしか含まれないため、カットの大まかな位置しかわからないこと、時間解像度が悪いので動きの激しいシーンでの誤検出が増えることなどの問題点がある。また、圧縮後のデータサイズを非類似度とする手法[10]も提案されているが、データサイズは符号化時の発生符号量制御の影響を受けやすいという問題がある。そこで、Iピクチャ単位でのカット検出よりも時間解像度及び安定度の高いカット検出をするための手法も提案されている。それらは近似的にBピクチャやPピクチャのDCT直流成分の画像を再構成し、あくまでも画素情報を利用したカット検出をするアプローチ[11][12]と、MPEGから直接得られる情報を利用してカット検出を行おうとす

るアプローチ[13]-[17]の2つに分類することができる。MPEGから直接得られる情報を利用したものほとんどは、フレーム内で用いられているマクロブロックの予測方式の割合をフレーム間類似度とする方法である。具体的には、Pピクチャの順方向予測符号化マクロブロックの割合を用いたもの[13]、Bピクチャの内挿的予測符号化マクロブロックの割合を用いたもの[14]、順方向と逆方向予測符号化マクロブロックの数の差を用いたもの[15]、そして、順方向・逆方向・内挿的予測符号化のマクロブロック数を組み合わせたもの[16][17]などが提案されている。

本稿では観測データとして差分動きベクトルを用いる。差分動きベクトルはカットがない場合には0付近の値をとり、カットがある場合には大きな値になるもしくは定義されないといったカット検出のための情報を含んでいる。また、MPEGによる圧縮時の発生符号量制御の影響を受けにくいため、安定した性質も持っている。様々なジャンルからなる映像データを用いた実験により、実際に差分動きベクトルを使ったカット検出法が高い検出精度を持つことを示す。

また、本稿では差分動きベクトルを観測したときのカットの判定基準として尤度比検定を用いる。従来は観測データに対してアドホックに決められたしきい値によりカットの検出がなされていた。提案する手法では、データから尤度を見積もり、尤度比による検定を行うことで過剰検出率一定のもとで未検出率を最小にするカット検出が行える。また統計的に意味付けられたしきい値の設定が可能になる。

2 尤度比検定によるカット検出手法

2.1 対象とするMPEG構造

MPEGは画像間の相関を利用してデータの圧縮を行っている。従って、相関が利用されている2つのフレーム間の類似度・非類似度は圧縮データから直接取り出すことが可能である。しかしながら、必ずしも任意の2フレームの間の類似度・非類似度を求められるとは限らない。

2つの時刻 t_a と t_b ($t_a < t_b$) に対し、時刻 t'_a ($t'_a \leq t_a$) のフレームを参照画像として画像間予測符号化された時刻 t'_b ($t'_b \geq t_b$) のフレームが存在する

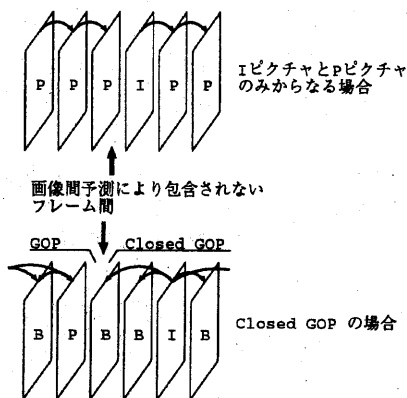


図 1: 画像間予測により包含されない画像間が存在する例。上は I ピクチャと P ピクチャのみからなるビットストリーム、下は Closed GOP の場合。フレームからフレームへの矢印は予測関係を表す。

か、時刻 t_b' ($t_b' \geq t_b$) のフレームを参照画像として画像間予測符号化された時刻 t_a' ($t_a' \leq t_a$) のフレームが存在するとき、時刻 t_a と t_b のフレーム間は画像間予測により包含されているということにする。あらゆる場所に存在するカットを検出するためには、全ての連続した 2 つのフレーム間について、このフレーム間が画像間予測により包含されているという条件が必要である。例えば closed GOP を含んでいたり、I ピクチャと P ピクチャとだけからなる圧縮データでは、画像間予測によって包含されないフレーム間が生じてしまう (図 1)。このような場合には、画像に復号してから類似度・非類似度を算出する以外にカットの見逃しを防ぐ手段はない。

ここでは、このような画像間予測により包含されないフレーム間がないような符号化が行われているか、もしくは画像間予測により包含されていないフレーム間のカットは検出対象としないものとする。なお、画像間予測により包含されていないフレーム間の対策については 4.4 で触れる。

2.2 対数尤度比検定

カット検出対象となる動画の各フレームには、表示順に時刻が対応づけられているものとする。フレームレートが R のとき、時刻の単位は $1/R$ である。このときに、時刻 $t_0 - 1$ と時刻 t_0 との間

のカット検出を統計的検定を用いて行うことを考える。未知な母数 θ は

θ_0 : 時刻 $t_0 - 1$ と時刻 t_0 との間にカットがない,

θ_1 : 時刻 $t_0 - 1$ と時刻 t_0 との間にカットがある

となる。また、検定すべき仮説は、

帰無仮説 H_0 : $\theta = \theta_0$,

対立仮説 H_1 : $\theta = \theta_1$

である。すなわち、連続する 2 つのフレームが同一のショットに含まれているという仮説を検定し、仮説が棄却されたときにカットありと判定する。この検定を対数尤度比検定で行う場合には、 θ に関する尤度関数 $L(\theta)$ および定数 c を用いて、

$$\log \frac{L(\theta_1)}{L(\theta_0)} > c \quad (1)$$

を棄却域とする。尤度関数は $L(\theta) = p(\mathbf{x}|\theta)$ により与えられる。ここで、 $p(\mathbf{x}|\theta)$ は θ が真の時にデータ $\mathbf{x}$ を観測する確率である。

次に効率的にカットの検出を行うための観測データ $\mathbf{x}$ をどのように選ぶかを考える。遅延時間および処理の簡便さを考慮すると、できるだけ短いフレーム区間のデータでカットの検出が行えることが望ましい。そこで $\mathbf{x}$ を、時刻 t_0 と $t_0 - 1$ の間のカットの有無が大きな影響を与えるフレーム区間からの情報に限定する。カット検出の対象となっている時刻 t_0 に対し、二つの時刻 f_{t_0} と b_{t_0} を

$$f_{t_0} = \max\{t | t < t_0, \text{pct}(t) \neq B\}, \quad (2)$$

$$b_{t_0} = \min\{t | t \geq t_0, \text{pct}(t) \neq B\} \quad (3)$$

と定義する。ここで、 $\text{pct}(t)$ は時刻 t のピクチャタイプであり、I ピクチャのとき I、P ピクチャのとき P、B ピクチャのとき B と表し、 $\text{pct}(t) = \{I, P, B\}$ とする。このとき、MPEG の画像間予測の構造 (図 2) から、圧縮の際に時刻 $t_0 - 1$ と時刻 t_0 の間のカットの有無の影響を強く受けるのは時刻 f_{t_0} のフレームから時刻 b_{t_0} のフレームまでである (観測するデータによっては、DCT 係数の AC 成分などのように発生符号量制御が原因で時刻 b_{t_0} 以降のフレームも影響を受ける)。よって発生符号量制御の影響を受けにくいデータを観測データ $\mathbf{x}$ とすれば、時刻 f_{t_0} から b_{t_0} までのフレームの符号化後のデータからカットの検出が行える。

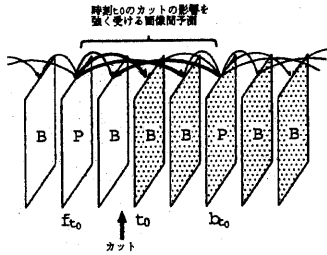


図 2: カットの有無の影響を強く受けるフレーム区間。時刻 t_0 でカットが生じているとき、カットをまたがった画像間予測を太い矢印で表している。時刻 f_{t_0} から b_{t_0} までのフレームが強く影響を受ける。

2.3 動きベクトル情報を用いた対数尤度比検定

時刻 t_0 のカットの有無の影響が時刻 f_{t_0} から b_{t_0} までのフレーム内に限定される情報としては、動きベクトルや予測誤差の DCT 直流成分などがある。符号化された動きベクトルデータは全体に占めるデータ量が少ないので、発生符号量制御の影響を受けにくい。また、カットを含まない画像間予測では動きベクトルは周囲の動きベクトルに似ているが、カットを含む画像間の予測では動きベクトルは不揃いになったり存在しなくなるなど性質が異なるため、カット検出のための有効な情報となる。そこで、ここでは動きベクトル情報を観測データとした場合の尤度関数を考える。

実際に観測するのは、動きベクトルのベクトルそのものでもよいし、MPEG の動きベクトル算出のもととなる差分動きベクトル、motion_code、ハフマン符号などとしてもよい。動きベクトルをもとに算出されたこれらのデータにはカット検出のための情報が含まれている。ここでは、これらの動きベクトルがもととなって算出されたデータのうちカット検出のために観測するデータを動きベクトル情報と呼ぶ。

時刻 t のフレームに含まれる動きベクトル情報を $m(t)$ と表記することにする。符号化の際には、 $t \neq t'$ のとき $m(t)$ と $m(t')$ は独立に求められて

いると考えても良いので、尤度関数 $L(\theta)$ は、

$$L(\theta) = \begin{cases} \prod_{t=f_{t_0}+1}^{b_{t_0}-1} p_t(m(t)|\theta) & : \text{pct}(b_{t_0}) = I, \\ \prod_{t=f_{t_0}}^{b_{t_0}} p_t(m(t)|\theta) & : \text{pct}(b_{t_0}) \neq I \end{cases} \quad (4)$$

として与えられる。ここで、 $p_t(m(t)|\theta)$ は時刻 t において θ が真のときにデータ $m(t)$ を観測する確率である。よって、 H_0 の棄却域はそれぞれ

$$\sum_{t=f_{t_0}+1}^{b_{t_0}-1} \log \frac{p_t(m(t)|\theta_1)}{p_t(m(t)|\theta_0)} > c : \text{pct}(b_{t_0}) = I, \quad (5)$$

$$\sum_{t=f_{t_0}}^{b_{t_0}} \log \frac{p_t(m(t)|\theta_1)}{p_t(m(t)|\theta_0)} > c : \text{pct}(b_{t_0}) \neq I \quad (6)$$

となる。

時刻 t のフレームが B ピクチャであるときには、 $m(t)$ は順方向予測に使われる動きベクトルと、逆方向予測に使われる動きベクトルとからなる。また、時刻 t のフレームが P ピクチャである場合には、 $m(t)$ は順方向予測の動きベクトルだけからなり、I ピクチャの場合には通常動きベクトルはない。また、B ピクチャ、P ピクチャにおいて、一般に時刻が離れた参照画像を使ったときほど予測誤差や動きベクトルが大きくなる傾向があるため、 $m(t)$ の分布は参照画像との時刻差に依存する。従って、 $p_t(m(t)|\theta)$ をピクチャタイプと参照画像との時刻差との関数と見なして、順方向予測の参照画像との時刻差が j である P ピクチャの動きベクトル情報の分布を $p_t(m(t)|\theta) = p_{(j)}^P(m(t)|\theta)$ 、また、順方向予測、逆方向予測の参照画像との時刻差がそれぞれ j, k である B ピクチャの動きベクトル情報の分布を $p_t(m(t)|\theta) = p_{(j,k)}^B(m(t)|\theta)$ のように分けて表記しておく。ここで、 $f_{t_0} < t_0 \leq t$ のときの $p_{(j,k)}^B(m(t)|\theta)$ と $t < t_0 \leq b_{t_0}$ のときの $p_{(j,k)}^B(m(t)|\theta)$ とは異なっていることに注意する。

次に、動きベクトル情報 $m(t)$ として、動きベクトルの符号化時に用いられる差分動きベクトルに限定した場合について考える。すなわち、MPEG では実際に符号化されるのは差分動きベクトルであつて、差分動きベクトルもしくは差分動きベクトルをもとにして算出された motion_code やハフマン符号などを観測データとした場合を考える。隣のマクロブロックの動きベクトルとの差分を取ることによって冗長な情報が十分取り除かれたと仮定すると、それぞれの差分動きベクトルは独立に分布

しているため、フレーム全体での差分動きベクトル情報の生起確率は以下のようにマクロブロックごとの差分動きベクトル情報の生起確率の積に分解することができる。

$$p_{(j)}^P(m(t)|\theta) = \prod_{i=0}^{n-1} p_{(j)}^P(m_i(t)|\theta), \quad (7)$$

$$p_{(j,k)}^B(m(t)|\theta) = \prod_{i=0}^{n-1} p_{(j,k)}^B(m_i(t)|\theta). \quad (8)$$

m_i はマクロブロック i の差分動きベクトル情報であり、 n は1フレーム中のマクロブロックの数である。よって仮説 H_0 の棄却域は $pct(b_{t_0}) = 1$ のとき、(5) 式より

$$\sum_{t=f_{t_0}+1}^{b_{t_0}-1} \sum_{i=0}^{n-1} \log \frac{p_{(t-f_{t_0}, b_{t_0}-t)}^B(m_i(t)|\theta_1)}{p_{(t-f_{t_0}, b_{t_0}-t)}^B(m_i(t)|\theta_0)} > c, \quad (9)$$

$pct(b_{t_0}) \neq 1$ のとき (6) 式より

$$\begin{aligned} & \sum_{t=f_{t_0}+1}^{b_{t_0}-1} \sum_{i=0}^{n-1} \log \frac{p_{(t-f_{t_0}, b_{t_0}-t)}^B(m_i(t)|\theta_1)}{p_{(t-f_{t_0}, b_{t_0}-t)}^B(m_i(t)|\theta_0)} \\ & + \sum_{i=0}^{n-1} \log \frac{p_{(b_{t_0}-f_{t_0})}^P(m_i(b_{t_0})|\theta_1)}{p_{(b_{t_0}-f_{t_0})}^P(m_i(b_{t_0})|\theta_0)} > c \end{aligned} \quad (10)$$

となる。ただし、Bピクチャは時刻 f_{t_0} のフレームと時刻 b_{t_0} のフレームを参照画像として予測符号化されるので、Bピクチャの画像間予測は時刻 $f_{t_0} < t \leq b_{t_0}$ の全てのフレーム間を包含する。従って $f_{t_0} < t \leq b_{t_0}$ の区間にBピクチャが含まれているときには、Pピクチャの情報を使わない (9) 式で仮説検定を行うことが可能である。

3 実験および考察

3.1 実験データ

提案した手法の性能を確かめるため、実際のビデオ映像を用いてカット検出実験を行った。実験に用いた映像素材は、ニュース2種類80分、映画6種類200分、ドラマ1種類30分、音楽(コンサート)60分、アニメーション30分、カーレース20分、紀行80分の合計約500分の映像である。これらをそれぞれのジャンルごとに2つに分け、データA、データBとした。データAには1852個のカット、データBには1889個のカットが含まれている。

これらの映像素材は、画像サイズ 720×480 (マクロブロック数1350)、符号化レート4Mbps~6Mbpsの固定ビットレート、GOP構成は $N = 15, M = 3$ でBBIBBPBBPBBPBBPという典型的なピクチャ構成としてMPEG2符号化した。

3.2 差分動きベクトルデータの対数尤度比

次に、差分動きベクトル情報を用いた対数尤度比検定を行うため、データAから m_i の生起確率分布を推定する。差分動きベクトルをそのまま用いず、参照画像とのフレーム間隔で正規化し、さらに x, y 成分語とに0~16の整数値に丸めた mc_1 (順方向予測動きベクトル)、 mc_2 (逆方向予測動きベクトル) を観測データとした。順方向予測符号化のとき $mc_2 = 17$ 、逆方向予測符号化のとき $mc_1 = 17$ 、イントラ符号化のとき $(mc_1, mc_2) = (17, 17)$ と表記することになると、 $m_i = (mc_1, mc_2)$ 、 $0 \leq mc_1 \leq 17, 0 \leq mc_2 \leq 17$ である。

$M = 3$ の構造を持つMPEGデータを用いているので、Bピクチャの順方向および逆方向の画像間予測により全てのフレーム間が包含できる。よって、Bピクチャの差分動きベクトル情報のみによる (9) 式で検定が可能である。Bピクチャの順方向予測と逆方向予測の方法に違いはないので、この対象性を利用すると推定する確率分布は $p_{(1,2)}^B((mc_1, mc_2)|\theta_0)$ と、 $t_0 = f_{t_0} + 1$ すなわち順方向予測がカットを挟んでいるときと $t_0 > f_{t_0} + 1$ すなわち逆方向予測がカットを挟んでいるときの $p_{(1,2)}^B((mc_1, mc_2)|\theta_1)$ の合計3種類である。

データAをサンプルとして推定した対数尤度比

$$\log \frac{p_{(1,2)}^B((mc_1, mc_2)|\theta_0)}{p_{(1,2)}^B((mc_1, mc_2)|\theta_1)} \quad (11)$$

の分布の例を図3に示す。これらを用いて、(9) 式によってカットの有無の検定を行うことができる。なお、フレームピクチャにおいてマクロブロックがフィールド予測を用いている場合、使われている動きベクトルの数はフレーム予測を用いている場合の2倍になってしまう。このようなことを考慮して、生起確率を求めるときにはフレームピクチャにおけるフィールド予測の差分動きベクトル情報 (mc_1, mc_2) をトップフィールド、ボトムフィールドの両方求め、それぞれの重みを $1/2$ とした。

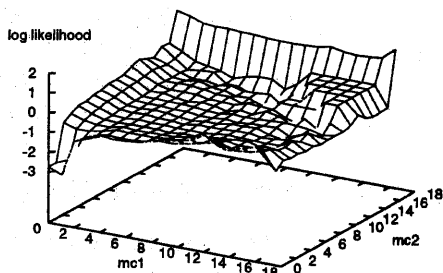


図 3: データ A から推定した対数尤度比の値 ($t_0 > f_{t_0} + 1$ のとき).

3.3 カット検出手法の性能

カット検出の検出性能の評価は、未検出 (false negative) 率および過剰検出 (false positive) 率により評価することとする。未検出率は未検出カット数/正カット数 $\times 100$ 、また、過剰検出率は過剰検出カット数/正カット数 $\times 100$ により算出する。

3.3.1 カット検出に適した観測データ

対数尤度比検定に用いる観測データとして、発生符号量制御の影響を受けにくい差分動きベクトルと DCT 直流成分が考えられるので、これら 2 つの観測データを用いた対数尤度比検定によるカット検出の実験結果について述べる。

DCT 直流成分を用いた尤度比の分布は DCT 直流成分の絶対値とマクロブロックの予測方式によりクラスに分け、差分動きベクトル情報を利用した場合と同様にデータ A からそれぞれのクラスの対数尤度比を推定した。

データ B に対して差分動きベクトル情報および DCT 直流成分を観測データとして対数尤度比検定によりカット検出を行った結果を図 4 に示す。どちらの場合にも、しきい値 α を変化させることにより未検出率、過剰検出率の関係を求めた。図 4 より、差分動きベクトル情報を用いたときとの検出性能がずっと良いことがわかる。両者の検出能力の差を調べたところ、暗い場面間のカット検出能力の違いがほとんどであった。図 5 はその一例であり、差分動きベクトルを用いた #51 と #101

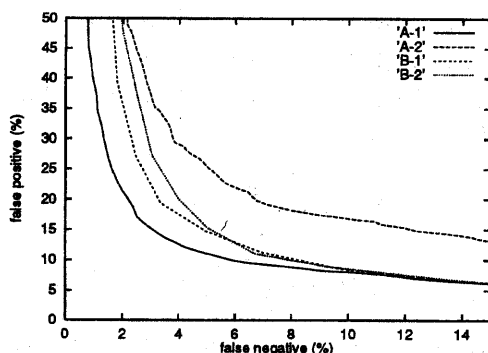


図 4: データ B を用いてカットの未検出率・過剰検出率を求めた実験結果。A-1 が差分動きベクトル情報を用いた対数尤度比検定、A-2 が予測誤差を用いた対数尤度比検定、B-1 が予測符号化マクロブロックの割合、B-2 が内装の予測符号化マクロブロックの割合によるカット検出。

のカットの両方が正しく検出されるが、予測誤差の DCT 直流成分を用いた場合には暗い場面間のカット (#101) でのピークが小さく、検出されにくいことを示している。

3.3.2 従来手法との比較

次に対数尤度比検定を用いたカット検出手法と従来提案されているカット検出手法の検出性能の比較を行う。従来提案されている手法として、フレーム内の内挿の予想符号化マクロブロックの割合、予測符号化マクロブロック (順方向予測符号化、逆方向予測符号化、内挿の予測符号化マクロブロックの総和) の割合を用いたカット検出手法の実験を同じデータを用いて行った。

これらのマクロブロックの割合に対しても、対等な条件で良い特徴量を見つけることができるように固定しきい値を用いてカットを検出することとした。検出に用いる固定しきい値の変化させることにより未検出率、過剰検出率の関係を求めた結果を図 4 に示した。図 4 から、差分動きベクトル情報を利用した尤度比検定法が最も優れた検出性能を示していることがわかる。

尤度の推定が正しく行われていれば、尤度比検定は過剰検出率一定のもとで未検出率を最小にす

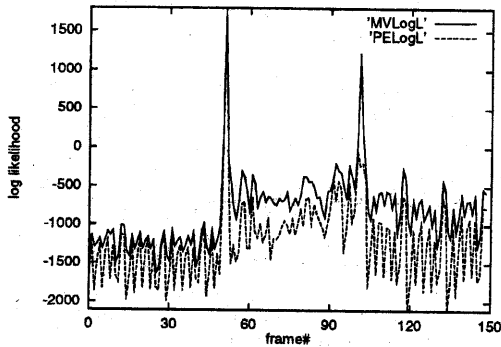


図 5: フレームごとの対数尤度比の例. MVLogL は差分動きベクトル, PELogL は予測誤差 DCT 直流成分を観測データとした場合. フレーム #51 と #101 にカットがある. PELogL は MVLogL に比べて暗いシーン間のカット (#101) に弱い.

る判定基準である. また, 従来はしきい値 c の値はヒューリスティックに決められていたが, 尤度比検定に基づく提案手法においては未検出と過剰検出に生じるコストを決めることにより, 一意に c を決定することができる. このように統計的意味付けが行える点が従来の手法との大きな違いである.

3.3.3 検出失敗例

差分動きベクトル情報を用いた対数尤度比検定で未検出となった場面は, 暗いシーンのカットがほとんどであった. 暗い場面での未検出を防ぐには, 観測データに画像全体の輝度を反映させるデータを加え, 明るさを尤度に反映させるか, もしくはこれを簡易的に行うために画像全体の輝度により検出に用いるしきい値 c をダイナミックに変化させる必要がある.

また, 逆に過剰検出となったシーンには, フラッシュの場面, 人や車がカメラの前を横切る場面, 急なパンのある場面などがあった. これらのうちフラッシュの場面と人等がカメラの前を横切る場面の過剰検出は, 3.4 で述べる方法のように時間解像度を荒くしたカット検出手法との併用で容易に防ぐことが可能である.

3.4 MPEG による符号化時のパラメータが与える影響について

ここでは, MPEG 符号化時のパラメータが差分動きベクトル情報を用いた尤度比検定に与える影響について考察する.

● 符号化レート

符号化レートは予測画像と符号化画像の差分画像の量子化の粗さに大きな影響を与える. 画像間予測符号化の使用の有無, 最適な画像間符号化方式, さらに, 動き補償に伴う動きベクトルはどれが最適かといったことは, 符号化レートを変えても変化することはない. よって, 差分動きベクトルを用いてカットの有無を検定する手法は, 符号化レートの変化の影響を受けない. 映画の中の 5 分間のシーンを 2 つの異なるビットレートで符号化した際の (mc_1, mc_2) の分布を比較した実験では, 両者の分布の差は小さく, 分布はビットレートに対して影響を受けにくいことが示された.

● 画像サイズ

画像サイズが変わると, 動きベクトルの絶対的な大きさも変化する. よって差分動きベクトルの生起確率も変化する. 同一の映像を異なるサイズ (704×480 , 352×480) で符号化した際の (mc_1, mc_2) の分布の比較実験を行った結果, 両者の分布が大きく異なっていた. 符号化画像サイズにより尤度比分布を学習し直すことが必要である.

● GOP のピクチャ構成

GOP のピクチャ数 N による影響は受けないが, P ピクチャと P ピクチャまたは P ピクチャと I ピクチャの間隔 M が大きくなると, 推定する必要のある生起確率分布の数が増加する. MPEG ビットストリーム中に含まれる画像間予測時の参照画像との時刻差 j, k 全てに対して $p_{(j)}^P(m_i(t)|\theta)$ および $p_{(j,k)}^B(m_i(t)|\theta)$ の分布をあらかじめ推定しておかなければならない.

● 画像間予測により包含されない画像間が存在する場合

このようなケースに対応するためには, 他の

カット検出手法との併用が必要になる。特に好ましい手法としては、IピクチャのDCT直流成分を利用したカット検出手法[9][18]がある。これら手法は単独で用いると検出率および時間解像度の点で問題があるが、本稿で提案した手法と組み合わせて用いるといくつかの利点がある。その一つは画像間予測によりカバーされない画像間のカットの検出が行えるということである。そのほか、フラッシュなどの2,3フレームだけの画像の変化によるカットの誤検出を訂正できること、および大まかにカットのありそうなGOPをIピクチャだけを用いて選択しておくことにより、処理速度をさらに向上できる[18]。

4 むすび

本稿では、MPEG から直接得ることのできる情報を利用して対数尤度比検定を行うことにより、カット検出を行う手法の提案を行った。この手法はMPEGビットストリームの復号を行わないために高速に処理が行えるという利点がある。また、MPEGビットストリームに含まれる情報として差分動きベクトル情報を利用した場合について尤度関数を求めた。実際に映像を使って実験を行った結果、従来のマクロブロックの種類を計数する手法よりも検出性能の向上が図れることが示された。

参考文献

- [1] D.C.Coll and G.K.Choma, "Image Activity Characteristics in Broadcast Television," IEEE Trans. Commun., vol.26, pp.1201-1206, 1976.
- [2] 大辻清太, 外村佳伸, 大庭有二, "輝度情報を使った動画ブラウジング," 信学技報, IE90-103, 1990.
- [3] 清水哲, 畑中良夫, 辻正和, "映像カット変化点検出機," テレビジョン学会全国大会, 9-4, 1988.
- [4] 大辻清太, 外村佳伸, 大庭有二, "画素変化面積測定による動画カット検出," 信学会秋季大会, D-210, 1991.
- [5] 外村佳伸, 安部伸治, "動画画像データベースハンドリングに関する検討," 信学技報, IE89-33, 1989.
- [6] 長坂晃朗, 田中謙, "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," 情処論, vol.22, No.4, pp.543-550, April, 1992.
- [7] 金子敏充, 堀修, "動きベクトル符号量を用いたMPEG 動画像からの高速カット検出," 信学技報, PRMU96-100, 1996.
- [8] 中島康之, "圧縮動画画像ファイルからの高速カット検出," 信学会秋期大会, D-265, 1993.
- [9] I.K.Sethi and N.Patel, "A Statistical Approach to Scene Change Detection," Storage and Retrieval for Image and Video Database III, SPIE vol.2420, pp.329-338, 1995.
- [10] 畑亮太, 赤間俊和, 藤本仁, 志水郁二, "圧縮動画画像のデータサイズを用いたカット検出法," 信学会総合大会, D-398, 1995.
- [11] B.L.Yeo and B.Liu, "Rapid Scene Analysis on Compressed Video," IEEE Trans. Circuit and Syst. for Video Tech., vol.5, pp.533-544, 1995.
- [12] S.F.Chang and D.G.Messerschmitt, "Manipulation and Compositing of MC-DCT Compressed Video," IEEE J. Select. Areas Commun., vol.13, pp.1-11, 1995.
- [13] 新倉康巨, 谷口行信, 阿久津明人, 浜田洋, "MPEG 2 映像からのカット点検出法," 信学会情報・システムソサエティ大会, D-272, 1996.
- [14] 山下賢二, 金子邦彦, 牧之内顕文, "双方向予測マクロブロックの率を利用したMPEGビデオカット検出法の実装・評価・改良," 第53回情処全大, 5N-4, 1996.
- [15] 椎尾一郎, "予測ブロック数を利用したMPEG動画画像カット検出法," 第52回情処全大, 2G-8, 1996.
- [16] J.Meng, Y.Juan and S.F.Chang, "Scene Change Detection in a MPEG Compressed Video Sequence," Algorithms and Technologies, SPIE vol.2419, pp.14-25, 1995.
- [17] H.C.Liu and G.L.Zick, "Scene Decomposition of MPEG Compressed Video," Algorithms and Technologies, SPIE vol.2419, pp.26-37, 1995.
- [18] H.J.Zhang, C.Y.Low and S.W.Smoliar, "Video Parsing and Browsing Using Compressed Data," Multimedia Tools and Applications, vol.1, no.1, pp.89-111, 1995.