

I-057

動き補償サイズ選択率適応制御に基づく MPEG-2/H.264 トランスコーダの高速化 A Fast MPEG-2/H.264 Transcoder Based on Adaptive Control of Motion Compensation Selectivity

望月 厚志[†] 児玉 知也[†]
Atsushi Mochizuki Tomoya kodama

1. まえがき

MPEG-2 は、現在、地上波デジタル放送やDVDなど広範に用いられている動画像符号化方式である。一方、2003年に規格化されたH.264/MPEG-4 AVC(以下H.264)[1]は、MPEG-2の約2倍の符号化効率を実現できるとされている。従って、既存のMPEG-2をH.264へ変換するMPEG-2/H.264トランスコーダの必要性が高まると考えられる。MPEG-2/H.264トランスコーダの課題の一つは、H.264はMPEG-2と比べ動き補償モードの種類が多く、各モードの評価と最適モードの選択に膨大な処理量が必要な点である。

本稿では、トランスコード時における動き補償サイズの選択率に基づき、当該動き補償サイズの評価が必要か否か判定し、省略することで、動きベクトル探索(以下ME)の処理量を削減する方法を提案し、実験によりその有効性を示す。

2. H.264の動き補償における処理量の問題

MPEG-2では動き補償サイズが16x16と16x8の2種類であるのに対し、H.264では16x16、16x8、8x16、8x8など7種類ある。各動き補償サイズに対しMEを行い、動き補償コスト(残差コスト、動きベクトルの符号コスト等の和)の比較を行うと処理量が膨大になる問題があった。

処理量を削減する方法として、動き補償サイズをMPEG-2のDCT情報から決定する方法[2]や大きいサイズから小さいサイズの順にMEを行い、動き補償コストが極小となるサイズが判明した時点で、以降のサイズのMEを省略する方法[3]などが提案されている。前者は、MPEG-2の符号化アルゴリズムの影響を受けやすく、後者は最低2つのサイズに対しMEを行う必要があった。

3. 提案方法

本稿では大きいサイズのME時に小さいサイズの残差コストも合わせて計算し、各サイズにおける残差コストを比較することで、小さいサイズのMEを打ち切り、処理量を削減する方法を提案する。これにより、1つの動き補償サイズに対するMEのみで打ち切りを行うことができる。また、H.264符号化時に選択された動き補償サイズの割合からME打ち切り判定の閾値を更新することで、MPEG-2の符号化アルゴリズムによらず打ち切りの程度を適切に制御することができる。

3.1 MEの打ち切り判定方法

本提案では、簡単化のためMEは16x16と16x8に対し適応した。本トランスコーダでは、最初に16x16のME(以下16x16ME)を行い、その後16x8のME(以下

16x8ME)が必要か判定する。必要な場合は、16x8MEを行い、必要ない場合は16x8MEを省略する。最後に適切な動き補償サイズを選択する。

16x8ME打ち切り判定方法のブロック図を図1に示す。16x16ME実行時、動きベクトルの残差コストを16x8単位で計算し、16x8の残差コストの最小値を記憶しておく。16x8の最小残差コストは16x8MEを行った場合の残差コストの目安として用いる。16x16ME終了後、16x16の最小残差コストと16x8の上下ブロックの最小残差コストの和との差分を計算する。そして、差分が閾値以下の場合、16x8MEを行っても、動き補償コストは16x16の動き補償コストより小さくならないと判断し、16x8MEを打ち切る。差分が閾値より大きい場合は、16x8MEを行う。

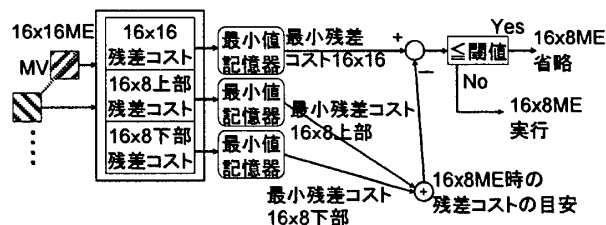


図1 ME打ち切り判定方法のブロック図

3.2 打ち切り判定における閾値更新方法

本稿では、16x16MEと16x8MEのうち16x8MEが選択された割合に基づき、閾値を更新する。閾値を変化させた場合の16x8ME選択率を図2に示す。閾値が小さいと16x8ME選択率が低く、不要な16x8MEを行っていることがわかる。一方、閾値が大きいほど16x8MEの選択率が高い。従って、不要な16x8MEの実行は少なくなる。しかし、16x8MEが適切なマクロブロック(以下、MB)に対しても過剰に16x16MEとしてしまう可能性が高まるため、画質の低下が起こる。16x8の選択率が所定の選択率になるよう閾値を制御することで、画像に合わせたバランスの良い閾値を設定できると考えられる。

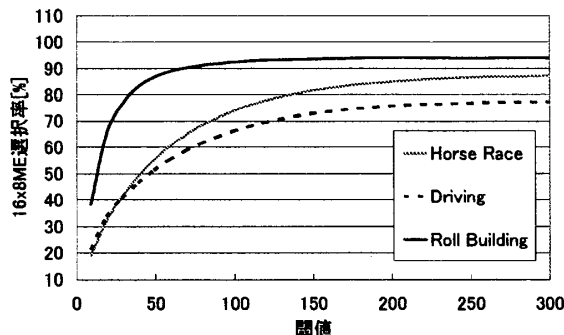


図2 閾値と16x8ME選択率の関係

[†]株式会社 東芝 研究開発センター

閾値の更新は式(1)に示す、閾値と 16x8ME選択率の関係を近似した式を用いる。

$$P_{16x8} = \frac{l \times Th}{1 + k \times Th} \quad (1)$$

ここで、 P_{16x8} は 16x8ME の選択率、 Th は閾値、 l と k は l/k が閾値 Th の値が無限大における P_{16x8} の極限値を表し、 1 以下の適当な値に決めておく。また、 P_{16x8} の所定の選択率 $target_rate_{16x8}$ を定めておく。

閾値更新は、 P_{16x8} が $target_rate_{16x8}$ となるような閾値を、過去の符号化結果から求める P_{16x8} と、あらかじめ定めておいた l と k の関係から求めることで行う。まず、ME が所定の回数行われたとき、過去の符号化結果から P_{16x8} を求める。次に、求めた P_{16x8} と現在の閾値 Th を式(1)に代入し、 l と k を求める。そして、 $target_rate_{16x8}$ を与える、新しい閾値 Th_new を式(2)から得る。

$$Th_new = \frac{target_rate_{16x8}}{l - k \times target_rate_{16x8}} \quad (2)$$

本トランスコードでは、16x8ME の選択率と閾値の関係を実測値から $l/k=0.95$ 、 $target_rate_{16x8}=0.80$ とした。

4. 性能評価

提案手法の有効性を確認するため、フルHD(1920x1080i)のMPEG-2 画像 4 種類を用いたトランスコードを行った。使用した画像はITE標準動画像のHorse Race、Drivingとビルを下から見上げる形でロールしながら撮影したRoll Building、モノスコパターが回転するTest Patternである。実験は 16x8MEを必ず行う場合(以下All_16x8)、16x8MEを全く行わない場合(以下No_16x8)、提案手法および16x16ME、16x8ME、8x16MEに対しMPEG-2Mvを中心に±1.75 画素MEを行う場合(以下Reference)の 4 種で行った。Referenceは動き補償サイズ、ME探索範囲を十分に確保した場合に相当する。表 1にH.264 符号化条件、表 2にMPEG-2 符号化条件を示す。

まず、表 3にAll_16x8 におけるH.264 符号化の実行時間を基準としたときのH.264 符号化実行時間削減率(%)を示す。提案手法によるAll_16x8 からの実行時間の平均削減率は6.09%と、No_16x8 の平均削減率7.51%に近い。

次に、表 4と表 5にAll_16x8 を基準としたBD-SNR[4]を示す。 Δ Bitrateは同一PSNRにおける符号量削減率(%)、 Δ PSNRは同一ビットレートにおける輝度信号に関する画質改善度合(dB)を示す。Horse Race、Drivingでは、各手法の符号化性能に大きな違いはみられない。これは、MB内の物体の動きがほぼ一様で、16x16MEのみでも十分動き補償を行えるためと考えられる。一方、回転運動によりMB内の物体の動きが異なるRoll Building、Test Patternの場合、No_16x8 の Δ PSNRはそれぞれ-0.11dB、-0.38dBである。対して、提案手法の Δ PSNRはそれぞれ-0.02dB、-0.08dBとAll_16x8に近い符号化性能であることがわかる。

また、図 3にTest Pattern、図 4にRoll Buildingにおけるレート歪曲線を示す。横軸がビットレート[kbps]、縦軸が輝度信号に対するPSNR[dB]である。レート歪曲線より、提案手法がNo_16x8 よりもAll_16x8に近い符号化性能を実現できている。

表 1 H.264 符号化条件

Profile	High Profile	
予測構造	IBBPBBP...	
I-picture 周期	15	
フレーム数	Horse Race	420
	Driving	420
	Roll Building	105
	Test Pattern	240
エントロピー符号化	CABAC	
ピクチャ構造	MBAFF	
量子化パラメータ	20, 24, 28, 32(B-picuterは+2)	
ME探索範囲	MPEG-2Mv を中心に±0.25Pel (十字5点)	
	Reference は MPEG-2Mv を中心に±1Pel -> ±0.5Pel -> ±0.25Pel の階層探索	

表 2 MPEG-2 符号化条件

予測構造	IBBPBBP...
I-picture 周期	15
progressive_frame	0
frame_pred_frame_dct	0
ピクチャ構造	フレーム構造
ビットレート	20Mbps

表 3 実験結果(実行時間)

画像	All_16x8 からの実行時間削減率 (%)		
	No_16x8	提案手法	Reference
Horse Race	7.72	6.65	-92.19
Driving	6.59	5.50	-100.96
Roll Building	9.62	7.64	-85.57
Test Pattern	6.07	4.52	-91.21
Average	7.51	6.09	-92.55

表 4 実験結果(Δ Bitrate)

画像	Δ Bitrate(%)		
	No_16x8	提案手法	Reference
Horse Race	0.31	-0.12	3.29
Driving	-0.48	-0.32	5.65
Roll Building	-5.10	-0.92	8.65
Test Pattern	-11.25	-1.97	10.23
Average	-4.13	-0.83	6.95

表 5 実験結果(Δ PSNR)

画像	Δ PSNR(dB)		
	No_16x8	提案手法	Reference
Horse Race	0.00	0.00	0.02
Driving	-0.01	0.00	0.07
Roll Building	-0.11	-0.02	0.21
Test Pattern	-0.38	-0.08	0.37
Average	-0.12	-0.02	0.16

最後に、図5にH.264符号化実行時間と符号量削減率の平均値を示す。横軸がAll_16x8の実行時間を100%としたときの相対実行時間[%]で縦軸がAll_16x8を基準とした Δ Bitrate[%]である。提案手法はAll_16x8、No_16x8を結ぶ直線よりも左上に位置し、実行時間と符号量削減率のバランスが良いことがわかる。

本提案手法に加え、符号化処理の並列度向上などの高速化を行った。その結果、地上波デジタル放送でも用いられる1440x1080iのMPEG-2画像に対し、MPEG-2復号からH.264符号化までを2.66GHzクアッドコアCPUを用い、平均30.8frame/秒のリアルタイムで処理することができた。

5. まとめ

本稿ではMPEG-2/H.264トランスコードにおける動き補償サイズ選択の処理量を削減するため、動き補償サイズを選択率に基づき動き補償サイズの評価を適応的に行う方法を提案した。動き補償サイズ16x16と16x8に対し、提案手法を適応し実験を行った。その結果、H.264の符号化実行時間を提案手法と16x8MEを必ず行う場合とで比べた場合、平均6.09%削減することができた。また、提案手法は16x8MEを必ず行う場合と比べ、 Δ PSNRが平均0.02dB低下とはほぼ同等であった。

参考文献

- [1]“Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding,” ISO/IEC 14496-10, 2005.
- [2]Gao cheo, Yong-dng Zhang, Shou-xun Lin, Feng Dai, “Efficient Block Size Selection for MPEG-2 to H.264 Transcoding”, Proceedings of the 12th annual ACM International Conference on Multimedia, 2004, pp.300-303
- [3]筑波健史, 永吉功, 花村剛, 富永英義, “MPEG-2/H.264トランスコードにおける符号化モード選択手法に関する検討”, 情報処理学会研究報告, 2005-AVM-51, Dec. 2005.
- [4]G.Bjontegaard, “Calculation of average PSNR differences between RD-Curves,” ITU-T VCEG, VCEG-M33, April, 2001.

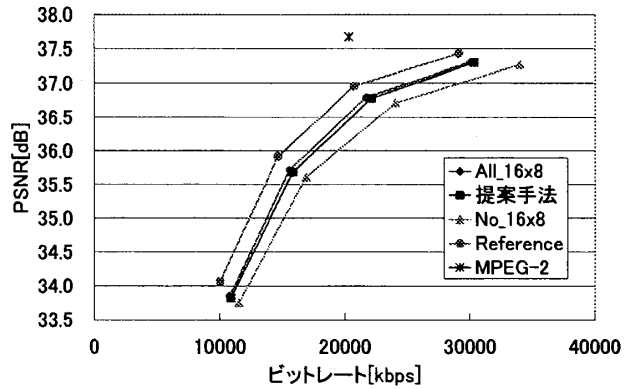


図3 Test Patternにおけるレート歪曲線

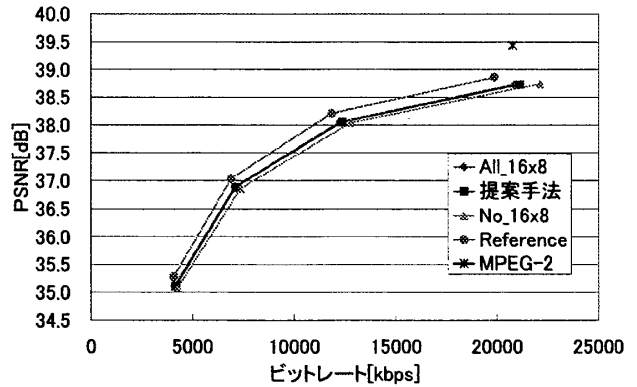


図4 Roll Buildingにおけるレート歪曲線

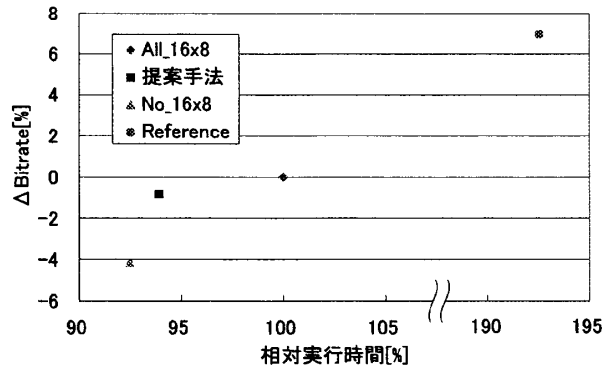


図5 実行時間と符号量削減率