

適応内挿フィルタを用いた複数フレームからのオーバーラップ動き補償

中島 勝己[†] 松田 一朗[†] 伊東 晋[†]

[†] 東京理科大学 理工学部

〒 278-8510 千葉県 野田市 山崎 2641

E-mail: †{nakajima,matsuda,itoh}@itohws01.ee.noda.tus.ac.jp

あらまし 近年、動画像符号化のための動き補償 (MC) 方式として、半画素精度、1/4 画素精度といった高精度な動ベクトルに基づいた手法が一般的に用いられている。これに対し筆者らは、動ベクトルを整数画素精度に限定する代わりに、フレーム毎に設計した複数の内挿フィルタを適応選択することで、高精度な MC 予測を実現する方式を提案している。またこの方式を、動き補償された複数のフレームを同時に参照できるように拡張することで、予測性能が向上することも確認している。本稿では、更にオーバーラップ MC と同様な窓関数による重み付け処理の導入を図った。シミュレーションの結果、上記の重み付け処理によって再生画像の SN 比が最大 1.0 dB 向上し、主観的にもブロック歪の低減効果によって優れた画質が得られることを確認した。

キーワード 動画像符号化, 動き補償予測, 適応内挿フィルタ, オーバーラップ動き補償, 窓関数

Overlapped Block Motion Compensation from Multiple Frames Using Adaptive Interpolation Filters

Masaki NAKAJIMA[†], Ichiro MATSUDA[†], and Susumu ITOH[†]

[†] Faculty of Science and Technology, Tokyo University of Science
2641 Yamazaki, Noda-shi, Chiba 278-8510, Japan

E-mail: †{nakajima,matsuda,itoh}@itohws01.ee.noda.tus.ac.jp

Abstract Currently, motion compensation (MC) based on precise motion vectors, typically with half-pel or quarter-pel accuracy, is commonly used in video coding schemes. On the other hand, we have proposed a novel MC method which employs motion vectors with integer-pel accuracy and adaptive interpolation filters to compensate various motions. Recently, the method has been extended to use motion-compensated multiple frames at once. In this paper, we combine our method with an overlapped block MC technique using a smoothed window function. Simulation results indicate that the proposed method provides a coding gain of up to 1.0 dB in SNR compared with our previous method.

Key words Video coding, Motion compensated prediction, Adaptive interpolation filter, Overlapped block motion compensation, Window function

1. ま え が き

現在、動画像符号化において 1/2 画素、1/4 画素といった非整数画素精度の動ベクトルによる動き補償 (MC) 予測が一般的に用いられている [1]。また、両方向予測や双予測のように、2 枚の参照画像を時間方向に平均することで予測性能を高める手法も多くの方に採用されている [2]。これらの MC 予測技術は、高精度な動き補償を実現するだけでなく、空間方向および時間方向の複数の画素値を加重平均することで、参照画像に含まれる雑音の低減効果も狙ったものである。この技術を発展させた

ものとして、空間方向の重みを画像毎に設計する適応内挿フィルタ (AIF)[3,4]、時間方向の重みを適応化する重み付きフレーム間予測 (Weighted Prediction) [5] といった手法が、近年盛んに検討されている。これらの手法は、主に H.264/AVC [6] の拡張として提案されているため、AIF の割り当ては動ベクトル (1/4 画素精度) の小数点以下の成分によって決まり、Weighted Prediction に用いる参照フレームも一般に 2 枚までといった制約が存在する。

これに対し、我々は動ベクトルの精度を整数画素精度に限定する代わりに、フレーム毎に用意した複数の内挿フィルタをマ

クロブロック (16 × 16 画素) 毎に切り替える手法を提案している [7]. この手法は、フィルタの数や、マスク形状に関する自由度が高く、また、両方向予測のような時間方向の重み付け処理に拡張することも容易である [8]. 更に文献 [9] では、5 枚程度の参照フレームを同時に用いて MC 予測を行う手法を提案し、その効果を確認した。しかし、これらの手法はマクロブロック単位で独立に予測値を生成しているため、ブロック歪による画質低下が生じる場合があった。本稿では、ブロック歪の低減策として、オーバラップ動き補償 [10] と同様な窓関数による重み付け処理 [11] の導入について検討し、主観画質と予測性能の改善を図る。

2. 複数のフレームを用いる適応内挿フィルタ [9]

文献 [9] では、フレーム毎に 16 種類の内挿フィルタ群を用意し、これらを 16 × 16 画素のマクロブロック単位で適応選択している。予測値を生成する際は、図 1 のように連続する N 枚の参照フレームを、マクロブロック毎に与えられた N 本の動ベクトル (整数画素精度) に従って動き補償すると共に、それぞれのフレームから K_n ($n = 1, 2, \dots, N$) 個ずつ抽出した画素群を内挿フィルタに基づいて加重平均する。ここで、マクロブロック b に属する画素 $p \in b$ の予測値 $\hat{f}_b(p)$ は次式により算出される。

$$\hat{f}_b(p) = \sum_{n=1}^N \sum_{k=0}^{K_n-1} a_{m_b}(n, k) \cdot f'_{t-n}(p + v_{n,b} + q_k) \quad (1)$$

但し、 $m_b \in \{1, 2, \dots, 16\}$ は当該マクロブロックにおいて選択された内挿フィルタの番号、 $\{a_m(n, k)\}$ は m 番目のフィルタの係数群、 $f'_{t-n}(p)$ は符号化対象フレーム (時刻 t) から n 枚離れた参照フレームの輝度値 (復号画像) をそれぞれ表している。また、 $v_{n,b}$ はマクロブロック b と参照フレーム $f'_{t-n}(p)$ との間で検出された動ベクトル、 $\{q_k\}$ は各動ベクトルの終点を中心として、図 1 のようにらせん状に配置された画素群の相対座標である。このように、本方式の内挿フィルタは空間方向だけでなく時間方向にも係数 $\{a_{m_b}(n, k)\}$ を配置した 3 次元構造を有している。

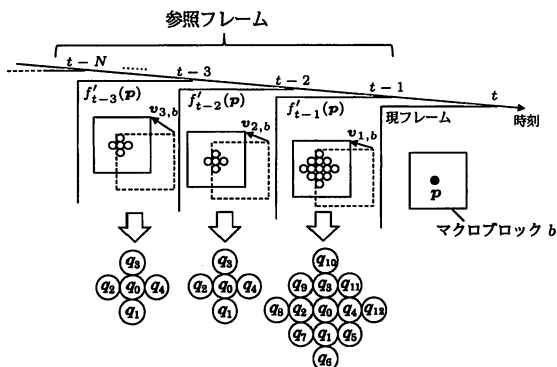


図 1 複数のフレームを用いた予測値の生成

3. 参照フレーム同士のブロックマッチングに基づいた動ベクトルの検出

上述のように、 N 枚の参照フレームに対して MC を適用するには、マクロブロック毎に N 本の動ベクトル $\{v_{n,b} | n = 1, 2, \dots, N\}$ が必要となる。このため、フレーム数 N の増加に伴って動ベクトルに関する付加情報の符号量が増大するという問題が生じる。この問題を解決するため、本方式では直前のフレームに対する動ベクトル $v_{1,b}$ のみを符号化し、残りの動ベクトル $v_{2,b} \sim v_{N,b}$ を復号側に設置した動き検出器を用いて推定する手法 [12] を導入する。具体的には、図 2 のように直前の参照フレーム (時刻 $t-1$) 上に当該マクロブロック b を $v_{1,b}$ だけ平行移動させたブロック b' を設定し、このブロックと $n \geq 2$ 枚目の参照フレームとの間でブロックマッチングを行うことにより動ベクトル $v'_{n,b}$ を検出する。

$$v'_{n,b} = \underset{v}{\operatorname{argmin}} \left\{ \sum_{p \in b'} (f'_{t-1}(p) - f'_{t-n}(p + v))^2 \right\} \quad (2)$$

但し、上記ブロックマッチングの探索範囲を広く設定すると演算量が膨大となるため、予め各参照フレームに存在する符号化済み動ベクトルを用いて $v'_{n,b}$ の探索中心 $\hat{v}'_{n,b}$ を求め、その ± 2 画素の範囲に限定してブロックマッチングを実行する。ここで $\hat{v}'_{2,b}$ は、図 3 のようにブロック b' とオーバラップする 4 つのマクロブロックの符号化済み動ベクトル $v_A \sim v_D$ を、それぞれオーバラップした領域の面積 $S_A \sim S_D$ で加重平均したものである [13]. 同様に $n \geq 3$ 枚目についても、 $v_{1,b} + v'_{n-1,b}$ だけ平行移動したブロック b' と各マクロブロックとのオーバラップ領域の面積比を算出することで、時刻 $t-n+1$ で符号化した動ベクトルから $\hat{v}'_{n,b} - v'_{n-1,b}$ を求めることができる。

最終的に復号側で (1) 式に基づいた予測値生成処理を実行す

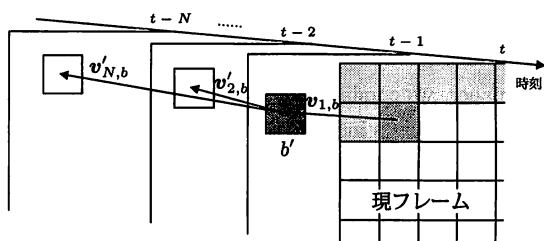


図 2 参照フレーム同士のブロックマッチング

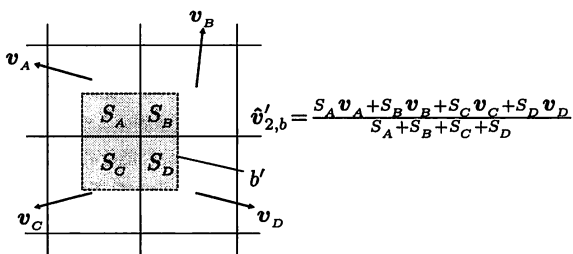


図 3 符号化済み動ベクトルと探索中心の関係

るためには、16種類の内挿フィルタの係数群 $\{a_{m_b}(n, k)\}$ に加え、マクロブロック毎に選択されたフィルタの番号 m_b と動ベクトル $v_{1,b}$ を付加情報として与える必要がある。これらの値は、上記の手順で $v_{n,b} = v_{1,b} + v'_{n,b}$ ($n \geq 2$) が決定されることを前提とし、後述の手順によってフレーム毎に繰り返し最適化している。

4. オーバラップMCの導入

本方式では、MCにおいてマクロブロック単位の平行移動モデルを採用しているため、ブロック境界で予測値が不連続となるブロック歪が発生し、最終的な再生画像の品質を低下させる原因となる。このような動き補償に起因したブロック歪を抑える有効な方法として、オーバラップMCと呼ばれる手法が知られている [10]。これは、隣接マクロブロックを拡大した領域についてそれぞれ求めた予測値を、互いにオーバラップする窓関数で重み付けした後に合成し、連続的な予測画像を生成する手法である。本方式では、窓関数によって重み付けされる前の予測値が、前節で述べたように複数フレームのMCと適応内挿フィルタを用いて生成されていることに注意する。すなわち、当該マクロブロック b_0 に属する画素 $p \in b_0$ の予測値 $\hat{f}_{b_0}(p)$ は次式により算出される。

$$\hat{f}_{b_0}(p) = \sum_{b \in \Omega_{b_0}} w(p - u_b) \cdot \sum_{n=1}^N g_{t-n, m_b}(p + v_{n,b}) \quad (3)$$

$$g_{t-n, m}(p) = \sum_{n=1}^N \sum_{k=0}^{K_n-1} a_m(n, k) \cdot f'_{t-n}(p + q_k) \quad (4)$$

但し、 Ω_{b_0} は b_0 およびその周囲8近傍に位置するマクロブロックの集合、 u_b はマクロブロック b の中心座標、 $g_{t-n, m}(p)$ は n 枚目の参照フレーム $f'_{t-n}(p)$ に対し、 m 番目の内挿フィルタの対応する係数のみを畳み込んで生成された信号を表している。また、 $w(p)$ は原点を中心に設定された2次元窓関数であり、本稿では raised cosine 関数 [14] に基づいた1次元窓関数 $\omega(x)$ を2次元座標 $p = (x, y)$ に拡張したものを採用している。

$$\omega(x) = \begin{cases} 1, & (|x| < 8 - d) \\ \cos^2 \left\{ \frac{(|x| + d - 8)\pi}{4d} \right\}, & (8 - d \leq |x| < \min(8 + d, 16)) \\ 0 & (\text{otherwise}) \end{cases} \quad (5)$$

$$w(p) = \omega(x) \cdot \omega(y) \quad (6)$$

ここで、 d は窓関数の重なりを制御するパラメータであり、本稿ではその値を可変とすることで図5に示すような様々な形状の窓関数 (計19種類) を設計している。但し、(3)式において各マクロブロックの中心にシフトした窓関数の重みの和が1と

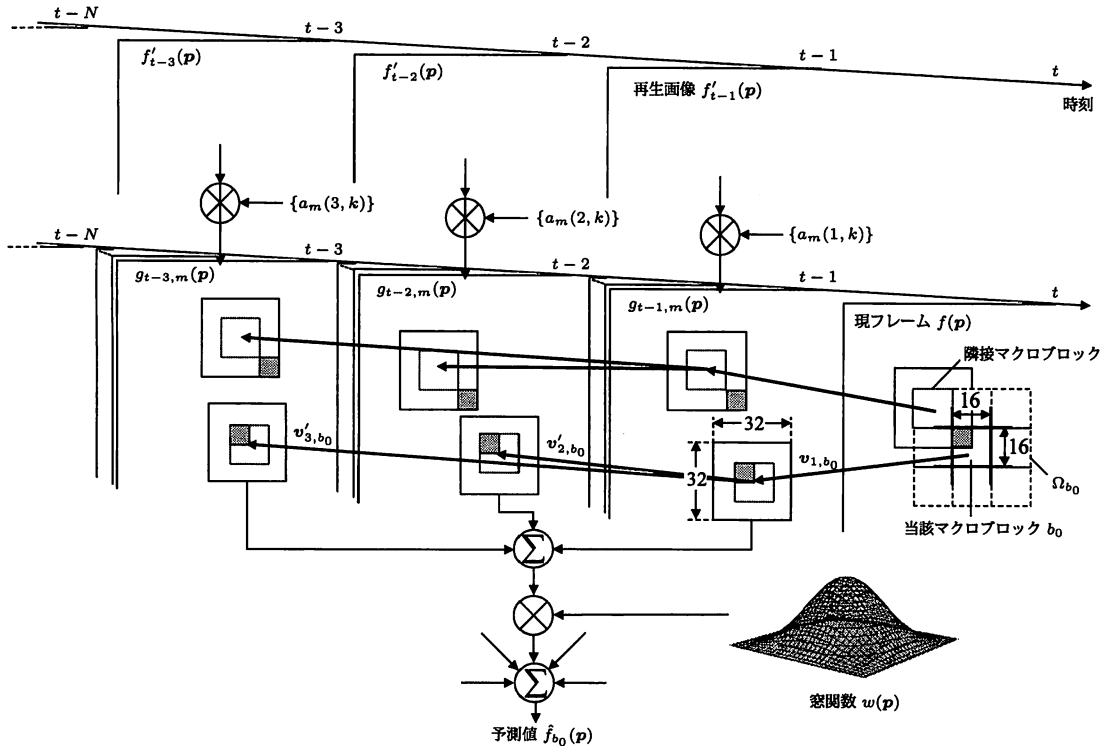


図4 オーバラップ動き補償による予測値の生成手順

なることを保証するため、 d の値はフレーム毎に設定するものとする。また、 $d > 8$ の場合に窓関数の影響が周囲 8 近傍のマクロブロック以外に及ぶことを防ぐため、 $|x| \geq 16$ の範囲においては $\omega(x)$ の値を強制的に 0 とおいている [15]。

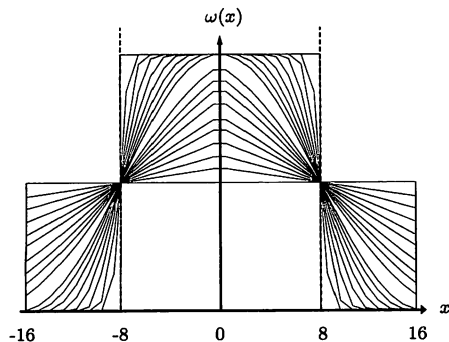


図 5 窓関数 $\omega(x)$

5. パラメータの修正手順

本方式において、予測画像 $\hat{f}_{b_0}(p)$ を生成するために必要なパラメータは以下の通りである。

- マクロブロック毎の動ベクトル ($v_{1,b}$)
- マクロブロック毎の内挿フィルタの選択情報 (m_b)
- 各内挿フィルタの係数 ($a_m(n, k)$)
- 窓関数の形状 (d)

これらのパラメータは、フレーム毎に (7) 式のコスト関数 J の値が最小となるように繰り返し修正され、最終的に付加情報として復号側にダウンロードされる。但し、動ベクトル $v_{2,b} \sim v_{N,b}$ は復号側で検出可能であるため、付加情報としてマクロブロック毎に伝送する動ベクトルは $v_{1,b}$ のみである。

$$J = D + \lambda(R_v + R_m + R_a + R_d) \quad (7)$$

上式における D は予測誤差の 2 乗和、 R_v, R_m, R_a, R_d はそれぞれ上に列挙した 4 種類のパラメータの伝送に要する符号量である。各フレームにおける具体的な修正手順を以下に示す。

- (1) 各マクロブロックに対してブロックマッチング法を適用し、動ベクトル $v_{1,b}$ の初期値を決定する。但し、ブロックマッチングのコスト関数に関しては重み付き 2 乗誤差に基づいた次式を採用する [16]。

$$J'_b = \sum_{p \in \Omega_b} \left\{ w(p - u_b) \cdot (f(p) - f'_{t-1}(p + v_{1,b})) \right\}^2 + \lambda R_v \quad (8)$$

このとき、窓関数 $w(p)$ の形状は前フレームと同一（前フレームがイントラ符号化の場合は $d = 8$ に固定）とする。また、3. で述べた手順に従い残りの動ベクトル $v_{2,b} \sim v_{N,b}$ を検出する。

- (2) マクロブロック毎に前フレームで使用した 16 種類の内挿フィルタをそれぞれ適用し、次式のコスト関数が最小とな

るフィルタ番号 m_b を決定する。

$$J''_b = \sum_{p \in \Omega_b} \left\{ w(p - u_b) \cdot \left(f_t(p) - \sum_{n=1}^N g_{t-n,m}(p + v_{n,b}) \right) \right\}^2 + \lambda R_m \quad (9)$$

但し、前フレームが I ピクチャの場合は、ブロックマッチングによる重み付き 2 乗誤差 ((8) 式の右辺第一項) の大小に基づいて 16 通りにラベル付けした結果を m_b の初期値とする。

- (3) 4. で述べた手順に基づいて予測画像を生成することを前提とし、最小 2 乗法により内挿フィルタの再設計を行う。従来方式 [7] では、同一の内挿フィルタを選択したマクロブロック群で構成される領域毎に、独立に最小 2 乗法を適用することが可能であった。これに対し、本方式ではそれぞれの領域が窓関数によって互いにオーバラップするため、画面全体を対象として 16 組の内挿フィルタを同時に求める必要がある。従って最小 2 乗法を解く際に必要な正規方程式の行列も縦横 16 倍のサイズとなる。

- (4) 上で求めたフィルタ係数を 1/64 精度で量子化すると共に、その符号量 R_a を加味してコスト J の値を評価し、その値が減少したフィルタについてのみ係数 $a_m(n, k)$ を実際に更新する。すなわち、再設計による改善量が小さいフィルタについては、前フレームで使用した係数をそのまま用いることを可能とし、付加情報の増加を抑えている。

- (5) 各マクロブロックについて、動ベクトル $v_{1,b}$ を ± 1 画素の範囲で修正しながら新たな内挿フィルタを各々試し、コスト J を最小とする動ベクトル $v_{1,b} \sim v_{N,b}$ とフィルタ番号 m_b の組み合わせを決定する。但し、 $v_{2,b} \sim v_{N,b}$ は動ベクトル $v_{1,b}$ を修正する度に、改めて検出し直す。

- (6) フレーム全体でコスト J が最小となるよう、窓関数の形状を決めるパラメータ d の値を更新する。

- (7) (3) ~ (6) の手順をコスト J が減少しなくなるまで繰り返す。

6. 特性評価

4 種類のモノクローム標準動画像 (CIF, 30Hz, 30 フレーム分) を対象として符号化シミュレーションを実施した。但し、予測誤差に対しては H.263 [17] に準拠した DCT 符号化 (量子化ステップサイズ Q 一定) を適用し、前節の修正手順に用いるコスト関数のラグランジュ乗数は $\lambda = 0.85 \cdot (Q/2)^2$ と設定した。図 6 は本方式において、参照フレーム数 N を変化した際の $N=1$ に対する符号化レートの削減率 Δ bitrate [18] を示している。このとき、各フレームに配置する係数の数 K_n については、それぞれ指定された N に対して平均的に優れた性能が得られるよう、予備実験に基づいて表 1 のように設定した。図 6 より、画像によって N の最適値は異なるものの、 $N=5$ のときに比較的良好な結果が得られていることがわかる。

図 7 に、各画像について求めた符号化レートと SN 比の関係を示す。ここで、「窓関数なし」は文献 [9] の手法 (本方式におい

て $d=0$ に固定した場合と等価), 「1/4 画素精度」は H.264 [6] と同様な 6 タップフィルタに基づいた 1/4 画素精度の MC 予測 ($N=1$) であり, どちらも窓関数に基づいた重み付け処理は行っていない. 同図より, オーバラップ MC の導入によって, 本方式 ($N=5$) の SN 比が「窓関数なし」($N=5$) に対して同一レートで 0.3~1.0 dB 程度向上している様子を確認できる. また再生画像についても, ブロック境界での動きの連続性に起因したブロック歪が低減しており, 主観画質が向上することを確認している.

N	1	2	3	4	5~7
K_1	25	13	13	13	13
K_2	—	5	5	5	5
K_3	—	—	5	5	5
K_4	—	—	—	5	5
$K_5 \sim K_7$	—	—	—	—	1

7. まとめ

本稿では, 複数フレームを用いた適応内挿フィルタによる動き補償予測手法に, オーバラップ MC と同様な重み付け処理を導入した. シミュレーション実験の結果, 本方式によって再生画像の SN 比を 0.3~1.0 dB 程度改善できることを確認した. また, 再生画像上のブロック歪が低減され, 主観的な画質も向上した.

文 献

- [1] B. Girod, "Motion-Compensating Prediction with Fractional-Pel Accuracy", IEEE Trans. on Communications, Vol. 41, No 4, pp. 604-612, Apr. 1993.
- [2] B. Girod, "Why B-Pictures Work: A Theory of Multi-Hypothesis Motion-Compensated Prediction", Proc. of 1998 International Conf. on Image Processing (ICIP-98), Vol. 2,

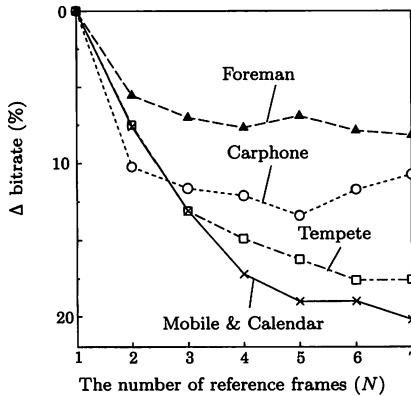


図 6 参照フレームの枚数と Δ bitrate の関係

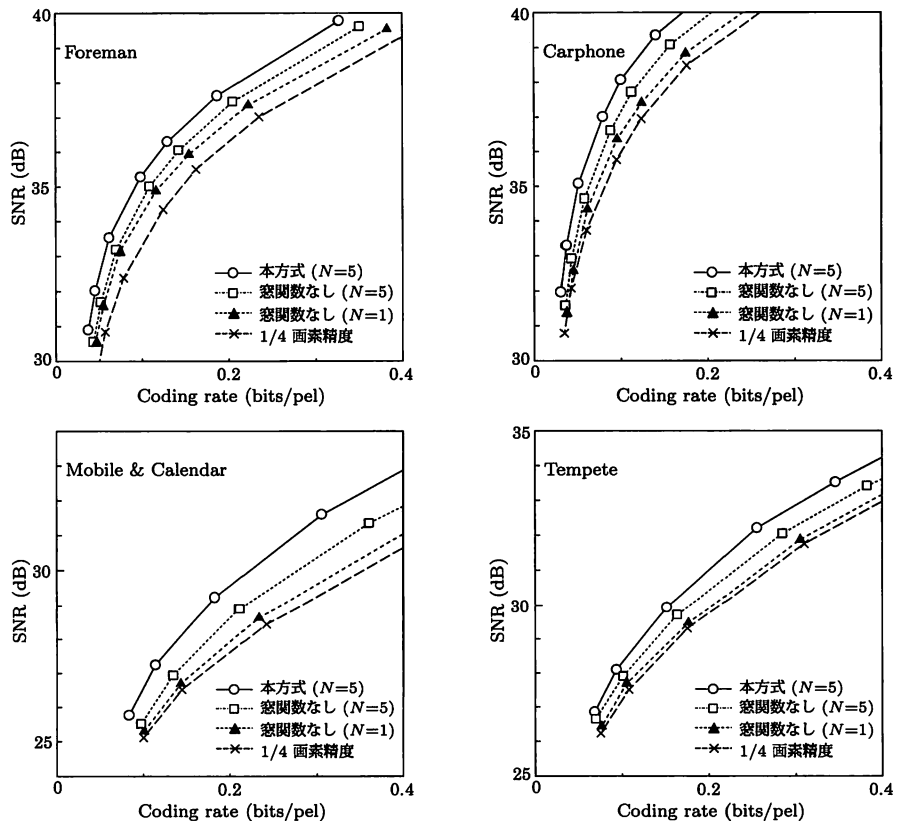


図 7 符号化特性

- pp. 213–217, Oct. 1998.
- [3] Thomas Wedi, “Adaptive Interpolation Filter for Motion and Aliasing Compensated Prediction”, Proc. of SPIE Visual Communications and Image Processing (VCIP-2002), Vol. 4671, pp. 415–422, Jan. 2002.
 - [4] Y. Vatis, B. Edler, D. T. Nguyen and J. Ostermann, “Motion-and Aliasing-Compensated Prediction Using a Two-Dimensional Non-Separable Adaptive Wiener Interpolation Filter”, Proc. of 2005 IEEE International Conf. on Image Processing (ICIP 2005), pp. 894–897, Sep. 2005
 - [5] M. Boyce. “Weighted prediction in the H.264/MPEG AVC video coding standard”, Proc. of 2004 International Symposium on Cir. and Sys.(ISCAS’ 04), Vol. 3, pp. 789–792, May. 2004.
 - [6] ITU-T Rec. H.264, “Advanced Video Coding for Generic Audiovisual Services”, 2003.
 - [7] 長島新一, 松田一朗, 伊東晋: “適応内挿フィルタを用いた非整数画素精度動き補償方式”, 信学技報, Vol. 104, No. 752, pp. 43–48, Mar. 2005.
 - [8] 長島新一, 松田一朗, 森岡一幸, 伊東晋: “適応内挿フィルタを用いた両方向動き補償予測の性能評価”, 第 28 回情報理論とその応用シンポジウム (SITA2005), Vol. 2, pp. 853–868, Nov. 2005.
 - [9] 中島勝己, 高松慶, 松田一朗, 伊東晋: “動き補償された複数フレームを用いた適応内挿フィルタの検討”, 第 7 回情報科学技術フォーラム (FIT2008), Vol. 3, No. I-18, pp. 235–236, Sep. 2008.
 - [10] C. Auyeung, J. Kosmach: “Overlapped Block Motion Compensation”, Proc. of SPIE Visual Communications and Image Processing (VCIP-1992), Vol. 1818, pp. 561–571, Nov. 1992.
 - [11] 荒木雅裕, 長島新一, 松田一朗, 伊東晋: “適応内挿フィルタを用いた非整数画素精度オーバーラップ動き補償”, 映像情報メディア学会年次大会講演予稿集, No. 14-3, Aug. 2005.
 - [12] Y. Suzuki: “Video Encoding Scheme Employing Intra and Inter Prediction Based on Averaged Template Matching Predictors”, IEICE Trans. Vol. E91-D, pp. 1127–1134, Apr. 2008.
 - [13] 清水淳, 嵯峨田淳, 渡辺裕, 小林直樹: “参照フレーム変更によるビットストリームスケールリング”, 信学論, Vol. J83-D-2, No. 3, pp. 967–976, Mar. 2000.
 - [14] J. Katto, J. Ohki, S. Nogaki, M. Ohta: “A Wavelet Codec With Overlapped Motion Compensation for Very Low Bit-Rate Environment”, IEEE Trans. on Cir. Sys. and Video Tech., Vol. 4, No. 3, pp. 328–338, June 1994.
 - [15] 荒木雅裕, 松田一朗, 伊東晋: “オーバーラップ動き補償における窓関数の適応選択”, 情報処理学会研究報告, Vol. 2007, No. 22 pp. 25–30, Mar. 2007.
 - [16] 野垣 他: “オーバーラップ動き補償予測に適した動ベクトル検出方式の検討”, 電子情報通信学会秋季大会, No. D-133, Sep. 1991.
 - [17] ITU-T Rec. H.263: “Video Coding for Low Bitrate Communications”, Nov. 1996.
 - [18] G. Bjontegaard: “Calculation of average PSNR differences between RD-Curves”, ITU-T SG16, Doc. VCEG-M33, Apr. 2001.