

帯域別動き補償を用いた動画像符号化と そのハードウェアによる実現

皆見 利行, 馬場 隆行, 小林 中, 黒木 修隆, 沼 昌宏, 山本 啓輔

神戸大学工学部

〒657-8501 神戸市灘区六甲台町 1-1

E-mail : numa@eedept.kobe-u.ac.jp

あらまし ウェーブレット変換に基づく動画像符号化における動き補償に関して、階層間の相関性から得られる近似ベクトルを用いて第 2 階層以降の動きベクトル探索範囲を限定し、さらに近似ベクトルとの差分を符号化することで、動きベクトル検出演算量と動きベクトル符号量を削減する手法を提案する。本手法を計算機上にて実装して行った実験の結果、従来の全探索法と同等の符号化効率を達成して復号画像の品質を保ちつつ、動きベクトル検出に要する処理時間を約 25 % 削減可能となった。さらに、提案するウェーブレット変換に基づく動画像符号化器 DVC-W をハードウェアによって実現した。実時間処理に必要となる動作周波数について、本手法では全探索法を用いた場合の約 78 % に削減されることを確認した。

キーワード ウェーブレット変換, 動き補償, RLE, 汎用エンジン, FPGA

Sub-Band Video Coding with Motion Compensation and Its Hardware Implementation

Toshiyuki Kaimi, Takayuki Umaba, Naka Kobayashi

Nobutaka Kuroki, Masahiro Numa, Keisuke Yamamoto

Faculty of Engineering, Kobe University

1-1 Rokkodai-cho, Nada, Kobe 657-8501

E-mail : numa@eedept.kobe-u.ac.jp

Abstract We propose a novel sub-band video coding technique based on wavelet transform with motion compensation employing prediction of motion vectors for the higher sub-bands based on those in the lowest sub-band. Correlation between motion vectors for sub-bands is successfully used to shorten the processing time as well as the total code length. Experimental results using software implementation have shown 25% shorter processing time than the conventional full search method. Furthermore, we have implemented the video encoder including the proposed technique, called DVC-W (Digital Video Codec based on Wavelet transform), by hardware using FPGA's and memories. The clock frequency needed for real-time processing has been reduced to 78% by the proposed technique.

Keyword Wavelet Transform, Motion Compensation, Reconfigurable Machine, FPGA

1. はじめに

昨年、第3世代携帯電話規格であるIMT-2000 [1] に基づく携帯電話による動画像送受信サービスが開始された。限られた消費電力でより品質の高い動画像の処理を行うために、動画像符号化技術およびハードウェアによる実現手法が重要となっている。

MPEG [2] などの動画像符号化では離散コサイン変換 (DCT: Discrete Cosine Transform) [3] に基づく符号化 (図 1 (a)) が主に用いられているが、DCT ではブロック単位で符号化を行うために、低ビットレート時の再生画像にブロック歪みが発生する点が問題となっている。そこで、DCT に代わる技術としてサブバンド符号化が期待されており、なかでも離散ウェーブレット変換 (DWT: Discrete Wavelet Transform) [4] を用いた符号化が注目されている。DWT は DCT と異なり、変換をブロック単位で行わないため、ブロック歪みを軽減できる点に特徴をもつ。また、非定常波で表されることが多い自然画像に対しては、周期関数基底を用いる DCT よりも優れた符号化効率を達成できることが知られている [5]-[7]。さらに、帯域間予測を用いたハール・ウェーブレット変換 (PHWT: Predictive Haar Wavelet Transform) [8] のように、高速処理が可能で、かつブロック歪みの少ない再生画像が得られる手法が提案されている。

動画像の符号化効率を向上させるためには、動き補償技術の適用が不可欠であるが、図 1 (b) に示すように、MPEG などの DCT 符号化において適用されている動き補償技術をそのまま DWT 符号化に適用すると、ブロック歪みの発生によって符号化効率の低下を招く。この問題を解決するために、図 1 (c) のように DWT を前提とする帯域別動き補償 [9] が提案されているが、演算量と符号化効率の点で課題を残している。

本稿では、動きベクトル検出演算量と符号化効率の問題となる DWT を前提とする動き補償に関して、DWT 後の階層間における同帯域成分の相関性に着目することで、動きベクトル検出演算量と動きベクトル符号量をともに削減し、処理時間短縮と符号化効率向上を実現する手法を提案す

る。また、提案する動画像符号化を汎用エンジン [10], [11] 上に実現することで、専用ハードウェアのプロトタイプ開発を行う。

2. 帯域別動き補償を用いた動画像符号化

サブバンド符号化の一つである DWT 符号化において、帯域別動き補償における動きベクトル検出演算量および動きベクトル符号量を削減する手法を提案する。

2.1 動き補償における動きベクトル検出手法

サブバンド符号化を前提とする動き補償に関して、次の動きベクトル検出手法が提案されている。

- i) 各サブバンドごとに独立した動きベクトル検出手法 (全探索法) [9]
- ii) 各階層における低域成分 LL の画像に対してのみ動きベクトルを検出する手法 [12]

i) は符号化効率が高い一方で、各階層、各サブバンド毎の処理を行うため、多大な演算量と処理時間を要する。また、各帯域について個別に動きベクトルを扱うため、伝送すべき動きベクトル符号量が増加する。ii) は、各階層で低域成分 LL の画像に対してのみ動きベクトル検出を行い、そこで得られた動きベクトルを他の帯域にも適用する。演算量に関しては i) よりも少なく、よって符号化に要する時間が短い。しかし、各成分の動きベクトルは必ずしも一致しないため、結果的に予測差分信号の絶対値が大きくなり、逆に符号量が増加する可能性がある。

2.2 動きベクトル検出演算量と符号量の削減

帯域別動き補償では、DWT 後に各階層、各帯域ごとに動きベクトル検出 (全探索法) を行う。そのため符号化効率が高いが、その一方で、各階層、各帯域ごとの処理を必要とするため、多大な演算量を要する。さらに、伝送すべき動きベクトル符号量の増大も問題となる。

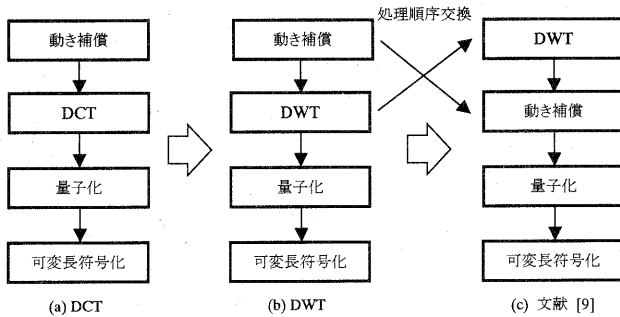


図 1 各符号化手法における処理手順

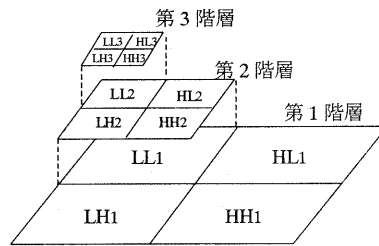


図 2 2次元信号のオクターブ分割

表 1 RM-V の仕様

比較項目	RM-V *
実現可能回路規模	650 K ゲート
搭載する FPGA	EPF10K130V×5
メモリバンク数	56
メモリバンク構成	SRAM : 256 K×16 ビット×6 バンク×4 256 K×32 ビット×1 バンク×4 SDRAM : 4 M×16 ビット×6 バンク×4 4 M×32 ビット×1 バンク×4
FPGA 間の配線	固定配線部分と FPGA により変更可能な部分
FPGA-メモリ間	固定配線部分と 他のモジュールから共有可能な部分

* 4 モジュール構成時

そこで DWT 後における階層間の同帯域成分の親子関係に注目する。例えば図 2 における (LL1, LL2, LL3) のように、階層間の同帯域成分は一般に相関性が高い。このことから、LL1 成分の動きベクトルと LL2 成分の動きベクトルについて、長さは異なるものの、方向は類似している可能性が高い。LL1 成分と LL3 成分との間も同様に考えられる。

このように相関性が高い画像それぞれに対して、同様の動き補償を行うため、第 n 階層 ($n = 2, 3, 4, \dots$) において検出した動きベクトル $mv_n = (vx_n, vy_n)$ は、第 1 階層で検出した動きベクトルを $mv_1 = (vx_1, vy_1)$ と近似係数 r_n を用いて、

$$vx_n \cong vx_1 / r_n \quad (1)$$

$$vy_n \cong vy_1 / r_n \quad (2)$$

と近似できることが多い。この動きベクトルを近似ベクトルと呼ぶ。ただし、DWT では、上で示したように、第 $n+1$ 階層の画像は、第 n 階層の画像に対して X, Y 方向ともに 1:2 の間引きを行った結果であるから、近似係数 r_n を、

$$r_n = 2^{n-1} \quad (n = 2, 3, 4, \dots) \quad (3)$$

で与える。この近似係数を利用して、動きベクトル検出に要する演算量を削減する。

第 1 階層の動きベクトル検出には全探索法を適用する。第 n 階層における動きベクトル検出は、式 (1), (2) 右辺に示す近似ベクトルを基準として、全探索法で適用する探索範囲よりも狭い範囲で動きベクトル検出を行うことで検出演算量を削減する。ここで、近似ベクトルとの差分を $\Delta mv_n = (\Delta vx_n, \Delta vy_n)$ とすると、第 n 階層の動きベクトル $mv_n = (vx_n, vy_n)$ は、

$$vx_n = vx_1 / r_n + \Delta vx_n \quad (4)$$

$$vy_n = vy_1 / r_n + \Delta vy_n \quad (5)$$

と表せる。以上の処理をすべての帯域成分において適用する。この結果、全探索法に近い動きベクトル検出精度を実現しつつ、動きベクトル検出演算量を削減できる。

また、提案手法では、第 2 階層以降の動きベクトル検出は、近似ベクトルを基準とし、全探索法よりも狭い探索範囲で動きベクトルを検出している。そして、第 2 階層以降の動きベクトルについては、式 (4), (5) に基づき生成している。すなわち、 mv_1 の情報と近似に対する差分 Δmv_n の情報があれば、第 2 階層以降の動きベクトルを生成することが可能である。そこで、第 1 階層の動きベクトルデータはそのまま符号化し、第 2 階層以降は、差分である Δmv_n の符号化を行うことで、動きベクトル符号量を削減する。

3. DVC-W のハードウェア化

DVC-W (Digital Video Codec based on Wavelet transform) は、PHWT に基づく動画像符号化システムであ

る。本節では、2 節で提案した手法を含む帯域別動き補償を用いた DVC-W をハードウェアにより実現した、DVC-W エンジンについて述べる。* DVC-W エンジンは、モジュールの協調並列動作とパイプライン処理を適用することで、低い動作周波数で実時間処理を可能とする。

3.1 汎用エンジン RM-V

DVC-W エンジンのプロトタイプ開発には、汎用エンジン RM-V (Reconfigurable Machine-V) [11] を用いる。表 1 に RM-V の仕様を示す。汎用エンジンは、高速メモリである EAB (Embedded Array Block) 内蔵の FPGA (Field Programmable Gate Array) とメモリから構成されており、専用ハードウェアのプロトタイプを構築するためのプラットフォームとしての有効性が示されている [10], [11]。

3.2 DVC-W エンジンの構成

DVC-W の符号化部を構成する、PHWT 部、動き補償部、量子化部、RLE 部の 4 モジュールについてハードウェア化の対象とした。DVC-W の実時間処理を可能にする DVC-W エンジンを実現するため、以下の特徴をもつプロトタイプ構築を目的とする。

- i) ハードウェアの特徴を利用したパイプライン化による高速処理
- ii) DVC-W を機能単位でモジュール化し、機能や仕様の変更に応じてモジュールを追加、削除、交換することが容易なモジュール間インターフェースの実現
- iii) SRAM や EAB の複数バンクへの並列アクセスによるスループットの向上
- iv) 分割した各モジュールの協調並列動作による待機時間短縮

DVC-W エンジンのハードウェア構成を図 3 に示す。RM-V の 4 枚のモジュールボードにそれぞれ、PHWT モジ

ール、動き補償モジュール、量子化モジュール、RLE モジュールを構築することによって、DVC-W エンジンを実現する。

RM-V の各モジュールに搭載される SRAM, FPGA が内蔵する高速メモリである EAB を、一時的に画素値を保持するバッファメモリとして利用し、複数のモジュールから相互にアクセスする構成とする。SRAM は、フレーム単位の画素値データを保持するバッファメモリとして主に利用する。4 つのモジュールが協調して動作することにより、DVC-W エンジンのモジュール単位での並列処理が可能となる。

以下、各モジュールについて述べる。

3.2.1 PHWT モジュール

PHWT モジュールは、ホスト・コンピュータから転送された YUV 変換後のデータを受け取り、帯域間予測を用いたハール・ウェーブレット変換 (PHWT) の結果を、動き補償モジュールに転送する。PHWT では最低でも 1 フレーム分のフレームバッファをもつ必要があり、さらにランダムアクセス性も要求される。よって、入力フレームバッファには PHWT モジュール上の SRAM を、出力フレームバッファには動き補償モジュール上の SRAM を用いる。PHWT モジュール内では、パイプライン処理および並列処理を用いた高速化手法が適用されている。

3.2.2 量子化モジュール

量子化モジュールでは、動き補償モジュールによって出力されたデータに対して、量子化係数テーブルを参照して整数除算による量子化処理を行う。同時アクセス可能なデータ数を増やすことにより、並列処理による高速化を実現している。また、条件分岐などの処理も不要であるため、パイプライン化によって高いスループットを得ている。具体的に

は、パイプライン段数が 17 である整数除算器を用いて、1 クロックごとに 1 回の整数除算を実行している。さらに、この量子化パイプラインを 2 本搭載しており、1 クロックあたり 2 画素分のデータの量子化を行っている。

3.2.3 動き補償モジュール

帯域別動き補償は、主にブロックマッチングによる動きベクトル検出処理と、動きベクトルを用いた予測差分信号の生成処理からなる。それぞれ、パイプライン処理と負荷分散処理を用いて高速化を図る。

画像の各部分の動きに対応するため、ブロック単位で動き補償処理を行うが、処理時間の大半を図 4 に示すブロックマッチング処理に要する。このブロックマッチング処理においては、前フレーム f_{n-1} のブロックを上下左右に移動させ、現フレーム f_n との差分の総和

$$D_{\text{sum}} = \sum_x \sum_y |f_n(x_i, y_j) - f_{n-1}(x_i + dx, y_j + dy)| \quad (6)$$

が最小となる方向 (dx, dy) を動きベクトルとして採用する。DCT を考慮した動き補償では、一般にブロックの大きさは 16×16 画素が用いられるが、サブバンド符号化を前提とする動き補償では、階層ごとにブロックの大きさが変化する。

ブロックマッチング処理では、各階層の処理においてパイプライン処理による高速化を図る。PHWT 後における各階層の LL, HL, HL, HH 成分データは、動き補償モジュール上の別々の SRAM に格納されているため、各成分に対して並列処理を行っている。予測差分信号生成処理においても、動きベクトル検出処理前と処理後のメモリ割付けが変化しないため、各クロックで各成分から 1 画素ずつ読み込んで処理する 4 並列処理を適用することで高速化を図っている。

3.2.4 RLE モジュール

RLE モジュールでは、量子化されたデータを可変長符号に置き換えることでデータを圧縮する。8×8 画素のブロック単位で処理が行われ、ブロック内の画素情報をあらかじめ設定された操作順序に従って取り出し、取り出した画素情報によって次に示

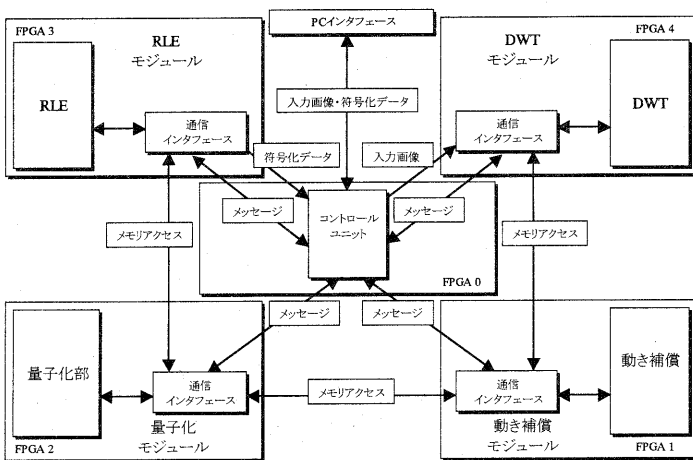


図 3 DVC-W エンジンのハードウェア構成

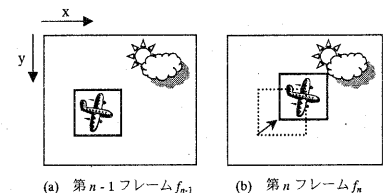


図 4 ブロックマッチング

すように符号化を行う。

i) 取り出す画素情報がゼロでない場合
その画素情報をシンボルとする符号化を行う。

ii) 取り出す画素情報がゼロである場合
次にゼロでない画素情報が取り出された段階で、ゼロの連続個数と非ゼロの値をシンボルとして符号化を行う。その際にゼロを 16 個単位で区切り、それぞれをシンボルとする符号化と、残るゼロの連続個数と非ゼロの値をシンボルとする符号化を行う。ブロックの最後の画素情報までゼロが連続する場合は、最後のゼロでない画素情報で符号化を終了し、End Block を示すシンボルを付加する。

4. 実験評価

4.1 提案アルゴリズムの評価

提案手法を含む DVC-W を計算機上に C 言語で実現し、AthlonXP1800+(1.53GHz)の PC 上で実験を行った。第 n 階層での探索範囲を $\pm s_n$ で表すと、全探索法ではすべての階層において X, Y 方向ともに $s = \pm 8$ とした。提案手法について、第 1 階層では全探索法と同様に $s_1 = \pm 8$ とするが、第 2 階層では予測された動きベクトルを用いて $s_2 = \pm 2$ に、第 3 階層では $s_3 = \pm 1$ に限定した。また、いずれの手法についても探索ピッチは 1 とした。動画データ “Carphone” に対する全体符号化時間と符号量の比較を表 2 に示す。表中の「MC あり/なし」は、動き補償を適用した場合と適用しない場合を示す。

表 2 より、符号化に要する時間は「MC なし」の方が短く、画像符号量に関しては「MC あり」の方が少ないことが確認できる。ここで、全探索法と提案手法の結果を比較する。提案手法を用いると全探索法に比べて符号化時間を約 33 % 削減でき、かつ画像符号量がほぼ等しくなっている。画像符号量においては、一般に全探索法より狭い探索範囲で動きベクトルを検出すると、画像の圧縮効率が低下するが、提案手法では階層間の相関性から得られる近似ベクトルを活用しているため、効率的な動きベクトル検出が実現できたと考えられる。全探索法はすべての動きベクトルをそのまま符号化していることから、動きベクトル符号量が增大している。それに対して提案手法を用いると、全探索法に対して動きベクトル符号量を 12 % 削減できている。

次に、復号画像の PSNR の測定により、提案手法の画質評価を行った。ただし PSNR は MSE (平均ノイズ電力) を用いて

$$\text{PSNR} = 10 \log_{10} \left(\frac{255^2}{\text{MSE}} \right) \quad (7)$$

で定義される数値であり、原画像にどれだけ近い画像であるかを測定する基準である。

表 2 ソフトウェアによる符号化時間と符号量 (Carphone)

評価項目		手法	MC あり	
			全探索法	提案手法
全体符号化時間 (s)		2.64	29.45	19.64
符号量 (KB)	画像	378	206	209
	動きベクトル		71	63
	合計	378	277	272
	比 (MC あり / MC なし)		0.73	0.72

圧縮前の “Carphone” のデータ量: 6,683 KB

“Carphone” に対する全探索法と提案手法の PSNR を図 5 に示す。図より全探索法による復号画像と提案手法による復号画像の PSNR がほぼ一致することが確認できる。また比較的動きの激しい “Football” に関しても、“Carphone” とほぼ同様の結果が得られることを確認している。

以上のことから、提案手法によって全探索法とほぼ同等の動きベクトル検出精度を実現しつつ、符号化時間を削減できることが確認できた。

4.2 DVC-W エンジンに対する実験と考察

提案手法を含む DVC-W をハードウェア化した DVC-W エンジンに RM-V [11] 上に実現した。動き補償モジュールは 4 帯域の処理を同時に並列で行うため、RM-V 上に実装した際、FPGA に実現できる回路規模の問題から一つのモジュールボードでの実装が困難であった。そこで、動き補償モジュールと動き補償を除く DVC-W エンジンに分けて性能評価実験を行った。ハードウェアの処理性能、およびソフトウェアとの処理時間の比較を表 3 に示す。

動き補償モジュールでは、最高動作周波数が 19MHz となり、ソフトウェアに対して、1.8 倍の高速化を実現している。ソフトウェアの約 1/80 の動作周波数で 1.8 倍の高速化効果が得られている。

動き補償を除く DVC-W では動作周波数が 6.5MHz となり、ソフトウェアの約 1/230 の動作周波数で約 2.5 倍の高速化を実現している。複数のモジュールを利用するために、モジュール間遅延の影響から動き補償モジュールよりも動作

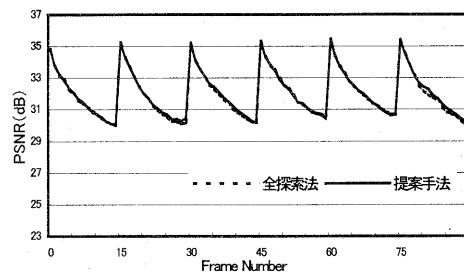


図 5 各フレームの PSNR の比較 (Carphone)

表 3 ハードウェアの処理性能

モジュール	動き補償	動き補償を除く DVC-W
総クロック数	1,888,734	171,136
動作周波数	19.0 MHz	6.5 MHz
1フレームあたりの処理時間	99.4 ms	26.3 ms
ソフトウェアによる処理時間*	180 ms	65.1 ms

*AthlonXP1800+(1.53GHz), メモリ 512 MB

周波数が低下するが、LSI 化による高速動作が期待される。ソフトウェアの約 1/230 の動作周波数で 2.5 倍の高速化効果が得られたため、低消費電力効果が期待できる。

また、本システムは 4 モジュールが並列協調動作するため、システム全体の処理速度は動き補償モジュールに依存する。実時間処理に必要な動作周波数は、全探索法では 73.3MHz、提案手法では 57.2MHz となり、約 22 % 低く設定可能となった。この点でも提案手法による演算量削減効果が確認でき、LSI 化する際の低消費電力化が期待できる。

5. まとめ

帯域別動き補償を用いた動画符号化において、動きベクトル検出演算量と符号量の削減手法を提案し、汎用エンジン RM-V 上のハードウェアとして DVC-W エンジンを実現した。提案手法により、実時間処理に必要な動作周波数を、全探索法と比べて約 22 % 低く設定できることが確認され、LSI として実現する際の低消費電力化が期待できることを示した。

謝辞

RM-V のモジュールおよびベースボードのレイアウト設計と実装をご担当下さった株式会社写真化学システム機器事業部電子デバイス部(現在、株式会社 SK エレクトロニクス サークキットボード事業部)、水尾学課長を初めとする皆様に感謝致します。

FPGA のアプリケーション開発に不可欠な、統合開発環境 MAX+PLUS II システムをご提供下さいました、日本アルテラ株式会社サードパーティ・ビジネス・リレーションズ 浮谷光明課長に感謝致します。

参考文献

- [1] 杉浦彰彦, 次世代携帯電話 IMT2000 の基礎技術と標準規格, 日本工業技術センター.
- [2] D. Le Gall, "Mpeg: A video compression standard

for multimedia applications," Commun. ACM, vol. 34.

- [3] N. Ahmed, T. Natarajan and K. R. Rao, "Discrete Cosine Transform," IEEE Trans. Comput., C-23, pp. 90-93, 1974.
- [4] O. Rioul and M. Vetterli, "Wavelet and Signal Processing," IEEE Sig. Proc. Magazine, 8, 4, pp. 14-38, 1991.
- [5] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using the wavelet transform," IEEE Trans. Image Processing, vol. 2, pp. 205-220, Apr. 1992.
- [6] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," IEEE Trans. Signal Processing, vol. 41, pp. 3445-3462, Dec. 1993.
- [7] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning of hierarchical trees," IEEE Trans. , CAS for Video Technology, vol. 6, pp. 243-250, Jun. 1996.
- [8] 黒木修隆, 野村孝徳, 亀田昌志, 沼昌宏, 平野浩太郎, "帯域間予測を用いたハール・ウェーブレット変換と画像符号化への応用", 電子情報通信学会論文誌 Vol. J77-A No. 12, pp. 1738-1746, 1994 年 12 月.
- [9] H. Gharavi, "Subband coding algorithm for video applications : videophone to HDTV conferencing," IEEE Trans. , CAS for Video Technology, Vol. 1, No. 2, pp. 174-182, June 1991.
- [10] 沼昌宏, "汎用エンジン RM-I~IV と応用例", 電子情報通信学会技術研究報告, VLD96-80, CPSY96-92, pp.119-126, 1996 年 12 月.
- [11] 山口卓也, 安西伸介, 水谷 敦, 黒木修隆, 沼昌宏, "汎用エンジン RM-V とその応用", 電気学会論文誌 C, vol. 120-C, no. 11, pp. 1629-1636, 2000 年 11 月.
- [12] X. Yang, "Scalable wavelet video coding using aliasing-reduced hierarchical motion compensation," IEEE Trans. , Image Processing, Vol. 9, No. 5, pp. 778-791, May 2000.