

LI-003

H.264 の重み予測を利用したネガポジ反転に関する一検討 A Study on Negative/Positive Inversion using Weighted Prediction in H.264

加藤 晴久[†]
Haruhisa KATO

米山 暁夫[†]
Akio YONEYAMA

滝嶋 康弘[†]
Yasuhiro TAKISHIMA

1. 序論

映像コンテンツの編集に用いられる様々な特殊効果の中で、衝撃的な場面を効果的に表す演出としてネガポジ反転などが用いられている。一方で、携帯端末向けの地上波デジタル放送(ワンセグ)や次世代DVDのBlu-ray及びHD DVDでは、映像圧縮方式としてH.264を採用している。多くの素材映像がH.264で符号化されている中で、各種編集効果を適用して2次利用する場合は、完全に復号してから編集する必要がある。しかし、復号と再符号化にかかる膨大な計算量と、再符号化による画質劣化が問題となっている。コンテンツへの需要の高まりから、コンテンツ生成や2次利用の高効率化が求められており、H.264の編集プロセスの改善が必要とされている。本稿では編集作業時間の大幅な短縮と編集後の画質維持を図るため、H.264で符号化された映像コンテンツに対する編集加工にH.264の重み予測を利用してネガポジ反転を実現する方式について考察する。

2. 従来のエフェクト技術

符号化された映像にエフェクトを付与する方法としては、画素領域で編集する方式と符号領域で編集する方式がある。Baseband変換は、符号化された映像コンテンツを復号してから画素領域でエフェクトをかけた後に再符号化する。しかし、復号と再符号化に膨大な時間が掛かるだけでなく、再符号化による画質劣化が問題となっている。

一方、[1]および[2]では、符号領域でコントラストを制御する。[1]はMPEG-2のDCT係数を定数倍することでコントラストを調整する。[2]は、JPEGのDCT係数を使ってコントラスト比を最適化する。しかし、ネガポジを反転する編集は想定されておらず、そのまま適応するだけではPSNRが大きく劣化する可能性がある。特に、H.264においてはIntra予測など空間的な依存関係が画質劣化を伝播させる。

3. H.264の重み予測

重み予測はH.264のMain以上のプロファイルで利用でき、重み係数 w_i とオフセット係数 o_i を用いて画素値の変化を予測する。量子化済み重み係数 \hat{w}_i とオフセット係数 \hat{o}_i はスライスヘッダに格納され、参照フレーム毎に設定できる。参照画素を p_i とすると、片予測による予測値 p' は式(1)で与えられ、双予測の場合は式(2)で与え

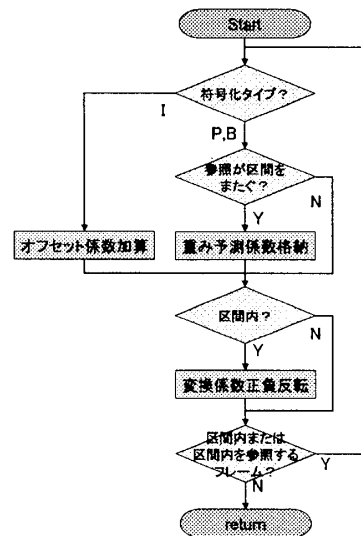


図1 編集処理

られる。

$$p' = \text{Clip}(((\hat{w}_1 \cdot p_1 + 2^{\log WD - 1}) \gg \log WD) + \hat{o}_1) \quad (1)$$

$$p' = \text{Clip}(((\hat{w}_1 \cdot p_1 + \hat{w}_2 \cdot p_2 + 2^{\log WD}) \gg (\log WD + 1)) + ((\hat{o}_1 + \hat{o}_2 + 1) \gg 1)) \quad (2)$$

Clipは0から255に丸め込む処理を表し、 \gg は右シフト演算を示す。 $\log WD$ は量子化済みの重み係数 \hat{w}_i を実数に戻す役割を果たす。

Weighted Predictionによって、画素そのものの変化を表現できるため、フェードなどのトランジションの予測精度を向上させることで、画質を大幅に改善することができる。なお、次節以降では簡単のため量子化前の重みパラメータ w_i 、 o_i を使って議論する。

4. 重み予測を利用したネガポジ反転

始めに、提案法はH.264コンテンツから編集に必要な符号情報を抽出する。ここでの復号は符号情報までの部分的な復号であり、フレームの符号化タイプ、ブロックの符号化モード、整数変換係数、参照フレーム情報を抽出する。編集区間に応じて重み係数情報を付与するとともに、整数変換係数を編集することにより画素値を変化させ、コントラストを反転した係数を生成する。続いて、編集によって生じる誤差を相殺する変換係数を加算することで、誤差の伝播による画質劣化を防ぐ。最後に、編集された符号情報を可変長符号化し、ネガポジ反転されたH.264コンテンツを出力する。

[†](株) KDDI 研究所, KDDI R&D Laboratories Inc.
{hkato, yoneyama, takisima}@kddilabs.jp

高速かつ高画質なネガポジ反転を実現するために、提案法は符号化タイプや参照フレームの性質によって異なる処理を適用する(図1)。

4.1 重み係数の設定と格納するフレームの選択

画素値 p_i をネガポジ反転した画素値 p'_i は画素の最大値 p_{max} を使って次式で表されるため、重み予測は重み係数 w_i に-1, オフセット係数 o_i に 255 を設定することで、ネガポジ反転を実現できる。

$$p'_i = p_{max} - p_i \quad (3)$$

入力フレームが片予測である場合は、 $w_1 = -1$ および $o_1 = 255$ を格納するが、入力フレームが双予測である場合は、編集区間をまたぐ参照フレームに対する重み予測係数を格納する。編集区間内であっても、ネガポジ反転したフレームを参照するフレームに重み予測は必要ない。すなわち、L0 予測の参照フレームが指定区間をまたぐ場合は $w_1 = -1, w_2 = 1, o_1 = 255, o_2 = 0$ を格納し、L1 予測の参照フレームが指定区間をまたぐ場合は $w_1 = 1, w_2 = -1, o_1 = 0, o_2 = 255$ を格納する。双予測の参照フレームがいずれも指定区間をまたぐ場合は、 $w_1 = -1, w_2 = -1, o_1 = 255, o_2 = 255$ を格納する。

よって、入力フレームが P または B であり、かつ参照フレームが指定された編集区間をまたぐ場合、重み予測係数を格納する。編集区間内にあつて参照フレームが編集区間をまたぐフレームは符号情報をネガポジ反転するフレームに相当し、編集区間外にあつて参照フレームが編集区間をまたぐフレームはネガポジ反転された符号情報を再反転させるフレームに相当する。

4.2 整数変換係数のネガポジ反転

重み予測は P および B スライスでのみ用いられるため、I スライスをネガポジ反転するには重み係数 w_i およびオフセット係数 o_i に応じた変化を与える必要がある。また、P および B スライスの残差成分には重み係数 w_i に応じた変化を与える必要がある。

まず、重み予測と同等の処理を整数変換係数に直接反映させることを検討する。4*4 整数変換行列および逆変換行列、変換係数をそれぞれ C, D, X で表すとき、Baseband 変換で重み予測された係数 X' は式(4)で表される。

$$X' = (C(w_i \cdot (D^T X D) + o_i \cdot I)C^T) \circ E^{-1} = w_i \cdot X + o_i \cdot (CIC^T) \circ E^{-1} \quad (4)$$

ここで \circ および T はそれぞれ Hadamard 積、転置操作を示す。また、 I は全要素が 1 の 4*4 行列であり、 E は直交変換のためのスケーリング行列を表す。((CIC^T) $\circ E^{-1}$) は DC 成分にだけ 64 が残る。よって、符号領域での重み予測は、変換係数 X に重み係数 w_i を乗じた後、オフセット係数 $64o_i$ を直流成分にだけ加算することになる。

よって、重み係数 $w_i = -1$ による重み予測と同等の効果を符号領域で変換係数に適用する操作として、IPB の全ての変換係数の正負を反転する。

一方、オフセット係数 $o_i = 255$ による重み予測と同等の効果を符号領域で変換係数に適用する操作として、オフセット係数の定数倍をブロックの DC 成分に加算する。ただし、I スライスにおいて画面左上のブロック以外の係数 X は周囲のブロックから Intra 予測で予測値を生成し、オフセット係数がすでに含まれている。また、PB スライスにおいては、参照フレームがすでにオフセット係数を含むので、オフセット係数を変換係数に加算する必要はない。よって、オフセット係数の定数倍を加算する対象は I スライスの画面左上のブロックだけとなる。なお、画面左上ブロックは固定値 (=128) の DC 予測に対する残差を符号化するので、固定値との差を吸収する定数倍は $64(o_i - 128(1 - w_i)) = -64$ となる。よって、画面左上ブロックの DC 成分に-64を加算する。ただし、左上の MB 符号化モードが 16x16 であれば、16 個の 4x4 ブロックすべてが固定値による予測になるため、16 個全ての DC 成分に同値を加算する必要がある。なお、図 1 では正負反転の前に加算処理を配置したため、-64 ではなく 64 を加算する。

4.3 四捨五入による誤差の補正

次に、前節の編集処理で変換係数を操作したことによって発生する副作用を補正する。副作用とは、編集処理手段によって変換係数の正負を反転させたため、変換符号化の逆変換を行う場合、四捨五入の方向も反転することを指す。

具体的な数値例として、次式はネガポジ反転処理を適用する前の 4x4 整数変換係数と、その逆変換処理後の画素値を示す。

$$\begin{pmatrix} 768 & 640 & 0 & 0 \\ 0 & 800 & 0 & 0 \\ -1024 & 320 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \xrightarrow{\text{逆変換}} \begin{pmatrix} 24 & 10 & -18 & -31 \\ 39 & 34 & 22 & 17 \\ 27 & 27 & 29 & 29 \\ -1 & -3 & -5 & -6 \end{pmatrix}$$

一方、次式はネガポジ反転処理を適用した後の 4x4 整数変換係数と、その逆変換処理後の画素値を示す。

$$\begin{pmatrix} -768 & -640 & 0 & 0 \\ 0 & -800 & 0 & 0 \\ 1024 & -320 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \xrightarrow{\text{逆変換}} \begin{pmatrix} -23 & -10 & 18 & 32 \\ -39 & -34 & -22 & -17 \\ -27 & -27 & -29 & -29 \\ 2 & 3 & 5 & 7 \end{pmatrix}$$

逆変換後の画素値を比較すると、一部の画素値(4x4 の四隅)が本来なるべき数値より 1 だけ大きくなっていることが分かる。1 ブロックでは小さな誤差であっても、Intra 予測では予測値として他のブロックへ伝播するため、誤差の蓄積により画面の右下にかけて画質が大きく劣化する可能性がある。

また、同様の誤差は逆整数変換だけでなく、予測画像の生成時にも発生しうる。例えば、平均値予測で 4 つの画素値 (2, 3, 4, 5) から予測画素を生成する場合、式(5)で予測値 4 が得られる。

$$((2+3+4+5)+2) \gg 2 = 4 \quad (5)$$

これを、ネガポジ反転した画像で予測画素を生成すると、式(6)で予測値252が得られる。

$$((253 + 252 + 251 + 250) + 2) \gg 2 = 252 \quad (6)$$

しかし、期待される予測画素値は $(255 - 4) = 251$ であるため、この場合+1の誤差が発生する。

いずれの誤差も、計算結果の小数が0.5となる場合においてのみ発生する。すなわち、H.264の仕様で規定されている $\text{Clip}(x+0.5)$ という式で四捨五入を行った時に、編集処理の前と後との双方のケースで繰り返りが生じる。しかし、逆整数変換で誤差が発生する箇所を正確に特定するには、復号が必要となるため、Basebandでの補正は計算量を十分に削減できない。よって、逆整数変換と予測画素生成による四捨五入誤差を符号領域で補正するために、提案法は各画素について誤差発生確率を算出し、確率に応じた補正量を変換符号化した行列を各変換係数に加算する。

まず、逆整数変換は変換行列との行列積および6bitの右シフトからなる。正規化のためのシフト演算は誤差には影響しないが、変換係数の左右から掛ける変換行列は1/2を含むため、変換後の全画素には2と4による除算が含まれる。除数nの剰余がrになる確率を $p_{r,n}$ とすると、誤差の発生確率は $p_{1,2}p_{0,4} + p_{0,2}p_{2,4}$ となる。ただし、4の被除数は変換係数の(1,1), (1,3), (3,1), (3,3)成分に限られるため、量子化処理によって0になっていることが多い。(1,1)成分が非0である可能性もあるが、(1,1)成分に対する4の除算はIntra予測に用いられない場所(4x4ブロック内部の(1,1), (1,2), (2,1), (2,2))に位置する。4の剰余による誤差の伝播は無視できるため、提案法は $p_{0,4} = 1, p_{2,4} = 0$ と設定する。一方、2に対する剰余が一様分布と仮定すると誤差発生確率は1/2となるため、-1/2を変換符号化した下記の行列を加算することで逆整数変換による誤差を補正する。

$$-\frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} \xrightarrow{\text{変換}} \begin{pmatrix} -16 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

予測画素生成による四捨五入に関しては、Intra予測モードによって誤差の発生確率が異なる。平均を計算しないmode0, 1は補正を必要としない。他のmode2~8に関しては、個々の予測値生成式から前段と同様に誤差発生確率を算出し、予測モードに応じた補正行列を加算することで予測画素生成による誤差を補正する。例えば、mode8の誤差発生確率と補正行列は次式で与えられる。

$$-\frac{1}{4} \begin{pmatrix} 2 & 1 & 2 & 1 \\ 2 & 1 & 2 & 1 \\ 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \xrightarrow{\text{変換}} \begin{pmatrix} -4 & 2 & 0 & -2 \\ -4 & 1 & 0 & -2 \\ 1 & 1 & 0 & 0 \\ 0 & -2 & -1 & 1 \end{pmatrix}$$

5. シミュレーション結果と考察

提案法の有効性を検証するため、Pentium4 3.2GHzのWindows PC環境で画質及び変換時間を評価する実験を

表1 フェード編集処理時間(sec)

Baseband 変換	デコード	3.72	125.2
	ネガポジ反転	0.20	
	エンコード	121.28	
提案法	-		0.58

行った。Mobile & Calendar と Flower Garden (360*240, 60frame, 29.97fps)を予めH.264の参照ソフトJM 10.2 [3]でH.264 (M=3, N=15, Main profile, CABAC, 1.0Mbps)に符号化しておき、15~45フレームをネガポジ反転した。提案法の比較対象として、Baseband変換と従来法[1]を用いた。ただし、[1]はMPEG2のコントラスト調整にしか対応していないため、Weighted Predictionを用いず提案法による係数制御だけで代用した。Baseband変換は画素領域でネガポジ反転した後に、前述の条件でH.264に再符号化してからPSNRを測定した。

速度比較では、Baseband変換および提案法の処理時間を表1に示す。Baseband変換によるネガポジ反転自体は0.2秒であるが、再符号化に掛かる時間が121.28秒と全体の大部分を占める。一方、提案法は再符号化が必要ないだけでなく、部分的な復号と一部の係数操作で済むためBaseband変換に対して215倍の高速化を達成した。

画質に関しては、ネガポジ反転後のPSNRの比較を図2と図3に示す。PSNR算出に用いた原画は、H.264で符号化する前の非圧縮画像をネガポジ反転した画像とした。第12フレーム以前と第45フレーム以後はネガポジ反転していないので、提案法と従来法のPSNRには差がない。しかし、ネガポジ反転した15~45フレームは、提案法と従来法のPSNRには顕著な差が生じる。補正処理がない従来法では、誤差がフレーム内だけでなくフレーム間でも伝播していくため、PSNRに大きな劣化が生じている。なお、従来法が編集区間前から画質劣化が始まっているのは、第13, 14フレームのBスライスが劣化した第15フレームを参照するためである。一方、画素を補正している提案法に、大きな画質劣化は見られない。

Baseband変換は原画像と同じ方法でネガポジを反転するが、再符号化による劣化が発生する。提案法は符号領域で処理するため、再符号化を必要とせず高い画質を維持している。Baseband変換の劣化度合いはビットレートに依存するが、この実験では提案法はBaseband変換より最大2.8dB以上高いPSNRを実現した。

図4から図9はそれぞれ原画像、従来法[1]および提案法で編集した画像を15フレーム目から5フレーム置きに表している。提案法は原画像とほぼ同じ効果を得ることができている。また、編集区間後のフレームも正しく再反転する。従来法では、画面の右下に掛けて若干明るくなっている。図10は、従来法で生成された画像と原画との差分を求め、差分絶対値が大きいほど赤くしている。図11は、提案法で生成された画像と原画像との差分を表している。図10からは右下ほど誤差が大きいことが分かり、図11からは誤差を補正できていることが分かる。

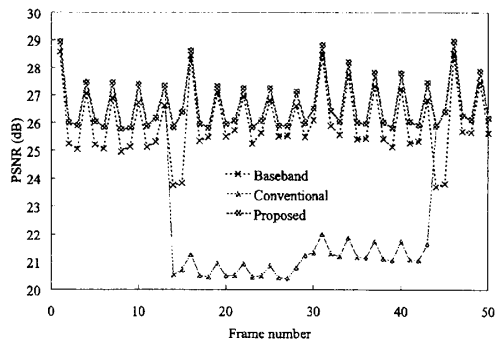


図2 画質比較 (Flower garden)

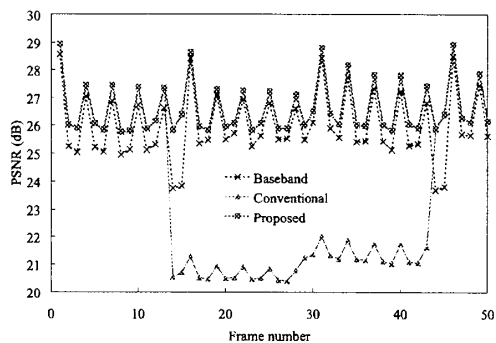


図3 画質比較 (Mobile & Calendar)

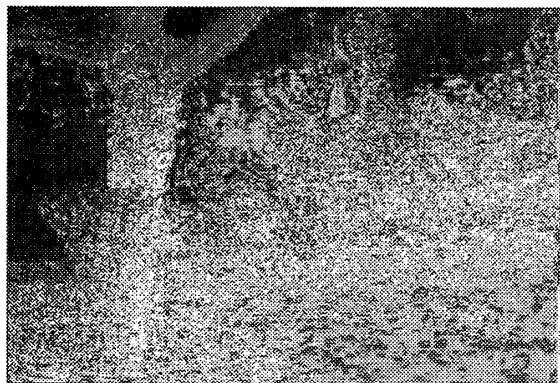


図10 従来法 [1] による編集画像と原画像との差分



図11 提案法による編集画像と原画像との差分



図4 原画像



図5 従来法 [1]



図6 提案法



図7 原画像



図8 従来法 [1]



図9 提案法

6. 結論

本稿では、符号領域におけるネガポジ反転方式について提案した。提案法は H.264 コンテンツに対して重み予測を利用することで高速なネガポジ反転効果を付与した。また、提案法はネガポジ反転に伴う誤差を補正することで高画質な編集を可能とし、補正処理を持たない従来法より大幅に画質を改善した。Baseband 変換と比較しても、提案法の画質は最大で 2.8dB 改善し、200 倍以上高速になることを実験から確認した。

参考文献

- [1] An, K., Sun, J. and Du, W.: A Brightness Adjustment Method for MPEG-2 Compressed Video Editing, in *International Conference on Consumer Electronics (ICCE)*, pp. 311-312 (2006).
- [2] Tang, J., Peli, E. and Acton, S.: Image enhancement using a contrast measure in the compressed domain, *IEEE Signal Processing Letters*, Vol. 10, No. 10, pp. 289-292 (2003).
- [3] Suehring, K.: H.264 / AVC reference software, <http://bs.hhi.de/suehring/tml>.