

## H.264/MPEG-4 AVC の重み付き動き補償における高速重み係数推定の一検討

A Study on Fast Weighting Factor Determination  
for H.264/MPEG-4 AVC Weighted Prediction加藤 晴久<sup>†</sup>  
Haruhisa KATO中島 康之<sup>†</sup>  
Yasuyuki NAKAJIMA

## 1. 序論

ディゾルブやフェードなどの映像効果は映像制作過程でしばしば利用されている。ディゾルブは、2つの画像クリップに対して重みを徐々に変化させながら切り替えていく重み付き平均化処理である。一方フェードは、ディゾルブで用いる2画像のうち片方を白画像または黒画像に設定することに相当する。しかし映像圧縮技術の観点から捉えると、これまでの符号化方式では画素全体の段階的な変化を補償する方式がなかったため、フェードシーンやディゾルブシーンにおける符号化効率の劣化が問題となっていた。

映像圧縮方式の国際標準規格 H.264/MPEG-4 AVC (ISO/IEC 14496-10) ではフェード・ディゾルブシーンの符号化効率改善を目的として、従来の動き補償方式のほかにフレーム間の重み付き動き補償方式が採用された。映像編集過程で明示的にフェード効果を挿入する場合は、そのフェードの種別および区間情報等を得ることによって重み付き動き予測情報を決定することができるものの、通常はすでにフェードが付加された映像から重み付き動き予測情報を推定しなければならない。

本稿では重み付き動き予測情報を高速に推定する方式を提案し、検証実験から提案方法が従来法と比べてフェードにおける圧縮効率を大幅に改善することが可能なことを報告する。

## 2. 従来の重み係数推定法

菊池らによる寄書 [1] や、K. Lillevold による寄書 [2] では、MPEG-4 AVC/H.264 の重み付き動き補償方式がフェード・ディゾルブシーンにおいて効果的であることが示されている。

文献 [1] における重み係数の推定法では、2枚の参照フレームに対して2種類の重み付け係数を用いて予測する方法が提案されており、過去と未来の参照フレームに対しては (0.5, 0.5)、参照フレームが2枚とも過去の場合は (-1, 2) を利用している。

また、文献 [2] では時間的なフレーム間距離により重み係数を決定する方法が提案されている。

一方、H.264 の参照ソフト [3] は各フレームの輝度平均の比を重み付き動き予測情報として利用している。但しこの場合、輝度値に比例しない変化に対しては、精度良く補償できない場合がある。例えば、白い画面へのフェードアウトでは輝度平均による推定は上手く機能しない。ま

た、重み付き動き予測情報のうちオフセット係数を用いていないため、重み付き動き補償方式の性能を十分に発揮できていない。

## 3. 提案する高速重み係数推定法

## 3.1 重み付き動き補償と係数の最適解

1枚の参照フレームを用いる場合、重み付き動き予測情報として重み係数  $w_1$  とオフセット係数  $w_2$  を用いると、処理対象画素  $p$  は重み付き動き補償された参照画素値  $p'_i$  によって式 (1) で近似される。

$$p_i \approx w_1 p'_i + w_2 \quad (1 \leq i \leq n) \quad (1)$$

但し、 $n$  は処理対象画像の画素数を示す。

このとき、処理対象画像の各画素  $p_i$  と重み付き動き補償された各画素  $w_1 p'_i + w_2$  との類似度を測る尺度として、二乗誤差の総和  $e$  を式 (2) で定義したとき、最適解は最小二乗法を用いて求められる。すなわち、重み係数  $w_1$  およびオフセット係数  $w_2$  による偏微分を0にする値は、それぞれ式 (3) および式 (4) で導かれる。

$$e = \sum_{i=1}^n (p_i - w_1 p'_i - w_2)^2 \quad (2)$$

$$w_1 = \frac{n[p p'] - [p][p']}{n[p'^2] - [p']^2} \quad (3)$$

$$w_2 = \frac{1}{n} ([p] - w_1 [p']) \quad (4)$$

ここで、 $[\ ]$  は式 (5) で表されるガウスの和記号を示す。

$$[p] = \sum_{i=1}^n p_i \quad (5)$$

式 (3) と式 (4) は最適な重み付き動き予測情報を導くが、式 (3) は積和演算や二乗和の算出にかかる処理負荷が比較的大きい。よって、重み係数  $w_1$  算出の処理負荷を低減する方式を検討する。

## 3.2 フェードにおける画素変化の特性

フェード中の画素値に注目すると、フェード前の画素値とフェード完了後の画素値との差に比例して変化する。特に黒からのフェードインおよび黒へのフェードアウト(以後、黒フェード)の場合は、画素値そのものに比例して変化する。例えば、黒へのフェードアウトでは、明るい画素も暗い画素も同じフレーム数を経て0 (NTSCでは16)に収束するために、画素値が大きいかほど変化幅も大きい。逆に、白からのフェードインおよび白へのフェードアウト

<sup>†</sup> (株) KDDI 研究所, KDDI R&D Laboratories Inc.

ト(以後、白フェード)の場合は、画素値に反比例して変化する。例えば、白へのフェードアウトでは、明るい画素も暗い画素も同じフレーム数を経て255(NTSCでは235)に収束するために、画素値が大きいほど変化幅は小さい。

よって、性質の異なるフェードを画一的に扱うのではなく、白と黒の2種類のフェードに分類して、それぞれに適した重み付き動き予測情報推定法を用意する。

### 3.3 フェード判別

フェードの種類に適した重み付き動き予測情報推定法を適用するため、白フェードか黒フェードかを判断する。但し、時間的な前後関係とフレームの画素平均等を比較するだけでは、黒フェードか白フェードの判断は困難である。例えば、白からのフェードインと黒へのフェードアウトでは、いずれも過去のフレームの方が画素平均値が大きいため区別できない。

よって、フェード時の画素値は差に比例して変化することを考慮して、任意の明るい画素と任意の暗い画素のどちらが大きくなるかによって判断する。すなわち、図1のように暗い画素の変化率が小さくかつ明るい画素の変化率が大きいとき、黒フェードと判断し、図2のように暗い画素の変化率が大きくかつ明るい画素の変化率が小さいとき、白フェードと判断する。

フェード種別が判断した後であれば、フェードインかフェードアウトかは時間的な前後関係とフレームの画素平均等を比較するだけで判断できる。

### 3.4 黒フェードに対する重み付き動き予測情報の推定

式(1)が比例関係そのものを表すので、黒フェードにおいては適切な重み係数  $w_1$  だけで重み付き動き補償が効果的に機能する。よって、オフセット係数  $w_2$  を省略して考える。

$$p_i \approx w_1 p'_i \quad (6)$$

重み係数  $w_1$  が二乗誤差  $e$  を最小にするために最小二乗法を用いると、重み係数  $w_1$  は式(7)で求められる。

$$w_1 = \frac{[pp']}{[p'^2]} \quad (7)$$

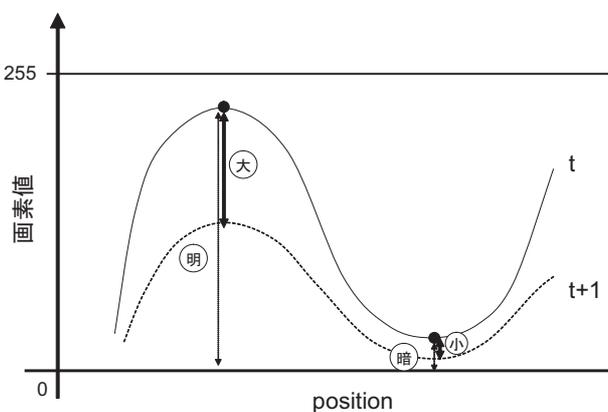


図1 黒フェードアウトにおける画素値変化

式(7)は式(3)と比較して、二乗和が必要ないため計算量の削減が図られている。

ここで、文献[3]で用いられている式(8)の画素平均比と比較する。

$$w_1 = \frac{[p]}{[p']} \quad (8)$$

式(7)に対する式(8)の近似精度を評価するため、 $n=2$ の場合を考える。ひとつの画素値をもう一方の画素値に対する比  $0 \leq r < 1, 0 \leq s < 1$  を使って式(9)のように表現したとき、 $r$  および  $s$  にどのような条件が成立すれば式(8)の近似が有効なのか検討する。

$$p_1 = a \quad p_2 = ra \quad p'_1 = b \quad p'_2 = sb \quad (9)$$

このとき、式(7)による重み係数  $w_1$  は、式(10)で求められる。

$$w_1 = \frac{ab + rsab}{b^2 + s^2b^2} \quad (10)$$

一方、式(8)は重み係数  $w_1$  を式(11)で導く。

$$w_1 = \frac{a + ra}{b + sb} \quad (11)$$

両者の差が十分小となる条件は、式(12)のように差の絶対値が任意の正定数  $\epsilon > 0$  に抑えられることである。

$$\left| \frac{ab + rsab}{b^2 + s^2b^2} - \frac{a + ra}{b + sb} \right| < \epsilon \quad (12)$$

画素値  $a, b$  は任意であるので、式(12)において画素値に依存せず大勢に影響する項だけに注目すると、式(13)となる。

$$|(1-s)(s-r)| < \epsilon \quad (13)$$

よって、 $s=1$  の近傍あるいは、 $r=s$  の近傍であれば、式(8)の画素平均比が式(7)の近似として有効であることを示す。すなわち、画素値すべてが一定あるいは画素値

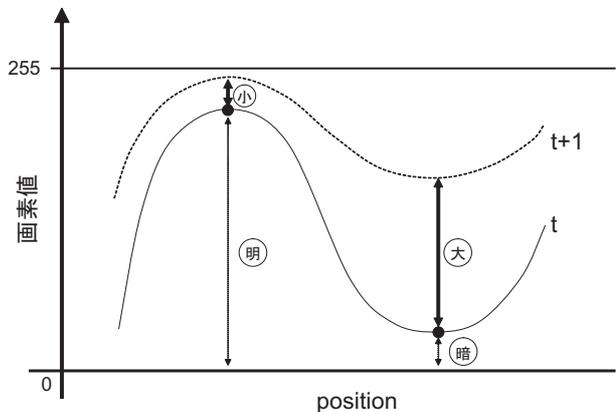


図2 白フェードアウトにおける画素値変化

の変化率が一致することを意味する。具体的には、 $s = 1$  はフェードインの開始点付近およびフェードアウトの終点付近を表し、一方、 $r = s$  は理想的なフェードが実現している状態を表す。あるいは、フェードの画素変化にノイズ成分が含まれていたとしても、平均が0の一般的な加法性ガウス雑音であれば、式(8)の平均処理が多少の変動を打ち消すことができるので多少のノイズに対してはロバスト性を備える。

よって、式(8)は計算負荷が小さい近似式であるだけでなく、上記の条件下で十分な精度を保ったまま重み係数を推定できることを示す。逆に、この条件下であれば  $n > 2$  のときも成立する。式(7)は積和演算の算出にかかる処理負荷も比較的大きいため、黒フェードにおける重み係数  $w_1$  の算出は式(8)を用いる。

一方、オフセット係数  $w_2$  は式(7)で推定した重み係数  $w_1$  を式(4)に代入することで、重み係数  $w_1$  の近似値に対する最適値を求めることもできる。しかし黒フェードにおけるオフセット係数は0であることが多く、効果が小さい上に、ビット量の増加が逆に圧縮効率を下げかねない。また、重み係数を式(8)で近似したときは、最適なオフセット係数  $w_2$  は常に0となる。よって、黒フェードにおけるオフセット係数は0とする。

### 3.5 白フェードに対する重み付き動き予測情報の推定

式(1)に示される画素値そのものに重み係数を乗ずる変換式では、比例関係を前提としているため、画素値に対しては反比例の関係にある白フェードを補償することが困難である。この不完全な変換式において精度を改善するためには、黒フェードでは省略したオフセット係数  $w_2$  で誤差を最小にする必要がある。

重み係数  $w_1$  だけでなくオフセット係数  $w_2$  を重み付き動き予測情報として導出するためには、白フェードにおける画素値の増減が画素値そのものではなく差に比例することを利用する。すなわち、白フェードにおける処理対象画素値  $p_i$  は、参照画素値  $p'_i$  と白(=255)との差に何らかの比例係数  $\alpha$  を介して近似できるものとして定式化できる。

$$p_i \approx p'_i + \alpha(255 - p'_i) \quad (14)$$

このとき、前小節と同様にして、 $\alpha$  は式(14)から式(15)のように推定する。

$$\alpha = \frac{[p] - [p']}{255n - [p']} \quad (15)$$

一方で、式(14)を式(1)の形式と対応させると、式(16)のように変形できる。

$$p_i \approx (1 - \alpha)p'_i + 255\alpha \quad (16)$$

もともとの重み付き動き補償方式(1)との対応関係を考慮すると、重み係数  $w_1$  は式(17)で求めることができる。

$$w_1 = 1 - \alpha \\ = \frac{255n - [p]}{255n - [p']} \quad (17)$$

同様に、オフセット係数  $w_2$  は式(18)で求められる。

$$w_2 = 255\alpha \\ = 255(1 - w_1) \quad (18)$$

しかし、式(17)で画素の和を得た後では、式(18)と式(4)は計算量的に大差がない。よって、式(17)で推定した重み係数  $w_1$  を式(4)に代入して重み係数  $w_1$  の近似値に対する最適値  $w_2$  を得る。

このとき、式(4)のオフセット係数  $w_2$  は重み係数  $w_1$  に依存する。重み付き動き予測情報は量子化されることを考慮して、予め重み係数  $w_1$  に量子化誤差を含めた上で代入すると、オフセット係数  $w_2$  の精度はさらに向上する。

以上のことから、黒フェード、白フェードいずれの場合であっても、重み付き動き予測情報は画素の和の組み合わせによって求めることができるので、積和演算や2乗和を計算する必要がなく提案方式の処理負荷は小さい。

## 4. シミュレーション結果と考察

H.264の参照ソフト(JM7.5c)を使って評価を行った。使用した画像は1秒間のフェードを付加したCIFサイズのForemanであり15fpsで符号化した。提案方式の効果を見るためフェード区間だけを使用し、重み付き動き補償を除いてはBaselineプロファイルを用いた。

重み付き動き予測情報の算出にかかる演算回数を表1に示す。提案方式は黒フェード、白フェードに関わらず画素の和だけを必要とするため、参照ソフトの画素平均比法とほぼ同等の処理負荷で実現可能である。一方、最小二乗法は画素平均だけではなく積和および二乗和も求める必要がある。加減算と乗除算が同じ処理負荷だとしても、提案方式は最小二乗法の1/4の負荷で済むことが分かる。

図3から図6はそれぞれ黒フェードアウト、黒フェードイン、白フェードアウト、白フェードインにおける各方式のPSNRを示す。図3から図6のすべてにおいて、重み付き動き補償の効果が顕著に表れていることが分かる。フェードインとフェードアウトを比べると、フェードインの方が全体的にPSNRが低い傾向がある。これは参照フレームが処理対象フレームより平坦であることに起因するものと考えられる。次に黒フェードと白フェードを俯瞰すると、黒フェードには重み付き動き予測情報の推定方式による差がほとんど見られない。フェードイ

表1 重み付き動き予測情報の算出にかかる演算数

推定方式	加減算	乗算	除算
参照ソフト(JM)	$2n-2$	0	1
最小二乗法	$5n-2$	$3n+5$	2
提案方式	黒	$2n-2$	0
	白	$2n+1$	1

(回 / フレーム)

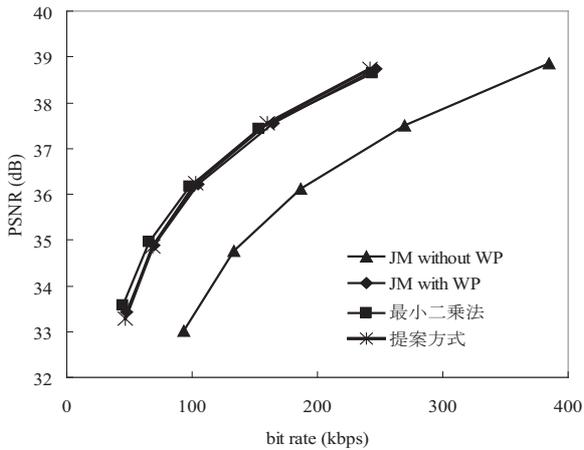


図3 黒へのフェードアウトにおける PSNR

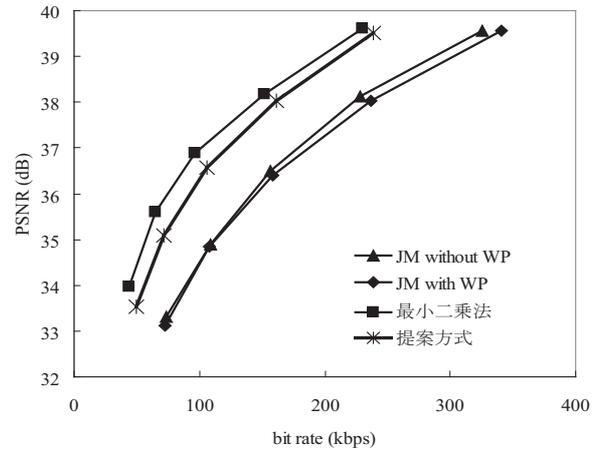


図5 白へのフェードアウトにおける PSNR

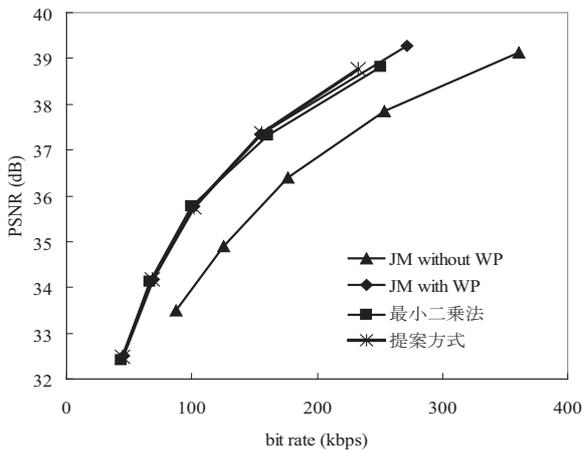


図4 黒からのフェードインにおける PSNR

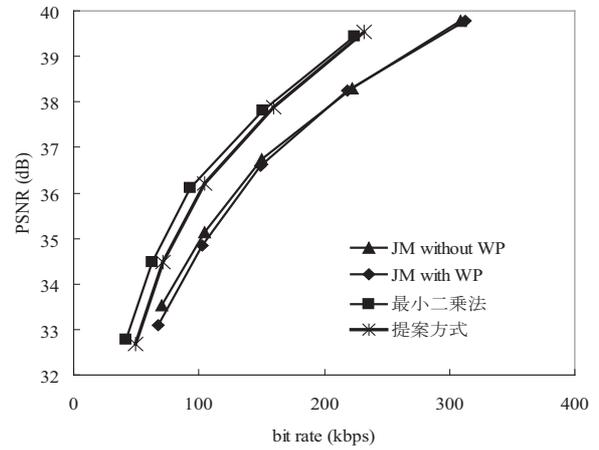


図6 白からのフェードインにおける PSNR

ン・フェードアウトに関わらず、黒フェードの変化が画素値に比例するので重み係数は単純な画素平均比で十分であることが確認できる。一方、白フェードでは推定方式による差が著しい。図5と図6からは、提案方式が参照ソフトと比べて最大2dB向上していることが分かる。最小二乗法による最適解と比べても0.1~0.8dBの差に抑えられている。提案方式の画質改善効果は、画素値に反比例する白フェードの変化を重み係数だけでなくオフセット係数が効果的に補正したためと考えられる。また、参照ソフトで重み付き動き補償の効果が現れていないのは、画素平均比による推定方式が不適切であることを示している。

## 5. 結論

本稿ではフェードの画像合成特性を用いて、フェード種別に応じた重み付き動き予測情報の推定方式を提案した。提案方式は画素変化からフェードの種別を判断するとともに、最適解を近似した重み付き動き予測情報の導出式

において計算量を大幅に削減し、処理負担を軽減した。

検証実験では提案方式が高速かつ高精度に重み付き動き予測情報を推定することを示した。従来の方式と比べて同等の計算量で2dBの画質向上を実現し、最小二乗法を用いた最適解と比較して同程度の画質で4倍以上の高速化を実現できることを確認した。

## 参考文献

- [1] Y. Kikuchi and T. Chujoh, "Improved multiple frame motion compensation using frame interpolation," Joint Video Team of ISO/IEC MPEG and ITU-T VCEG, JVT-B075, 2002.
- [2] K. Lillevold, "Improved direct mode for B pictures in TML," ITU-T Video Coding Experts Group, Q15-K44, August 2000.
- [3] K. Suehring, "H.264 / AVC reference software." [Online]. Available: <http://bs.hhi.de/suehring/tml>