

ROI を用いた J P E G 2 0 0 0 動画配信圧縮方法

朝井 公教^I 瀧ヶ平 将行^{II} 一條 健司^{III} 吉岡 良雄^{III}

^I 弘前大学理工学研究科

^{II} (株)メイテック

^{III} 弘前大学理工学部

電子情報システム工学専攻

LSI デザインセンター東京グループ

電子情報システム工学科

1. はじめに

現在ネットワーク上での動画配信が始まっているが、一般には各家庭においてネットワークの帯域幅が異なる。そのため、帯域幅に合わせ複数のファイルを準備する必要があり、配信サーバのリソース、特にファイルスペースに影響を与えてしまう。このような問題を改善するため、単一データから各家庭の帯域幅に合わせたサイズのデータを作成し、伝送する事を考えたい。

そこで本研究は、動画像形式として MotionJPEG2000 に着目し、Region of Interest (ROI) 機能を利用した独自のフレーム間圧縮をサポートする形式を考え、動画配信における配信データ量を抑える事を試みた。

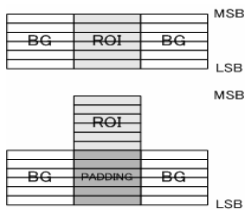
2.1. JPEG2000 と MotionJPEG2000

JPEG2000 は、ウェーブレット変換を用いた画像圧縮形式である。Embedded Block Coding with Optimized Truncation (EBCOT) というエントロピー符号化を用いる事で、ハフマン符号化などの一般的な符号化よりも高い符号化効率で圧縮する特徴を持つ。

MotionJPEG2000 は、動画の各フレームを JPEG2000 で個別に圧縮し連続的に並べたストリーム構造を持つ形式である。そのため、前後のフレームを参照することなく目的のフレームを復元できる利点がある。従って、編集作業・途中再生が容易な動画像形式である。

2.2. Region of Interest (ROI)

ROI とは、画像の注目領域に対応するウェーブレット係数をシフトし、係数の重要度を上げる処理である。ウェーブレット係数が大きい値だと優先的に符号化される特性から、ROI 領域が優先的にデコードされる。



ROI 領域での係数の全ビットプレーンを BG(背景)領域での係数のビットプレーンより優先させる符号化の方法を Max-shift といい、今回はこの方法を採用した(図1)。

図1. ROI の符号化

Compressing method using Region Of Interest of JPEG2000 in motion image delivery

^I Kiminori Asai, Graduate School of Science and Technology, Hirosaki Univ.

^{II} Masayuki Takigahira, MEITEC Corp.

^{III} Kenji Ichijo, Yoshio Yoshioka, Faculty of Science and Technology, Hirosaki Univ.

2.3. 従来のフレーム間圧縮(例. MPEG)

MPEG-1 及び MPEG-2 では、I/P/B という 3 形式のピクチャが存在する。I ピクチャはフレーム間の圧縮を行わない(従って、参照フレームなしで復元できるが圧縮率が低くなる)。P/B ピクチャはデコードの際、予測元となる参照フレームを必要とする。P ピクチャの場合は前方予測用のピクチャを 1 枚(図 2(1))、B ピクチャの場合は前方予測と後方予測用のピクチャを 1 枚ずつ必要である(図 2(2))。B ピクチャは前後 1 ピクチャずつから差分を取得するため高圧縮率だが、参照するピクチャが存在していない場合は復元出来ない。

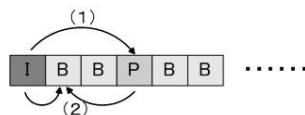


図2. MPEG-1 及び MPEG-2

3. 本提案手法

3.1. ROIを用いたフレーム間圧縮

MotionJPEG2000 は、MPEG の I ピクチャと同様に参照無しピクチャの連続で構成される。本研究で新たに提案する方法は、前フレームとの差分をマスクして ROI 符号化を行う方法(3.2 参照)であり、図 3 に示すような構造をとる。以下に、この方法の利点を挙げる。

- (1) 配信者は ROI 領域(差分領域)のみを配信するだけでよいので、低ビットレート配信が可能であり、演算量も少なくなる。
- (2) 途中から再生を行う際は、開始フレームの BG 部分を含め配信を開始するだけで、他フレームを参照しなくても画像全体を復元可能である。
- (3) 編集に関して、カットされた次のフレームにおいて BG 領域を一緒に送る事で再圧縮を必要としない高速な編集が可能となる。

このように、従来のフレーム間圧縮の利点を維持し、欠点を克服する事が出来る。

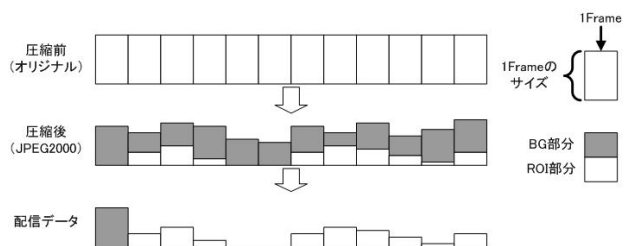


図3. ROI を用いた JPEG2000 動画配信圧縮

3.2. ROIマスク生成アルゴリズム

動き(差分)検出は、前フレームの画素と現在のフレームにおける画素の差分を取る方法とした(図4)。この方法に用いた式は、式1の通りである。

$$\left(\frac{R^2 + G^2 + B^2}{3}\right) \gg \text{ShiftLevel} \quad \dots\dots\dots \text{式1}$$

前フレームと現在フレームの2画像で、各画素の値における式1の値が一致した部分を動きが無いとし、動きのある部分(式1の値が不一致)にROIをかけた。

式1は、RGB各コンポーネントの画素平均をとり、下位ShiftLevelビットを無視し、残りの上位ビットを取り出す式である。つまり式1を2画像間で比較する事により動き(画素値が変動する)検出を行っている。

受信者は、図4に示すようにROI領域のみの復元を行い、復元された差分フレームを再生時の前フレームに重ね合わせる事で、現在フレームの全体画像を生成する事が出来る。

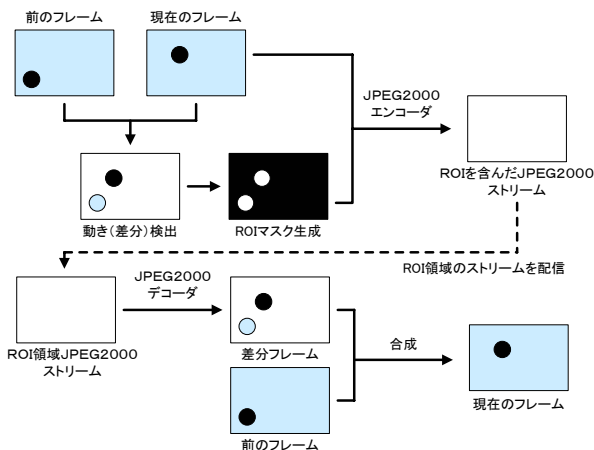
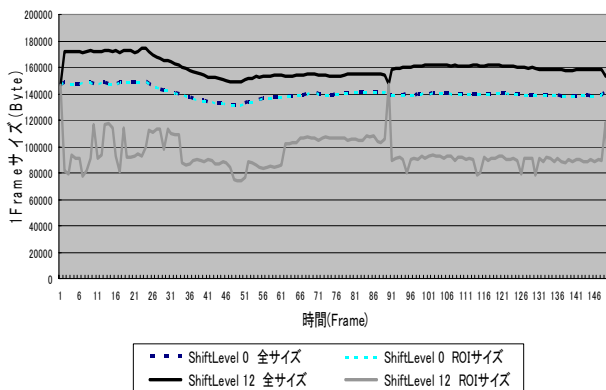


図4. 動き(差分)検出、及び圧縮・復元の流れ

4. 実験結果・考察

4.1. フレーム間圧縮の効果

サンプルデータ7つ(参考URL[2])について実際に圧縮を行い、その中のtennisの結果を次に示す。



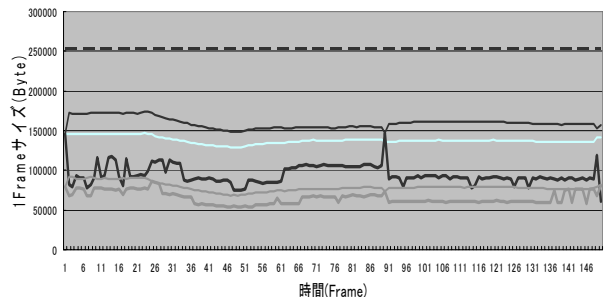
グラフ1. ShiftLevel別のフレームレート

グラフ1より、破線で示されたShiftLevel0の例では、ファイルサイズがROIサイズとほぼ等しい。これから、ROI領域が画像の大部分にかけている事で効果が得られない事が分かった。その原因は、元画像に含まれるノイズで静止部分の画素値が変動し、敏感に捉えた事が考えられる。そのため、これを回避するために式1で動き検出の感度を甘くし、動きの変化を捉え始めたShiftLevel12(グラフ1では実線)を用いる事で効果が得られた。

4.2. 配信データの種類と効果

グラフ2は、フレーム間圧縮を行わない圧縮の場合、フレーム間圧縮を行う場合でlossless圧縮を行った場合、また、lossy圧縮を行った場合の3種類のデータと、元データ(無圧縮)についてフレームレートを比較したグラフである。

グラフ2より、ROIによるフレーム間圧縮をかけた全体ファイルサイズはフレーム間圧縮を行わない場合と比べて平均15%増加したが、実際に配信されるROI部分のみのサイズでは逆に平均20%縮んだ事がわかる。特に、lossless圧縮時における動き検出の効果が大きい事が分かった。また、lossy圧縮時では配信時のフレームレートが少なく出来る事が分かった。



グラフ2. 配信データの種別フレームレート

5. まとめと今後の課題

ROIによるフレーム間圧縮により配信時のビットレートが下げられる見込みがある事が調べられた。しかし、ROIが画像の大部分に適用された際に上昇するビットレートをどう回避するか課題が残った。本研究で提案したROIによるフレーム間圧縮は、様々な用途に応用が出来ると考える。そのため今後は、より多くのコンテンツを用いてその効果を検証する必要がある。

6. 参考文献

[1]JPEG[<http://www.jpeg.org/>]
 ・FinalCommitteeDraft (ISO/IEC 15444) Part1,2,3
 [2]SAMPL[<http://sampl.eng.ohio-state.edu/~sampl/>]
 ・Sequence Photo Sample (Sample1, Sample2, flowg, grandmom, mom, mom&daughter, tennis)