

マルチステップ中断法動きベクトル検出アルゴリズムとこれを適用した低消費電力 CMOS 差分絶対値和回路の開発

長谷川祥太 榎本忠儀

中央大学 大学院 理工学研究科 情報工学専攻
〒112-8551 東京都文京区春日 1-13-27

あらまし マルチステップ中断法 (MS-BOS)、動きベクトル検出アルゴリズムは複数個 (n) の異なるサイズの探索領域 (SW) (MS-BOS)を用いて、SW 毎にブロックマッチング (BM)する手法である。本論文では、高速性と高画質を両立できる SW サイズと最適な n 値を検討する。従来の全探索法 (FS) の SW サイズは $\{(2p+16)画素 \times (2p+16)ライン\}$ である。これに対して MS-BOS の n ステップ目の SW サイズは $\{(2p'+16)画素 \times (2p'+16)ライン\}$ である。動きの遅い動画の場合、 p' が 2、 n が 2 の時、FS の画質を維持して、動きベクトル検出速度を FS を適用した時の 63 倍以上に高速化できた。一方、動きの速い動画の場合、 p' が 5、 n が 3 の時、FS 並みの画質で、検出速度を FS を適用した時の 6 倍以上に高速化できた。本アルゴリズムと 0.18- μ m CMOS 技術を用い、QCIF 用カウンタ形差分絶対値和回路を設計した。本回路の消費電力は 781.8 μ W (クロック周波数 260 MHz、電源電圧 1.8 V)であった。本回路にゲーテッドクロックを適用すると、動きの速い動画、動きの遅い動画、の消費電力、 P' はそれぞれ 131.1 μ W ($p'=5$)、12.3 μ W ($p'=2$)となり、それぞれ FS の約 16.8%、1.6%に削減された。

キーワード 動きベクトル検出、消費電力、動画、中断法、マルチステップ中断法、差分絶対値和回路

"Multi-Step Breaking-off-Search (MS-BOS)" Motion Estimation Algorithm and Low-Power Absolute Difference Accumulators for H.263 Encoding

Shota Hasegawa and Tadayoshi Enomoto

Graduate School of Science and Engineering, Chuo University
1-13-27 Kasuga, Bunkyo-ku, Tokyo 112-8551, Japan

Abstract: A fast-motion-estimation (ME) algorithm called a "Multi-step Breaking-Off-Search (MS-BOS)" method was developed. Sizes of the search areas were optimized to improve block-matching speed while maintaining picture quality. It can improve processing speed of the full-search (FS) method by a factor of 6 to 63, depending on picture types, while maintaining visual quality of the full-search method. At clock frequency of 260 MHz and supply voltage of 1.8 V the power dissipation of a 0.18- μ m CMOS accumulation type motion estimator using MS-BOS and gated-clock technique was reduced to 12.3 μ W which was only 1.6% that using FS and gated-clock technique.

key words: CMOS, motion-estimation algorithm, "Multi-step breaking-off-search", H.263, full-search, absolute difference accumulators

1 はじめに

携帯 TV 電話等の携帯機器へ搭載する動画画像符号化 LSI の低消費電力化は必須である。特に動きベクトル検出 (Motion vector Estimation; ME) の高速化が重要な課題である。

全探索法 (Full Search; FS) は探索領域 (Search Window; SW) 内の全マクロブロック (M-Blk) に対して差分絶対値和、 D_s 、を求めるため、高精度に動きベクトルを検出できるが、ブロックマッチング数、 N_b 、が極めて多い。FS の画質を維持したまま、 N_b を大幅に削減できる ME アルゴリズムとして我々はこれまでに中断法 (Breaking Off Search; BOS) [1]、2 ステップ中断法 [2]、M-Blk 毎に最適な ME 停止条件を適応的に設定できる新たな ME 法 (Breaking Off Search Adaptively minimizing number of block matchings ;

BOSA) [3]、他を開発してきた BOS では SW の中心から外に向けて渦巻き状に探索をすると、効率よく動きベクトルが検出できることがわかっている。これは、最小 D_s を持つ最適 MB が、SW の中心付近で見つかることが多いためである。また、渦巻き探索を用いると、動きベクトルの約 50 %が SW の中心付近の 1/2.5 から 1/3 の狭い領域で見つかることがわかっている [1]。そこで、FS や BOS が用いる SW サイズより狭い SW サイズで第 1 ステップ目のブロックマッチング (BM) (ステップ 1) を行う。次に、ステップ 1 で検出された動きベクトルを中心に、やはり狭い SW で第 2 ステップ目の BM を行う。以後同様な BM を継続することにより、つまり、最終的には異なる狭い SW で n 回 BM ($n=1, 2, 3, \dots$) を実行することにより、最適な動きベクトルを検出する。以下では、これをマルチステップ中断法 {Multi-Step Breaking off Search; MS-BOS} と呼ぶ。上

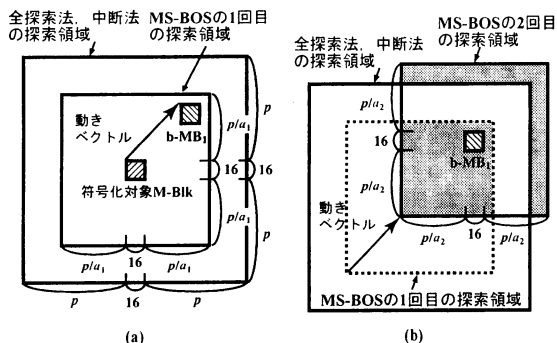


図 2.1 マルチステップ中断法の探索手順。(a) ステップ 1。(b) ステップ 2。

述したように複数回の動きベクトル検出はいずれも狭い探索領域で行われるので、処理量の削減が期待でき、さらに FS と同様に、通常の水平スキャン方式を用いることができる。従って、BOS が用いていた渦巻きスキャン方式を避けることができる。渦巻きスキャン方式は ME 回路への画素データ供給が複雑になるだけでなく、ME 回路を大型化させ、さらに、回路の消費電力、 P を増加させてしまう [2]。

以下、第 2 節で MS-BOS を詳細に述べる。第 3、4 節では、FS 並の平均最小 D_s 、平均ピーク S/N 比、平均 R_{sn} を維持し、処理量、平均 N_b を大幅に削減できる最適な SW サイズと BM ステップ数、 n を検討する。

第 5 節では、MS-BOS とゲートドクロックを採用することにより ME 回路の基本部分である差分絶対値和回路 (Absolute Difference Accumulators) の低電力化を述べる。

2 マルチステップ中断法 (MS-BOS)

マルチステップ中断法 (MS-BOS) は FS および BOS の SW サイズ $\{(2p+16)\text{画素} \times (2p+16)\text{ライン}\}$ (以下ではこれを p とする) より狭い SW $\{(2p/a_n+16)\text{画素} \times (2p/a_n+16)\text{ライン}\}$ (以下ではこれを p' ($=p/a_n$) とする) で、第 n ステップ目 (ステップ n) の探索を行なう。ここで、 n は 1、2、3、 $\dots$ 、 p' ($=p/a_n$) は整数 (割り切れない場合は切り捨てる) である。ステップ n の SW 位置はステップ $(n-1)$ で求めた動きベクトルより決定され、BM は SW 内の全 M-Blk に対して BM を実行する FS あるいは BOS [1] を用いて行なわれる。

図 2.1 に MS-BOS の探索手順を示す。図 2.1(a) に示すように、符号化対象 M-Blk を中心に SW $\{(2p/a_1+16)\text{画素} \times (2p/a_1+16)\text{ライン}\}$ を設定し ($a_1 > 1$)、ステップ 1 の探索を行ない、最小 D_s (D_1) を持つ b-MB₁ を得る。

次に、b-MB₁ を中心に、ステップ 2 の SW $\{(2p/a_2+16)\text{画素} \times (2p/a_2+16)\text{ライン}\}$ を設定し ($a_2 > 1$)、ステップ 2 の探索を行なう (図 2.1(b))。ここで、FS および BOS の SW (p) を超える部分やステップ 1 の SW (p') と重なる部分では BM 処理を行なわない。なお、ステップ 2 の探索で得られた最適 MB (b-MB₂) の D_s (D_2) が D_1 より大きい、あるいは等しい場合は、b-MB₁ を最適 M-Blk とし探索を終える。一方、 D_2 が D_1 より小さい場合、 D_2 よりさらに小さな D_s を持つ M-Blk が存在する可能性があるため、ステップ 2 の探索と同様に、ステップ 3 の探索を行なう。以後、最小 D_s を持つ M-Blk を求めるため、ステップ 3 と同様な探索を継続し、最適 M-Blk を確定する。なお、各ステップでは整数画素精度で BM される。

以下では、 n を 2、3 に設定する (つまり、ステップが 2、3) MS-BOS をそれぞれ 2S-BOS、3S-BOS と呼ぶ。また、各ステップを FS で処理する 2S-BOS、3S-BOS をそれぞれ 2FF、3FFF と呼び、各ステップを BOS で処理する 2S-BOS、

3S-BOS をそれぞれ 2II、3III とよぶ。

3. 動きの遅い動画の解析

3.1 解析画像と解析条件

本節で用いるテスト画像は "Claire" と呼ばれる動きの遅い (平均最小 D_s が小さい) 動画で、フレームサイズは QCIF (176 画素 $\times$ 144 ライン)、フレーム数は 492、である。以下では、P ピクチャに対して H.263 に準拠したソフトウェアエンコーダで解析し、データレート、 R_d 、を 64 kbps、フレームレート、 R_f を 15 フレーム/秒に設定した。なお、3.3 節を除いては、半画素精度の BM 処理を割愛している。

3.2 検出速度と画質

図 3.1 に平均 N_b 、平均検出速度、平均 F_s ($\propto 1/N_b$) と SW サイズの関係を示す。 p は全探索法 (FS) 中断法 (BOS) の SW サイズで、MS-BOS の SW サイズ、 p' は p/a_n ($a_n=2$) である。MS-BOS の平均 N_b は、いずれの場合も、FS のそれより小さく、2II、3III が約 1/8.5、2FF、3FFF が 1/3.5 以下である。つまり、2II、3III の平均 F_s は全探索法のそれより 8.5 倍以上、2FF、3FFF の平均 F_s は、FS のそれより 3.6 倍以上、速いことになる。

図 3.2 に平均最小 D_s 、平均ピーク S/N 比 (平均 R_{sn}) と SW サイズの関係を示す。FS、BOS およびいずれの MS-BOS もその平均最小 D_s は 416 から 424 の範囲にあり、アルゴリズム間の差はほとんど見られない。同様に、FS、BOS、およびいずれの MS-BOS もその平均 R_{sn} は 40.0dB から 40.15dB の範囲にあり、アルゴリズム間の差は小さい。動きの遅い動画に対して、 a_n が 2 の時、いずれのアルゴリズムを用いても画質に差がほとんど見られないことがわかる。以上の結果を表 3.1 にまとめる。

3.3 SW サイズの最適化

前述した 4 種類の MS-BOS に対して、平均 N_b 、平均最小 D_s 、平均 R_{sn} と a_n の関係を検討した。このうち、2II を用いて得た解析結果を図 3.3 (a)、(b)、(c) に示す。本解析では、整数画素精度で求めた最適 M-Blk の周囲を半画素精度で 8 回 BM する階層探索が用いられている。但し、半画素精度の BM 数 (8 回) は平均 BM 数、平均 N_b に含めていない。 a_n の増加に伴い、平均 N_b はわずかながら減少している。 a_1 が 1.2、 a_2 が 1 の時、平均 N_b は 160 で最大となり、平均 F_s は平均 N_b が 684 の全探索法の 4 倍である。一方、 a_1 、 a_2 がともに 5 の時、平均 N_b は 8 で最小となり、平均 F_s は FS より 86 倍速いのみならず、最も速いとされる 3 ステップ法 ($N_b = 25$) [4] よりも 2.5 倍以上速いことがわかる。平均最小 D_s は、 a_n の値にかかわらず、410 あるいは 410 をわずかに上回る程度であり、FS の平均最小 D_s (409) とほぼ等しい。これは 2II で得た動き補償画像の品質が FS のそれとほとんど変わらないことを示している。平均 R_{sn} は、平均最小 D_s と同様に、 a_1 、 a_2 の値にかかわらず、約 41.7dB で、FS の平均 R_{sn} (41.76dB) とほぼ等しい。これは 2II の再生画像が FS のそれと遜色のない画質を保つことを示している。図 3.4 (a)、(b)、(c) に FS、2II ($a_n=2$)、2II ($a_n=5$) で得られた復号画像を示す。画質の優劣は見えない。

3.4 ステップ数 (n) の最適化

表 3.2 はステップ 1 の平均 N_b (N_{b1})、ステップ 2 の平均 N_b (N_{b2})、ステップ 3 の平均 N_b (N_{b3})、平均最小 D_s 、および平均 R_{sn} と探索領域サイズ $\{(2p'+16)\text{画素} \times (2p'+16)\text{ライン}\}$ の関係である。 p' は 2、3、4、7 である。2S-BOS の平均最小 D_s 、平均 R_{sn} をそれぞれ平均最小 D_{s2} 、平均 R_{sn2} 、同様に、3S-BOS の平均最小 D_s 、平均 R_{sn} をそれぞれ D_{s3} 、 R_{sn3} と定義する。表 3.3 にステップ 1、ステップ 2 およびステップ 3 の BM 時に BM を実行した 1 フレーム当たりの M-Blk 数の

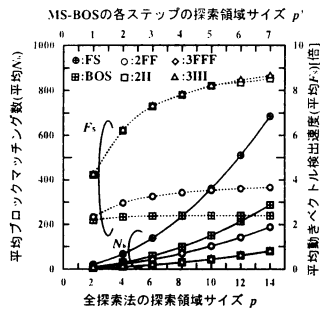


図 3.1 平均 N_b 、平均 F と FS の探索領域サイズ p の関係
("Claire", 64 kbps, 15 fps, $p=14$ 画素, $a_s=2$)

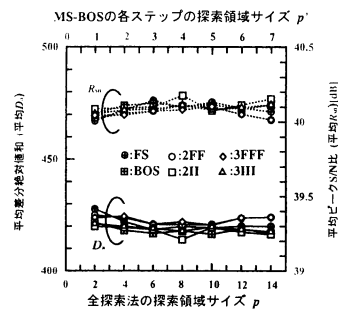


図 3.2 平均 D 、平均 R_m と FS の探索領域サイズ p の関係
("Claire", 64 kbps, 15 fps, $p=14$ 画素, $a_s=2$)

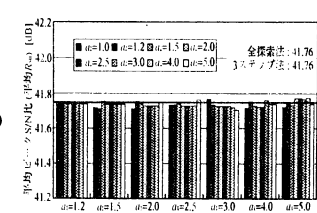
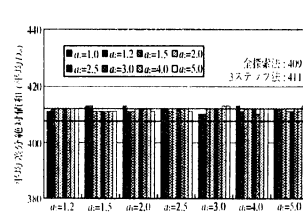
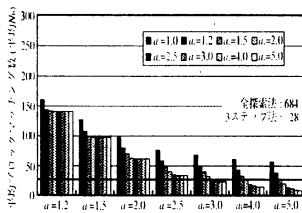


図 3.3 平均 N_b 、平均最小 D 、平均 R_m と a_s の関係 ("Claire", 211, 64 kbps, 15 fps, $p=14$ 画素). (a) 平均 N_b . (b) 平均最小 D . (c) 平均 R_m .



図 3.4 "Claire" の復号画像 (64 kbps, 15 fps, $p=14$). (a) FS. (b) 2II ($p=14$, $p'=7$, $a_s=2$). (c) 2II ($p=14$, $p'=2$, $a_s=5$).

表 3.1 各種動きベクトル抽出アルゴリズムの画像解析結果 ("Claire", $R_f=15$ fps, $R_d=64$ kbps, $p=14$ 画素).

アルゴリズム	平均 N_b (%)	平均 F (倍)	平均 D (%)	平均 R_m (dB)
FS	68.479 (100.0)	1.00	420.023 (0.000)	40.069 (0.000)
BOs	287.434 (41.969)	2.383	416.839 (-0.764)	40.115 (+0.046)
2II ($p'=7$)	80.242 (11.716)	8.535	416.303 (-0.894)	40.154 (+0.085)
3III ($p'=7$)	79.147 (11.556)	8.653	417.526 (-0.598)	40.115 (+0.046)
2FF ($p'=7$)	187.850 (27.428)	3.646	424.003 (+0.948)	40.014 (-0.055)
3FFF ($p'=7$)	187.858 (27.429)	3.646	424.003 (+0.948)	40.014 (-0.055)
2II ($p'=2$)	10.814 (1.579)	63.332	419.415 (-0.145)	40.093 (+0.024)
3III ($p'=2$)	10.780 (1.574)	63.532	421.032 (+0.240)	40.067 (-0.002)
2FF ($p'=2$)	22.684 (3.312)	30.192	423.717 (+0.879)	40.055 (-0.014)
3FFF ($p'=2$)	22.742 (3.321)	30.115	423.334 (+0.788)	40.049 (-0.020)

表 3.2 n ステップ目の BM 終了後の平均 N_b (N_{bn})、平均最小 D 、および平均 R_m と探索領域サイズ ($(p'+16)$ 画素 $\times$ ($p'+16$)ライン) の関係.
("Claire", $R_f=15$ fps, $R_d=64$ kbps, $p=14$ 画素).

アルゴリズム	p'	平均 R_m [dB]	平均 D	平均 N_b	平均 N_{b1}	平均 N_{b2}	平均 F
3FFF	7	40.014	424.003	187.858	185.556	2.295	3.646
	4	40.065	421.581	69.386	68.101	1.243	9.871
	3	40.062	422.448	42.819	41.879	0.903	15.995
	2	40.049	423.334	22.742	22.121	0.568	30.115
3III	7	40.115	417.526	79.147	77.516	1.631	8.653
	4	40.113	417.755	30.311	29.351	0.956	22.595
	3	40.089	418.911	19.043	18.309	0.726	35.966
	2	40.067	421.032	10.780	10.271	0.482	63.532

表 3.3 n ステップ目の BM 時に BM を実行した 1 フレーム当たりの M-Blk 数の平均値 (M_{bn}) (個/フレーム) と動きベクトルを更新した M-Blk 数の平均値 (M_{mn}) (個/フレーム). ("Claire", $R_f=15$ fps, $R_d=64$ kbps, $p=14$ 画素).

アルゴリズム	p'	M_{b1}	M_{b2}	M_{b3}	M_{b4}	M_{b5}
3FFF	7	99.000 (100 %)	7.904 (7.984 %)	7.904 (7.984 %)	0.092 (0.093 %)	0.025 (0.025 %)
	4	99.000 (100 %)	7.736 (7.815 %)	7.736 (7.815 %)	0.201 (0.203 %)	0.021 (0.021 %)
	3	99.000 (100 %)	7.766 (7.844 %)	7.766 (7.844 %)	0.280 (0.283 %)	0.050 (0.051 %)
	2	99.000 (100 %)	7.435 (7.510 %)	7.435 (7.510 %)	0.690 (0.697 %)	0.067 (0.068 %)
3III	7	99.000 (100 %)	6.967 (7.037 %)	6.850 (6.919 %)	0.004 (0.004 %)	0.000 (0.000 %)
	4	99.000 (100 %)	6.971 (7.041 %)	6.837 (6.906 %)	0.025 (0.025 %)	0.000 (0.000 %)
	3	99.000 (100 %)	6.958 (7.029 %)	6.854 (6.923 %)	0.088 (0.088 %)	0.004 (0.004 %)
	2	99.000 (100 %)	6.870 (6.940 %)	6.774 (6.842 %)	0.385 (0.389 %)	0.021 (0.021 %)

平均値、 M_{b1} 、 M_{b2} 、 M_{b3} (個/フレーム)と動きベクトルを更新した M-Blk 数の平均値、 M_{r1} 、 M_{r2} 、 M_{r3} (個/フレーム)、を示す。なお 1 フレーム当たりの総 M-Blk 数は 99 個である。

MS-BOS (FS 形)

SW サイズが最大 ($p' = 7$) の時、表 3.2 より、平均 N_{b1} は 185.556 回、平均 N_{b2} は 2.295 回、平均 N_{b3} は 0.008 回である。ステップ数が増加すると、平均 N_b は急激に減少する。一方、 D_{a2} 、 D_{a3} はともに 424.003 で等しく、 R_{sn2} 、 R_{sn3} もともに 40.014dB で等しい。平均 M_{b1} 、平均 M_{b2} 、平均 M_{b3} はそれぞれ 99 (= 総 M-Blk 数/フレーム) (100%)、7.904 (7.984%)、0.025 (0.025%)である。表 3.3 より、平均 M_{r1} 、平均 M_{r2} 、平均 M_{r3} はそれぞれ 7.904 (7.984%)、0.092 (0.093%)、0 (0%)である。以上の結果より、 p' が 7 の時、ステップ 3 では、BM はほとんど実行されず、動きベクトルを更新する M-Blk も無い。従って、平均最小 D_a は減少されないの、ステップ 3 は割愛してよいことがわかる。

SW サイズが最小 ($p' = 2$) の時、表 3.2 より、平均 N_{b1} は 22.121 回、平均 N_{b2} は 0.568 回、平均 N_{b3} は 0.053 回である。ここでもステップ数が増加すると、平均 N_b は急激に減少するが、 p' が 7 の時 (平均 N_{b3} は 0.008 回) と比べて、平均 N_{b3} は大分大きい。表 3.3 より、平均 M_{b1} 、平均 M_{b2} 、平均 M_{b3} はそれぞれ 99 (100%)、7.435 (7.510%)、0.690 (0.697%)である。平均 M_{r1} 、平均 M_{r2} 、平均 M_{r3} はそれぞれ 7.435 (7.510%)、0.690 (0.697%)、0.067 (0.068%)である。つまり、ステップ 3 で、動きベクトルがわずかに更新される (平均 $M_{r3} = 0.068\%$)。しかし、平均最小 D_{a2} 、平均最小 D_{a3} はそれぞれ 423.717、423.334 で両者はほぼ等しく、かつ平均 R_{sn2} 、平均 R_{sn3} はそれぞれ 40.055B、40.049dB で両者はほぼ等しいことから、“Claire”のように、動きの遅い画像にたいしては、ステップ 3 を割愛してもよいことがわかる。

MSBOS (BOS 形)

3III に対して、3FFF と同様な解析を行なった。SW サイズの大小に無関係に、 D_{a2} と D_{a3} はほぼ等しく、 R_{sn2} 、 R_{sn3} もほぼ等しい値が得られている。さらに、いずれの SW サイズにおいても、3FFF の平均 M_{b3} 、平均 M_{r3} と比べ、3III の平均 M_{b3} 、平均 M_{r3} は小さいことより、動きの遅い画像に対して、3III の場合もステップ 3 は割愛しても差し支えないと言える。

以上の結果より、動きの遅い動画像、“Claire”に対して、2FF、2II ($p' = 2$) は FS ($p = 14$)並の画質 (平均最小 $D_a = 420.023$ 、平均 $R_{sn} = 40.069$ db) を維持し、BM 速度をそれぞれ 30 倍以上、63 倍以上、高速化できることがわかる。

4 動きの速い動画像の解析

4.1 解析画像と解析条件

本節で用いるテスト画像は“Carphone”と呼ばれる動きの速い(平均最小 D_a が大きい)動画像でフレームサイズは QCIF (176 画素×144 ライン)、フレーム数は 382、である。他の解析条件は 3.1 節と同様である。

4.2 検出速度と画質

図 4.1 に平均 N_b 、平均 $F_s(\propto 1/N_b)$ と p の関係を示す。なお、 p' は p/a_n であり、 $a_n=2$ とする。MS-BOS の平均 N_b は、いずれの場合も、FS のそれより小さく、2II、3III が FS の約 1/6.5、2FF、3FFF が FS の約 1/3.3 である。従って、2II、3III の平均 F_s は FS の約 6.5 倍、2FF、3FFF のそれは FS の約 3.3 速くなる。

図 4.2 に平均最小 D_a 、平均ピーク S/N 比(平均 R_{sn})と p の関係を示す。2FF、3FFF の平均最小 D_a は 1,760~1,780 で、FS のそれ(1,753)とほぼ同一である。一方、2II、3III の平均最小 D_a は 1,810~1,830 で、若干大きくなる。FS、2FF、3FFF の平均 R_{sn} は約 30.1dB~30.2dB で、差はほとん

どない。一方、2II、3III の平均 R_{sn} は 30.0dB で、多少劣化が大きくなる。動きの速い動画像に対して、 a_n が 2 の時、2FF、3FFF の平均 N_b は 2II、3III より若干大きい。しかし、画質では 2II、3III、BOS より優れており、FS 並である。これより、動きの速い動画像に対して、2FF、3FFF が優れた動きベクトル検出アルゴリズムと言える。以上の結果を表 4.1 にまとめる。

4.3 SW サイズの最適化

前述した 4 種類の MS-BOS に対して、平均 N_b 、平均最小 D_a 、平均 R_{sn} と a_n の関係を検討した。このうち、2II を用いて得た解析結果を図 4.3 (a)、(b)、(c)に示す。本解析では最適 M-Blk の周囲を半画素精度で 8 回 BM する階層探索が用いられている。但し、半画素精度の BM 数 (8 回) は平均 BM 数 (平均 N_b)に含めていない。

a_n が 5 の時、平均検出速度は最も速く、FS の 30 倍以上となる。 a_n が増加すると、平均最小 D_a は増加 (劣化)し(図 4.3 (b))、 a_n が 5 の時、最大(1,530 以下)となる。しかし、FS の平均 D_a (1,421) の 8%増しに止まっている。平均 R_{sn} は a_n が増加するとわずかに減少する(図 4.3 (c))。 a_n が 5 の時、平均 R_{sn} は最小 (32.305dB)となるが、それでも、FS の平均 R_{sn} (32.443dB)より 0.138dB 下回るだけである。図 4.4(a)、(b)、(c)に FS、2II ($a_n = 2$)、2II ($a_n = 5$)で得られた復号画を示す。画質の優劣は見えない。

4.4 ステップ数 (n) の最適化

表 4.2 はステップ 1 の平均 N_b (N_{b1})、ステップ 2 の平均 N_b (N_{b2})、ステップ 3 の平均 N_b (N_{b3})、平均最小 D_a 、および平均 R_{sn} と探索領域サイズ $\{(2p'+16)$ 画素 $\times$ $(2p'+16)$ ライン $\}$ の関係である。 p' は 2、3、4、5 である。2S-BOS の平均最小 D_a 、平均 R_{sn} をそれぞれ D_{a2} 、 R_{sn2} 、同様に、3S-BOS の平均最小 D_a 、平均 R_{sn} をそれぞれ D_{a3} 、 R_{sn3} と定義する。SW サイズ $\{(2p'+16)$ 画素 $\times$ $(2p'+16)$ ライン $\}$ 、 p' は 2、3、4、5 である。表 4.3 にステップ 1、ステップ 2 およびステップ 3 の BM 時に BM を実行した 1 フレーム当たりの M-Blk 数の平均値 (M_{b1} 、 M_{b2} 、 M_{b3}) (個/フレーム)と動きベクトルを更新した M-Blk 数の平均値 (M_{r1} 、 M_{r2} 、 M_{r3}) (個/フレーム)、を示す。なお 1 フレーム当たりの総 M-Blk 数は 99 個である。

MS-BOS (FS 形)

SW サイズが最大 ($p' = 5$) の時、表 4.2 より、平均 N_{b1} は 100.788 回、平均 N_{b2} は 12.041 回、平均 N_{b3} は 2.114 回である。ステップ数が増加すると、“Claire”と同様、平均 N_b は急激に減少する。また、 D_{a2} 、 D_{a3} はそれぞれ 1,787.905、1,766.257、 R_{sn2} 、 R_{sn3} は、それぞれ 30.159dB、30.244dB であり、3 回目の BM により、画質が向上する。表 4.3 より、平均 M_{b1} 、平均 M_{b2} 、平均 M_{b3} はそれぞれ 99 (= 総 M-Blk 数/フレーム) (100%)、42.978 (43.413%)、5.703 (5.760%)である。平均 M_{r1} 、平均 M_{r2} 、平均 M_{r3} はそれぞれ 42.978 (43.413%)、5.703 (5.760%)、2.222 (2.244%)である。これより、SW が大きいにも関わらず ($p' = 5$)、ステップ 3 で 5.703 個の M-Blk が BM を実行し、2.222 個の M-Blk が動きベクトルを更新していることがわかる。これにより、3 回目の BM は割愛されるべきではなく、これは同一条件の“Claire”と大分異なる特性である。

SW サイズが最小 ($p' = 2$) の時、表 4.2 より、平均 N_{b1} は 22.121 回、平均 N_{b2} は 3.631 回、平均 N_{b3} は 1.023 回である。この平均 N_{b3} は p' が 5 の時の平均 N_{b3} (0.809 回)と大差はない。表 4.3 より、平均 M_{b1} 、平均 M_{b2} 、平均 M_{b3} はそれぞれ 99 (100%)、41.281 (41.698%)、11.595 (11.712%)である。平均 M_{r1} 、平均 M_{r2} 、平均 M_{r3} はそれぞれ 41.281 (41.698%)、11.595 (11.712%)、4.886 (4.936%)である。つまり、3 回目の BM で、11.595 個の M-Blk が BM を実行し、4.886 個の M-Blk が動きベクトルを更新している。また、 D_{a2} 、 D_{a3} は

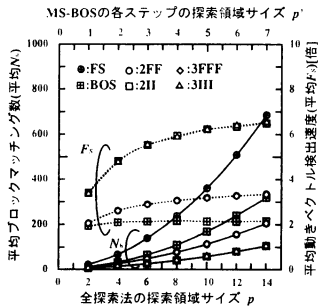


図 4.1 平均 N_s 、平均 F_s と FS の探索領域サイズ p の関係
("Carphone", 64 kbps, 15 fps, $p = 14$ 画素, $a_s = 2$)

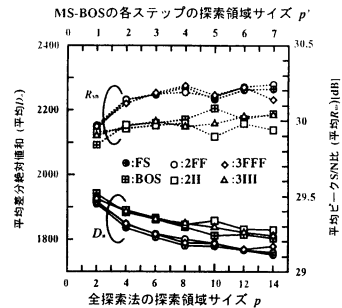


図 4.2 平均 D_s 、平均 R_m と FS の探索領域サイズ p の関係
("Carphone", 64 kbps, 15 fps, $p = 14$ 画素, $a_s = 2$)

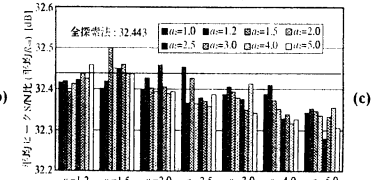
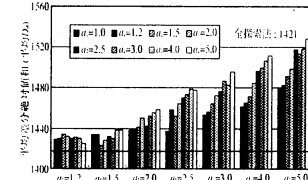
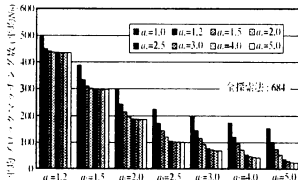


図 4.3 平均 N_s 、平均最小 D_s 、平均 R_m と a_s の関係 ("Carphone", 2II, 64 kbps, 15 fps, $p = 14$ 画素). (a) 平均 N_s . (b) 平均最小 D_s . (c) 平均 R_m .



図 4.4 "Carphone" の復号画像 (64 kbps, 15 fps, $p = 14$). (a) FS. (b) 2II ($p = 14$, $p' = 7$, $a_s = 2$). (c) 2II ($p = 14$, $p' = 2$, $a_s = 5$).

表 4.1 各種動きベクトル検出アルゴリズムの画像解析結果 ("Carphone", $R_f = 15$ fps, $R_d = 64$ kbps, $p = 14$ 画素).

アルゴリズム	平均 N_s (%)	平均 F_s (個)	平均 D_s (%)	平均 R_m [dB] (dB)
FS	684.879 (100.0)	1.00	1752.554 (0.000)	30.210 (0.000)
BOS	318.952 (46.570)	2.147	1803.439 (+2.903)	30.046 (-0.164)
2II ($p' = 5$)	57.84 (8.445)	11.841	1857.715 (+6.000)	29.897 (-0.313)
3III ($p' = 5$)	59.986 (8.759)	11.417	1840.159 (+4.999)	29.916 (-0.294)
2FF ($p' = 5$)	112.914 (16.487)	6.065	1787.905 (+2.017)	30.159 (-0.051)
3FFF ($p' = 5$)	114.943 (16.783)	5.958	1766.257 (+0.782)	30.244 (+0.034)
2II ($p' = 2$)	13.927 (2.033)	49.177	1889.731 (+7.827)	29.948 (-0.262)
3III ($p' = 2$)	14.624 (2.135)	46.834	1867.877 (+6.580)	29.994 (-0.216)
2FF ($p' = 2$)	25.794 (3.766)	26.552	1846.674 (+5.370)	30.139 (-0.071)
3FFF ($p' = 2$)	26.775 (3.909)	25.579	1823.389 (+4.042)	30.163 (-0.047)

表 4.2 n ステップ目の BM 終了後の平均 N_s (N_{bn})、平均最小 D_s 、および平均 R_m と探索領域サイズ ($(p'+16)$ 画素 $\times$ ($p'+16$) ライン) の関係.
("Carphone", $R_f = 15$ fps, $R_d = 64$ kbps, $p = 14$ 画素).

アルゴリズム	p'	平均 R_m [dB]	平均 D_s	平均 N_s	平均 N_{s1}	平均 N_{s2}	平均 N_{s3}	平均 F_s
3FFF	5	30.244	1766.257	114.943	100.788	12.041	2.114	5.958
	4	30.195	1782.837	78.998	68.101	9.064	1.834	8.670
	3	30.196	1803.886	49.489	41.879	6.208	1.403	13.839
	2	30.163	1823.389	26.775	22.121	3.631	1.023	25.579
3III	5	29.916	1840.159	59.986	51.865	6.884	1.237	11.417
	4	29.974	1840.835	41.222	35.053	5.067	1.101	16.614
	3	30.011	1847.278	26.034	21.786	3.471	0.778	26.307
	2	29.994	1867.877	14.624	11.942	2.098	0.584	46.834

表 4.3 n ステップ目の BM 時に BM を実行した 1 フレーム当たりの M-Blk 数の平均値 (M_{bn} (個/フレーム)) と動きベクトルを更新した M-Blk 数の平均値 (M_{rn} (個/フレーム)). ("Carphone", $R_f = 15$ fps, $R_d = 64$ kbps, $p = 14$ 画素).

アルゴリズム	p'	M_{b1}	M_{r1}	M_{b2}	M_{r2}	M_{b3}	M_{r3}
3FFF	5	99.000 (100.0 %)	42.978 (43.413 %)	42.978 (43.413 %)	5.703 (5.760 %)	5.703 (5.760 %)	2.222 (2.244 %)
	4	99.000 (100.0 %)	43.103 (43.538 %)	43.103 (43.538 %)	6.595 (6.661 %)	6.595 (6.661 %)	2.670 (2.697 %)
	3	99.000 (100.0 %)	42.746 (43.178 %)	42.746 (43.178 %)	8.530 (8.616 %)	8.530 (8.616 %)	3.611 (3.647 %)
	2	99.000 (100.0 %)	41.281 (41.698 %)	41.281 (41.698 %)	11.595 (11.712 %)	11.595 (11.712 %)	4.886 (4.936 %)
3III	5	99.000 (100.0 %)	33.578 (33.918 %)	25.422 (25.678 %)	3.584 (3.620 %)	3.351 (3.385 %)	1.411 (1.425 %)
	4	99.000 (100.0 %)	33.292 (33.628 %)	24.897 (25.149 %)	4.081 (4.122 %)	3.886 (3.926 %)	1.719 (1.736 %)
	3	99.000 (100.0 %)	32.784 (33.115 %)	24.459 (24.707 %)	5.049 (5.100 %)	4.708 (4.756 %)	2.114 (2.135 %)
	2	99.000 (100.0 %)	33.054 (33.388 %)	24.714 (24.963 %)	7.330 (7.404 %)	6.503 (6.568 %)	3.103 (3.134 %)

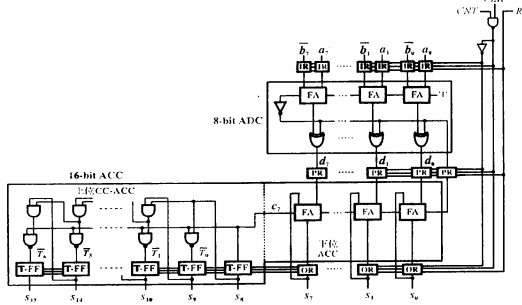


図 5.1 逐次桁上げ伝播形加算回路とカウンタを用いた 16 b 累算形差分絶対値和回路 (累算形 ADA) の構成。

それぞれ 1,846.674、1,823.389、 R_{sn2} 、 R_{sn3} はそれぞれ 30.139dB、30.163dB であり、ステップ 3 まで探索することにより、画質が向上することから、"Carphone" のように、動きの速い画像にたいして、3 回目の BM は割愛されるべきではないことがわかる。

MS-BOS(BOS 形)

3III に対しても、3FFF と同様な解析を行なった。表 4.3 より、平均 M_{b1} 、平均 M_{b2} 、平均 M_{b3} はそれぞれ 99 (100%)、24.714 (24.963%)、6.503 (6.568%) である。平均 M_{r1} 、平均 M_{r2} 、平均 M_{r3} はそれぞれ 33.054 (33.388%)、7.330 (7.404%)、3.103 (3.134%) である。つまり、ステップ 3 で、6.503 個の M-Blk が BM を実行し、3.103 個の M-Blk が動きベクトルを更新している。3FFF に比べ、これらの値は削減されているが、"Carphone" のように、動きの速い画像にたいしてステップ 3 は割愛されるべきではないことがわかる。

以上の結果より、動きの速い動画像、"Carphone"、に対して、FS ($p=14$) 並の画質 (平均最小 $D_a=1,752.554$ 、平均 $R_{sn}=30.210$ dB) を維持できるのは、 p' が 5 の 3FFF に限定される。この場合、BM 速度は FS の約 6.0 倍程度になることがわかる。

5 差分絶対値和回路の設計

5.1 回路構成

2 段パイプライン構成、16 累算形差分絶対値和回路 (ADA) を図 5.1 に示す。本回路は、1 画素分のデータの差分絶対値を得る 8 b 差分絶対値回路と 1M-Blk 分の差分絶対値を累算する 16 b 累算回路で構成される。8 b 差分絶対値回路は逐次桁上げ伝播形の 8 b 加算回路で、16 b 累算回路は逐次桁上げ伝播形の 8 b 加算回路 (下位 8 b) と 8 b カウンタ (上位 8 b) で構成されている [3]。これを 0.18- μ m、CMOS (ダブルウェル、1 層ポリシリコン、2 層ポリ Si、3 層 Al メタル配線) でレイアウト設計した。nMOS、pMOS のマスク上のゲート長は共に 0.2 μ m、nMOS、pMOS の閾値電圧、 V_t はそれぞれ 0.435 V、-0.415 V、標準電源電圧、 V_D は 1.8 V である。

レイアウトより RC 抽出した結果を用いて SPICE 解析した結果、本回路の消費電力、 P は 783 μ W ($f_c=260$ MHz、 $V_D=1.8$ V)、170 μ W ($f_c=130$ MHz、 $V_D=1.2$ V) であった。

5.2 稼働率の低減による回路の低電力化

260 MHz 動作の累算形差分絶対値和回路 (ADA) は 1 回路で最大 686 回の BM 処理が可能である。フレームレート、 R_f が 15 fps の QCIF 画像の場合、 p が 14 画素の FS、BOS、および p が 7 画素の (3FFF を含めた) MS-BOS の処理が可能である。各アルゴリズムで得られる平均 N_b を 686 で規格化すれば、各アルゴリズムを処理する累算形 ADA の平均

稼働率、 α として定義できる。従って、BM が終了した時点で、クロックパルスの供給を停止して、回路を停止すれば、累算形 ADA の P はさらに削減される。つまり、各アルゴリズムを採用した累算形 ADA の平均 P' は 260 MHz 動作時の P (783 μ W) の α 倍 (αP) となる。具体的には、BM が終了した時点で、図 5.1 に示す制御パルス (CNT) を "1" として、クロックパルスを停止する ("ゲートクロック" と呼ばれている)。これより、FS ($\alpha=0.9975$ 、 $p=14$) の P' は 781.8 μ W である。一方、3FFF ($\alpha=0.1675$ 、 $p'=5$ 、"Carphone") の P' は 131.1 μ W となり、FS の 16.8% に削減される。2II ($\alpha=0.01579$ 、 $p'=2$ 、"Claire") の場合、 P' はわずか 12.3 μ W で、FS の 1.6% となる。

3S-BOS ($p'=7$) の場合、最大 BM 回数が 558 回である。この場合、 f_c を 210 MHz まで低減できるので、上記よりさらに P' を削減できる。

5.3 回路の並列化によるさらなる低電力化

累算形 ADA を並列化すると、並列数に反比例して f_c を低減できる。2 並列の場合、 f_c は 130 MHz でよい。回路規模は 2 倍になるが 3FFF ($\alpha=0.1675$ 、 $p'=5$ 、"Carphone")、2II ($\alpha=0.01579$ 、 $p'=2$ 、"Claire") を適用すると、 P' はそれぞれ 56.95 μ W、5.37 μ W となる。

6 おわりに

携帯 TV 電話、携帯機器、等へ搭載する画像処理向 LSI の消費電力の低減を図るため、FS の画質を維持したまま、動きベクトル検出処理量を大幅に削減できるマルチステップ中断法 (MS-BOS) を開発した。 n 回目のステップ目の探索領域 (SW) サイズ、 p' を縮小して最適化を図ると、動きの遅い動画像 ("Claire") に対して、 $n=2$ 、 $p'=2$ に設定すると、FS と遜色のない画質を維持し、検出速度を 3 ステップ法の 2.5 倍、FS の 63 倍、にそれぞれ向上させることができた。一方、動きの速い動画像 ("Carphone") に対して、 $n=3$ 、 $p'=5$ に設定すると、FS と遜色のない画質を維持し、検出速度は 6 倍以上速いことがわかった。

新たに開発した 0.18- μ m CMOS カウンタ形差分絶対値和回路と MS-BOS ($\alpha=0.01579$ 、 $p'=2$ 、 $n=2$ 、 $a_n=5$ 、"Claire") を導入した結果、消費電力はわずか 12.3 μ W まで低減できた。これは FS を用いた場合の 1.6% である。

以上の結果より、携帯 TV 電話等、動きの遅い動画像を対象とする場合、 n 、 p を小さく設定できる MS-BOS が有効である、と言える。

謝辞 ご協力頂いた中央大学理工学部複本研究室の皆さまに心より感謝致します。

参考文献

- [1] T. Enomoto, Y. Sasajima, A. Hirobe, and T. Ohsawa, "Fast Motion Estimation Algorithm and Low Power CMOS Motion Estimation Array LSI for MPEG-2 Encoding", in Proc. of International Symposium on Circuits and Systems (ISCAS'99), vol-IV, pp. 203-206, in Orlando, Florida, May/June 1999.
- [2] T. Enomoto and A. Kotabe, "Fast Motion Estimation Algorithm and Low Power 0.13 μ m CMOS Motion Estimation Circuits", in Proc. of International Symposium on Circuits and Systems (ISCAS'2001), Sidney, Australia, vol-II, pp. 449-452, May 2001.
- [3] 小林伸彰、渡邊友樹、榎本忠儀、「適応的に処理回数を最小化する高速動きベクトル検出 (ME) アルゴリズムとこれを適用した低消費電力 CMOS ME 回路の開発」、信学技報、ICD2004-69, pp. 43-48、2004 年 8 月 19 日。
- [4] T. Koga, et al., Nat. Telecomms. Conf., pp. G5.3.1-G5.3.5, Nov. 1981.