

額縁形探索窓を採用した SHV 向け 動きベクトル検出アルゴリズム

Picture Frame Shaped Search Window Algorithm for Block Motion Estimation in SHV

情報工学専攻 13N810009J 関 健太郎

あらまし

SHV 向けに動きベクトル検出処理速度を超高速度化した額縁形探索窓動きベクトル検出アルゴリズムを開発した。本アルゴリズムは1次～3次探索で構成される。1次探索は複数の候補点より最適な2次探索の探索開始点を得る予備探索、2次探索は額縁形探索窓探索、3次探索はダイヤモンド探索である。本アルゴリズムを H.264/AVC に準拠したソフトウェアエンコーダに実装し、6種類の HDTV(High Definition Television)画像(“Bronze with Credits”, “Ice Hockey”, “Whale Show”, “Tractor”, “Riverbed”, “Inter-section”)に適用し、画質、探索速度を評価した。その結果、本アルゴリズムは全探索法と遜色のない高画質を維持した上で、既存の高速アルゴリズムより探索速度を高速化できた。

キーワード：H.264, 動きベクトル検出, サブサンプリング

1 はじめに

8K UHD (Ultra High Definition Television)と呼ばれるスーパーハイビジョンの放送が2016年に開始される予定である。フル解像度のUHDでは、1フレーム当たりの総画素数は9,950万(7,680画素×4,320ライン×RGB)と極めて多い。この膨大なデータを実時間で符号化できる動画像符号化プロセッサを実現するためには、処理量が特に大きい動きベクトル検出(Block Motion Estimation; BME)を高速処理できるアルゴリズムの開発が必須である。これまでに、高速 BME アルゴリズムとして、Simplified Unsymmetrical-cross Multi-Hexagon-grid Search (S-UMHS)^[1], Enhanced Predictive Zonal Search (EPZS)^[2]等が開発された。

更なる高速化を目指して、我々は額縁形探索窓(Picture Frame shaped Search Window; PFSW)を採用したサブサンプリングアルゴリズムを開発した^[5]。PFSW アルゴリズムを H.264/AVC に準拠したソフトウェアエンコーダ JM12.3^[3] に実装し、5種類の HDTV 画像(“Bronze with Credits”, “Ice Hockey”, “Whale Show”, “Tractor”, “Riverbed”)に適用し、評価した。その結果、PFSW アルゴリズムは再生画像の画質を全探索法(Full Search; FS)のそれと同等の画質を維持できた。また、PFSW アルゴリズムの探索速度は、“Bronze with Credits”の場合、FS, S-UMHS, EPZS アルゴリズムのそれぞれ325.55倍、3.795倍、1.656倍、高速化された。

以下、第2章でPFSWアルゴリズムを、第3章でPFSWの画像解析結果を、第4章ではPFSWアルゴリズムの速度・画質を既報アルゴリズムと比較する。

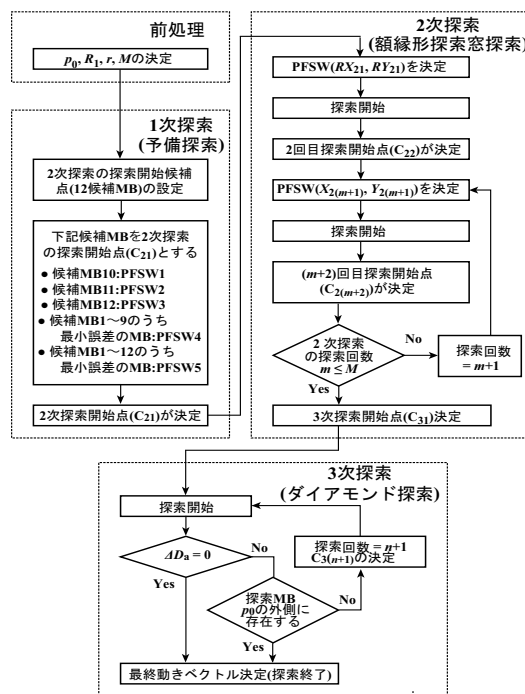


図 1.1 額縁形探索窓(PFSW)アルゴリズム

2 額縁形探索窓(PFSW)アルゴリズム

額縁形探索窓(PFSW)アルゴリズムは図 2.1 に示すように、1次、2次、3次探索で構成される。

1次探索は(i)符号化対象マクロブロック(MB)の周辺MBの動きベクトル情報より12個の探索開始候補点(候補MB)を設定し、次に(ii)これらの中から1個の候補MBを選択し、これを2次探索の探索開始点とする。2次探索は額縁形探索窓(PFSW)を採用した繰り返しサブサンプリング探索である。探索窓サイズを縮小しながら、サイズが最小となるまで、これを複数回繰り返すことにより、より最適なMBを検出する。3次探索は2次探索で得られた最適MBを探索開始点とするダイヤモンド探索で、最終MBを検出する。

2.1 1次探索

2.1.1 探索開始候補点(候補MB)の設定

2次探索の探索開始点を決定するため、1次探索で12個の探索開始候補点(候補MB)を設定する。図2.2に現フレーム、前(参照)フレーム、前々フレームおよびMBの位置関係を示す。Aは現フレーム中の符号化対象MB、B～Eは現フレーム中のMB、PはAと同位置で前フレーム中のMB、F～Iは前フレーム中のMB、QはAと同位置で前々フレーム中のMBである。今、MB名がZの動きベクトルを

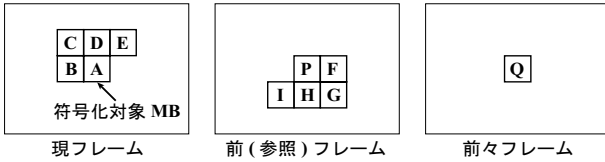


図 2.1 符号化対象 MB と周辺 MB

MV_Z , MV_Z の x 方向成分を MV_{Zx} , y 方向成分を MV_{Zy} と定義し, この定義を用いて, 上記 12 個の候補点を下記のように分類する.

- 1 個目, 2 個目, 3 個目, 4 個目, 5 個目, 6 個目, 7 個目, 8 個目の候補 MB は前フレーム中で, P から $MV_B, MV_C, MV_D, MV_E, MV_F, MV_G, MV_H, MV_I$ だけ離れた位置の MB, 9 個目の候補 MB は P である.
- 10 個目の候補 MB は前フレーム中で, P から x 方向に $MV_{Bx}, MV_{Dx}, MV_{Ex}$ の中間値だけ, y 方向に $MV_{By}, MV_{Dy}, MV_{Ey}$ の中間値だけ離れた位置の MB である^[6]. 例えば, $MV_{Bx} \geq MV_{Dx} \geq MV_{Ex}$, $MV_{Ey} \geq MV_{By} \geq MV_{Dy}$, とすれば, 候補 MB は P から x 方向に MV_{Dx} だけ, y 方向に MV_{By} だけ離れた位置の MB である.
- 11 個目の候補 MB は前フレーム中で, P から x 方向に $MV_{Bx} \sim MV_{Ix}, MV_{Px}$ の平均値だけ, y 方向に $MV_{By} \sim MV_{Iy}, MV_{Py}$ の平均値だけ離れた位置の MB である.
- 12 個目の候補 MB は前フレーム中で, P から x 方向に $\{(MV_{Px}-MV_{Qx})+MV_{Px}\}$, y 方向に $\{(MV_{Py}-MV_{Qy})+MV_{Py}\}$ だけ離れた位置の MB である. 2 次探索の探索開始点は 1~12 個目の各候補点の全てでブロックマッチングを行い, 差分絶対値和 (D_a) を求め, この中から D_a が最小の候補点を 2 次探索の探索開始点とする.

2.2 2 次探索(PFSW 探索)

2 次探索は額縁形探索窓(PFSW)を採用した繰り返しサブサンプリング探索である. 第 1 回探索は 1 次探索 1~5 の何れかで決定された探索開始点から動きベクトル検出(BME)を開始し, 大きく設定された額縁形探索窓で BME を実施する. 第 2 回探索以後, 窓サイズを順次縮小しながら, BME を繰り返すことにより, より最適な MB を検出する.

2.2.1 探索窓サイズの設定

A. 第 1 回探索

前フレーム中に設定される第 1 回探索の探索窓, 探索中心を図 2.2 に示す. 全探索法(FS)の探索窓サイズ(x 方向 F_x , y 方向 F_y)= (X_F, Y_F) ($\pm p_0$ 画素, $\pm p_0$ 画素), A, P, Q の位置(C_1)を $(x_1=0, y_1=0)$ とする. また, 第 1 回探索の探索開始点 (C_{21})を前フレーム中の (x_{21}, y_{21}) とする. $C_{21}(x_{21}, y_{21})$ は $C_1(x_1=0, y_1=0)$ から x 方向に x_{21} 画素, y 方向に y_{21} 画素離れた位置にある.

第 1 回探索の額縁形探索窓サイズ(x 方向 2_1 , y 方向 2_1)= (X_{21}, Y_{21}) を, FS の探索窓サイズ(X_F, Y_F)より小さく設定する. つまり, $\{\pm(p_0-x_{21}), \pm(p_0-y_{21})\}$ とする. 図 2.2 では, 一例として, $(X_F, Y_F)=(\pm 10, \pm 10)$, $(x_{21},$

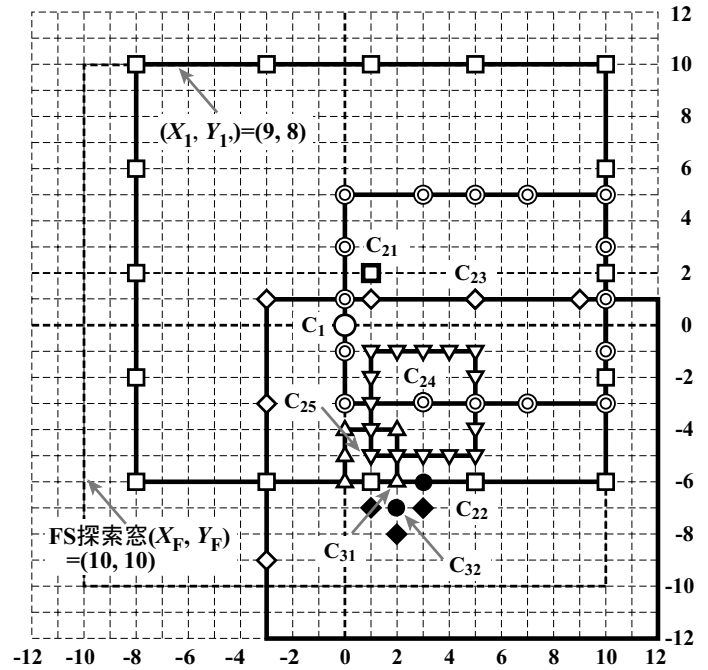


図 2.2 額縁形探索窓(PFSW)アルゴリズムの処理過程

$y_{21})=(1, 2)$ としているので, $(X_{21}, Y_{21})=(9, 8)$ となる. (X_{21}, Y_{21}) をさらに縮小する事も考慮し, 縮小窓サイズを $(RX_{21}, RY_{21})=\{\pm R(p_0-x_{21}), \pm R(p_0-y_{21})\}$ と定義する. R は縮小計数 ($R \leq 1$) で, 下記に定義する 6 個の「距離」と距離計数 ($d \leq 1$) から決定される. 6 個の距離が小さい場合(距離 $\leq p_0 d$), R を小さな値(例えば, $R=0.5$)とし, 探索窓サイズ小さくする. 一方, 6 個の距離が大きき場合(距離 $> p_0 r$), R を大きな値(例えば, $R=1.0$)とし, 探索窓サイズを大きくする. 6 個の距離は B, D, F の各動きベクトルと P の動きベクトルの同一方向成分の差で定義される. つまり, 6 個の距離は $MV_{Bx}-MV_{Px}, MV_{Dx}-MV_{Px}, MV_{Fx}-MV_{Px}, MV_{By}-MV_{Py}, MV_{Dy}-MV_{Py}, MV_{Fy}-MV_{Py}$ である.

B. 第 2 回探索以後

2 次探索, 第 m 回探索の探索窓サイズ(x 方向 2_m , y 方向 2_m)= (X_{2m}, Y_{2m}) は下式で設定される ($m \geq 2$).

$$X_{2m} = \{(M-m)/(M-1)\} X_{2(m-1)} + 1 \quad (1)$$

$$Y_{2m} = \{(M-m)/(M-1)\} Y_{2(m-1)} + 1 \quad (2)$$

ここで, M は第 2 次探索の最大繰り返し探索回数である. 第 2 回探索 ($m=2$) の (X_{22}, Y_{22}) は

$$X_{22} = \{(M-2)/(M-1)\} X_{21} + 1 \approx X_{21} + 1$$

$$Y_{22} = \{(M-2)/(M-1)\} Y_{21} + 1 \approx Y_{21} + 1$$

となる. 第 3 回探索 ($m=3$) 以後の探索窓サイズも同様に設定できる. 第 $(M-1)$ 回探索の $(X_{2(M-1)}, Y_{2(M-1)})$ は

$$X_{2(M-1)} = \{(M-M+1)/(M-1)\} X_{2(M-2)} + 1$$

$$= \{X_{2(M-2)}/(M-1)\} + 1$$

$$Y_{2(M-1)} = \{(M-M+1)/(M-1)\} Y_{2(M-2)} + 1$$

$$= \{Y_{2(M-2)}/(M-1)\} + 1$$

となる. 最終回 ($m=M$) の (X_{2M}, Y_{2M}) は

$$X_{2M} = \{(M-M)/(M-1)\} X_{2(M-1)} + 1 = 1$$

$$Y_{2M} = \{(M-M)/(M-1)\} Y_{2(M-1)} + 1 = 1$$

となり, 2 次探索は終了する.

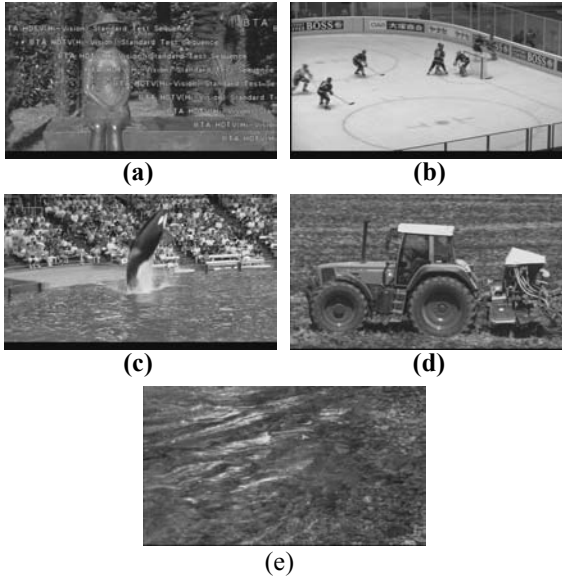


図 3.1 評価に用いた HDTV 動画像 (a) “Bronze with Credits”, (b) “Ice Hockey”, (c) “Whale Show”, (d) “Tractor”, (e) “Riverbed”

2.2.2 2次探索の探索手順

A. 探索点の設定

探索点は額縁状探索窓上に 16 点設定される。第 1 回探索の探索点は、図 2.3 に示すように、上部額縁には、 C_{21} の x 座標点(x_{21})から $-RX_{21}$, $-RX_{21}/2$, 0 , $+RX_{21}/2$ だけ離れた位置の 4 点(□)に設定される。以下同様に、残りの下部、左、右探額縁に 4 点ずつ設定される。第 2 回探索以後の探索点も、第 1 回探索の探索点と同様に、額縁状探索窓上に通常 16 点設定される。

B. 最適探索点の決定

最適探索点は額縁上の 16 点と探索開始点の予測誤差(D_a)のうち最小 D_a の MB とする。第 1 回探索では、まず額縁上の 16 点の探索点と符号化対象 MB(A)との間でブロックマッチングを行い、 D_a を求める。これらの D_a と計算済みの探索開始点(C_{21})の D_a のうち最小 D_a の MB を第 2 回探索の探索開始点(C_{22})= (x_{22}, y_{22}) とする。第 m 回($m=2 \sim M$)の最適探索点も第 1 回探索と同様なブロックマッチングにより検出される。

2.3 3次探索(ダイヤモンド探索)

図 2.3 に示すように、3 次探索は $C_{31}=(2, -6)$ (Δ) を中心として、上下左右の 4 点{本図では 2 点(\bullet)}で第 1 回探索を行い、最小 D_a の MB を求める。この MB が $C_{32}=(2, -7)$ とすれば、次に、 C_{32} の周囲の 3 点(\blacklozenge)で探索を行う。以後同様に、3 点(あるいは 2 点)探索を継続する。探索の停止は探索ステップ毎に全探索点の差分絶対値和(D_a)の減少が停止する{減少分(ΔD_a)が 0 となる}時点とする。また、 D_a の

表 3.1 画質と処理量の解析条件

プロファイル@レベル	MP@HL
GOP 構造	$N=15, M=3$ (IBBPBBPBBPBBPBBIBB.....)
データレート(R_d)	15 Mbps
フレームサイズ	1,920×1,088 (1080+8 black lines)
フレームレート(R_f)	25 fps
探索窓サイズ(p_0)	$\pm 16 \sim \pm 192$ 画素 (HDTV)
フレーム数	450 (1,080i), 150 (1,088p)
1/4 画素探索	使用
参照フレーム数	1
ブロックサイズ	16×16, 16×8, 8×16, 8×8, 8×4, 4×8, 4×4 画素
レートコントロール	使用
Early Skip	非使用
InitialQP	画像ごとに適応的に設定
縮小計数($R \leq 1$)	6 個の距離 $\leq 0.25p_0: R=0.5$ 6 個の距離 $> 0.25p_0: R=1.0$
距離計数($d \leq 1$)	0.25
FS の窓サイズ($\pm p_0, \pm p_0$)	10

減少が停止する以前に、探索点が FS の探索窓 ($\pm X_F, \pm Y_F$) の外側に出る場合も 2 次探索を停止する。この場合、外側に出る前に得られた最小 D_a の探索点を符号化対象 MB の最終動きベクトルとする。

3 額縁形探索窓(PFSW)アルゴリズムの特性

3.1 テスト画像と解析条件

H.264/AVC に準拠したソフトウェアエンコーダ JM12.3^[7] に PFSW アルゴリズムを実装し、画質と処理量を評価した。用いた画像は 6 種類の HDTV 動画像で、動きの複雑な “Bronze with Credits”, “Ice Hockey”, 動きの速い “Whale Show”, “Tractor”, 動きの遅い “Riverbed”, である(図 3.1)。解析条件を表 3.1 にまとめる。

3.2 PFSW の探索速度と画質の評価

図 3.2 に “Bronze with Credits” で評価した各種アルゴリズムの解析結果を示す。(a) は平均差分演算回数/MB(N_b), (b) は平均ピーク S/N(R_{sn}), (c) はフレーム毎に推移する R_{sn} , (d) は R_{sn} とデータレート(R_d) の関係である。(a), (b) 中の \circ は FS, \square は従来の高速アルゴリズムである S-UMHS^[1], ∇ は同じく従来の高速アルゴリズムである EPZS^[2], \bullet は PFSW である。表 3.2 に “Bronze with Credits”, で評価した上記アルゴリズムの N_b , R_{sn} , 速度比(F_s), FS を基準としたピーク S/N 差をまとめる。最も高い R_{sn} が得られる最小の窓サイズは $p_0 = \pm 64$ 画素である。

3.3 PFSW の探索速度と画質の評価

3.3.1 速度解析

p_0 が増加すると、FS, S-UMHS の N_b は増加するが、EPZS, PFSW はほぼ一定である。全ての探索範囲において PFSW の N_b が、最も少ない。つまりそれぞれの範囲で、PFSW が最も高速なアルゴリズムと言える。 $p_0 = \pm 64$ 画素では、PFSW は FS の 325.55 倍、S-UMHS の 3.795 倍、EPZS の 1.656 倍高速化されている。

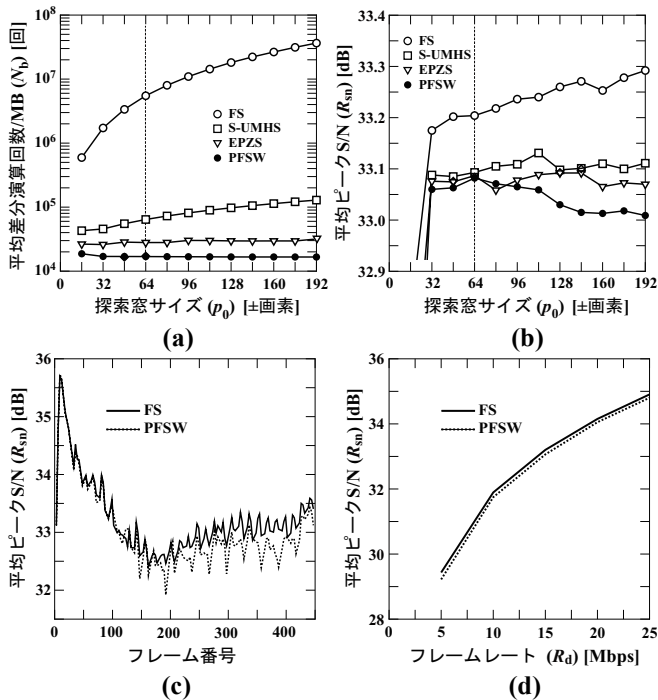


図 3.2 “Bronze with Credits”で評価した各種アルゴリズムの解析結果 (a) 平均差分演算回数/MB(N_b)と p_0 の関係, (b) 平均ピーク S/N(R_{sn})と p_0 の関係, (c) フレーム毎に推移する R_{sn} , (d) R_{sn} とデータレート(R_d)の関係

表 3.2 に “Ice Hockey”, “Whale Show”, “Tractor”, “Riverbed”, で評価した各種アルゴリズムの解析結果も合わせて示す. 各画像とも, p_0 の値は最も高い R_{sn} が得られる最小窓サイズの値である.

いずれのテスト画像においても, PFSW が最も速いアルゴリズムであることがわかる. 例えば, “Whale Show”, $p_0=\pm 112$ 画素では, PFSW は FS の 697.47 倍, S-UMHS の 5.472 倍, EPZS の 2.148 倍, 高速化されている.

3.3.2 画質解析

$\pm 32 \leq p_0 \leq \pm 192$ 画素で PFSW の R_{sn} は高速アルゴリズムである, S-UMHS, EPZS, の R_{sn} はほぼ同等であるが, FS と比べ, 若干(0.1~0.2dB)劣化している. しかし, $p_0=\pm 64$ 画素では, PFSW の R_{sn} (33.082dB) は FS の R_{sn} (33.204dB) より 0.122dB 下回るだけで, 画質に遜色は無いと言える.

図 3.2(c) に FS と PFSW の R_{sn} を示す. 両 R_{sn} はいずれのフレームにおいてもほぼ一致している. 図 3.2(d) にデータレート(R_d)を変数とする FS と PFSW の R_{sn} を示す. 両 R_{sn} にほとんど差はみられない. 図 3.2(b)~(d) より, PFSW は FS 並みの画質を補償していることがわかる. PFSW の R_{sn} を FS の R_{sn} と比べると, “Bronze with Credits”では 0.122 dB の減少(劣化), “Ice Hockey”では 0.054 dB の減少(劣化), “Whale Show”では 0.065 dB の増加(改善), “Tractor”では 0.002 dB の減少(劣化), “Riverbed”では 0.004 dB

表 3.2 動画解析結果 ($R_d=15\text{Mbps}$, $R_F=25\text{fps}$, P ピクチャ)

テスト画像 (p_0)	アルゴリズム	N_b [回]	F_s [倍]	F_s [倍]	F_s [倍]	R_{sn} [dB]	FS との差 [dB]
Bronze with Credits (± 64)	FS	5,498,639.94	1.00			33.204	0.000
	S-UMHS	64,093.28	85.79	1.000		33.093	-0.111
	EPZS	27,972.85	196.57	2.291	1.000	33.087	-0.117
	PFSW5	16,890.21	325.55	3.795	1.656	33.082	-0.122
Ice Hockey (± 112)	FS	12,760,735.59	1.00			40.347	0.000
	S-UMHS	40,214.90	317.31	1.000		40.248	-0.099
	EPZS	25,940.43	491.92	1.550	1.000	40.320	-0.027
	PFSW5	20,794.96	613.65	1.934	1.247	40.293	-0.054
Whale Show (± 112)	FS	17,431,853.10	1.00			31.357	0.000
	S-UMHS	136,764.11	127.46	1.000		31.386	0.029
	EPZS	53,682.20	324.72	2.548	1.000	31.408	0.051
	PFSW5	24,993.04	697.47	5.472	2.148	31.422	0.065
Tractor (± 64)	FS	4,626,846.60	1.00			39.669	0.000
	S-UMHS	54,737.63	84.53	1.000		39.578	-0.091
	EPZS	33,129.12	139.66	1.652	1.000	39.559	-0.110
	PFSW5	22,431.65	206.26	2.440	1.477	39.667	-0.002
Riverbed (± 16)	FS	850,072.38	1.00			36.137	0.000
	S-UMHS	61,133.16	13.91	1.000		36.148	0.011
	EPZS	26,460.73	32.13	2.310	1.000	36.166	0.029
	PFSW5	18,692.28	45.48	3.227	1.416	36.133	-0.004

の減少(劣化), である. PFSW と FS の差が極めて小さいことから, PFSW は FS と全く遜色のないアルゴリズムで, 画質面で最も優れたアルゴリズムの一つと言える.

4 おわりに

額縁形探索窓 (Picture Frame-Shaped Window; PFSW) を採用した PFSW アルゴリズムを開発し, 5 種類の HDTV テスト画像を用いて, 探索速度と画質を評価した. この結果, 本アルゴリズムは多種多様な画像(動きの遅い画像, 動きの速い画像, 絵柄が複雑な画像)に柔軟に対応でき, かつ FS と同等の画質を維持できるアルゴリズムであることがわかった. さらに, PFSW アルゴリズムは, 既報告アルゴリズム中, 最も高速処理できる動きベクトル検出アルゴリズムであることもわかった.

謝辞

本論文の研究にあたり, 熱心な御指導と御助言を下された榎本名誉教授, 小林准教授に心より感謝致します.

また, 本論文をまとめるにあたり数々のご指導下さった指導教授の趙教授ならびに情報工学専攻教授の方々に心より御礼を申し上げます.

参考文献

- [1] X. Yi, J. Zhang, N. Ling, and W. Shang, “Improved and simplified fast motion estimation for JM”, JVT-P021.doc, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, 16th Meeting, Pozan, Poland, July 2005.
- [2] A. M. Tourapis, Proc. in Visual Communications and Image Processing, in San Jose, CA, USA, pp. 1069-1079, Jan. 2002.
- [3] JM Reference Software [Available Online].
<http://bs.hhi.de/suehring/tml/download/>