

最急降下法を用いた動きベクトル検出の高速化と インターレース画像への適用

今村 幸祐[†] 中西 靖人^{††} 橋本 秀雄[†]

[†] 金沢大学工学部

〒 920-8667 石川県金沢市小立野 2-40-20

^{††} 金沢大学大学院自然科学研究科

〒 920-8667 石川県金沢市小立野 2-40-20

E-mail: †{imamura,hasimoto}@is.t.kanazawa-u.ac.jp, ††vel810@gin.ec.t.kanazawa-u.ac.jp

あらまし 本論文では、勾配法の一つである最急降下法を用いた高速な動きベクトル検出アルゴリズムを提案する。提案法は、勾配法において一般的に発生する局所的な解に陥る問題を、適応的な初期値設定法と階層化手法の適用により改善し、予測特性の改善と動きベクトル検出処理の高速化を行うものである。シミュレーションにより、予測特性および演算量を他の手法と比較することで提案法の有効性を示す。また、インターレース画像に対する提案法の有効性を、フレーム予測およびフィールド予測に適用することで検証する。

A Fast Motion Vector Detection Using Steepest Descent Method and Applying it to Interlaced Images

Kousuke IMAMURA[†], Yasuto NAKANISHI^{††}, and Hideo HASHIMOTO[†]

[†] Faculty of Engineering, Kanazawa University

2-40-10 Kodatsuno, Kanazawa, Ishikawa, 920-8667 Japan

^{††} Graduate School of Natural Science & Technology, Kanazawa University

2-40-10 Kodatsuno, Kanazawa, Ishikawa, 920-8667 Japan

E-mail: †{imamura,hasimoto}@is.t.kanazawa-u.ac.jp, ††vel810@gin.ec.t.kanazawa-u.ac.jp

Abstract In this paper, we propose a fast motion vector detection algorithm using steepest descent method which is one of gradient methods. Proposed method is capable of avoiding the local minima problem by employing adaptive initial vector setting and hierarchical algorithm, and obtaining improvement of predictive performance and computational complexity. From simulation, we compare predictive performance and computational complexity with other methods, and show effectiveness of the proposed method. And also, we show performance of proposed method in the case of applying it to interlaced images.

1. まえがき

動き補償予測において、ブロックマッチング法を用いた動きベクトル検出は、アルゴリズムが簡単であることから、国際標準規格である MPEG-1, MPEG-2 をはじめとして、多くの符号化方式で採用されている。探索範囲内の候補ベクトルを網羅的に探索する full search (FS) では、最良の

予測特性を与えるベクトルを検出できる。しかし、膨大な演算量を必要とするという問題点があるため、動きベクトル検出の高速化のために three-step search (3SS), one-dimensional full search (1DFS), conjugate search (CS), block-based gradient descent search (BBGDS), one-at-a-time search (OTS), one-dimensional gradient descent search (1DGDS) などの手法が提案されてきた [1]~[4]。

3SS や 1DFS [1] は, FS に比べて search window 内の探索点数を削減することで高速化を実現した手法である。しかし, 演算量は search window の大きさに依存するため, サイズが大きく, 大きな動きをもつ画像に適用するには不向きである。CS や BBGDS [2], OTS [3] は search window の概念をもたず, 網羅的探索も必要としないため, 演算量の削減効果は大きい。しかし, その一方で, これらの手法を適用すると, 局所的な解に陥りやすく, 予測特性が劣化するという難点がある。そのため, 1DGDS [4] 等のように, この局所解問題を回避する手法を組み込んだ方式も提案されている。

また, 標準テレビ画像や HDTV といったインターレース構造を持つ画像においては, その構造上, インターレース構造を持たない画像と同様に, 単純に動きベクトル検出を行っても効果的ではない。特に勾配法を用いた検出法においては, インターレース構造によって, 評価関数が単調に変化することは少なく, 予測特性が大きく低下してしまう。そのため, インターレース画像に対しては, フレーム予測だけでなくフィールド予測も行われる。

本論文では, 勾配法の一つである最急降下法 [5] に適応的な初期値の設定法と階層化手法を適用する手法を提案する。本手法により, 局所解に陥る問題を回避し, 良好な予測特性を維持しつつ, 演算量を大幅に削減する。シミュレーションにより, 他方式との比較を行うことで提案法の有効性を示す。また, インターレース構造を持つ画像に対しても, フレーム予測およびフィールド予測の両方に提案法を適用することで, 有効性を検証する。

2. 最急降下法を用いた動きベクトル検出法 [6]

2.1 動きベクトル検出基本アルゴリズム

2.1.1 定義

動きベクトルの検出にあたり, 以下の定義を行う。

- 現フレームの点 (x, y) の輝度値を $I_k(x, y)$, 前フレームの点 (x, y) の輝度値を $I_{k-1}(x, y)$ と表し, 検出すべき動きベクトル (a_x, a_y) を \mathbf{a} で表す。

- 前フレームから現フレームの点 (x, y) で予測した輝度値を $I_{k-1}(x, y, \mathbf{a})$ とすると, 予測誤差 $DPD(x, y, \mathbf{a})$ は次の式で表される。

$$\begin{aligned} DPD(x, y, \mathbf{a}) &= I_k(x, y) - I_{k-1}(x, y, \mathbf{a}) \\ &= I_k(x, y) - I_{k-1}(x - a_x, y - a_y) \end{aligned} \quad (1)$$

- 予測誤差の 2 乗値のブロック内総和をとって評価関数 E_{dp} とする。

$$E_{dp}(\mathbf{a}) = \sum_x \sum_y [DPD(x, y, \mathbf{a})]^2 \quad (2)$$

2.1.2 基本アルゴリズム

提案する動きベクトル検出法の基本アルゴリズムは 3 つのステップからなる。各ステップをステップ 1 ～ ステップ 3 とし, 以下で説明する。

ステップ 1 : 勾配 (微係数) の算出

動きベクトルを検出する前フレーム内のブロックに対して, 予測誤差で構成する評価関数が最も減少する方向 (勾配が最大となる方向) は, 評価関数の微係数で与えられる。ブロックの動きベクトル $\mathbf{a} = (a_x, a_y) = (x - \hat{x}, y - \hat{y})$ に対する微係数の x 成分は式 (3) で求められる。 y 成分についても同様である。

$$\begin{aligned} \frac{\partial E_{dp}(\mathbf{a})}{\partial a_x} &= \frac{\partial E_{dp}(\mathbf{a})}{\partial \hat{x}} \\ &= 2 \sum_x \sum_y [DPD(x, y, \mathbf{a}) \cdot \frac{\partial DPD(x, y, \mathbf{a})}{\partial \hat{x}}] \end{aligned} \quad (3)$$

ここで, 輝度値の予測誤差 (DPD) の勾配を直接求めることは困難であるため, 隣接した画素値の空間勾配を用いて次の式で近似する (y 方向についても同様)

$$\frac{\partial DPD(x, y, \mathbf{a})}{\partial \hat{x}} = -\frac{1}{2} [I_{k-1}(\hat{x} + 1, \hat{y}) - I_{k-1}(\hat{x} - 1, \hat{y})] \quad (4)$$

ステップ 2 : 次元探索

ステップ 1 で求めた探索方向に向けて, 一定のステップ幅で次元探索を行う。探索開始点のベクトルを $\hat{\mathbf{a}}$ とし, 探索ベクトル \mathbf{a} は次式で算出される。

$$\mathbf{a} = \hat{\mathbf{a}} - \lambda \cdot \frac{\partial E_{dp}(\hat{\mathbf{a}})}{\partial \mathbf{a}} / \left\| \frac{\partial E_{dp}(\hat{\mathbf{a}})}{\partial \mathbf{a}} \right\| \quad (5)$$

探索方向に沿って評価関数 E_{dp} を算出し, これが減少から増加に転じた時点で, 最小値が求めたものとして探索を終了する。本稿では, λ を 0.5 画素精度としている。

また, 内挿処理を簡単にするために, 検出するベクトル精度を半画素精度とする。

ステップ 3 : 繰り返し処理

ステップ 1, 2 を逐次的に繰り返す。ステップ 2 においてベクトルが変化しなくなるか, 規定の繰り返し回数に達した時点で, 処理を終了する。

2.2 局所解回避の手法

2.2.1 適応的初期値の設定

最急降下法は初期値依存性が強いいため, 初期値が適切でないとい局所的な解に陥る可能性が高くなる。そこで,

uncovered background など生じる静止領域に対応するために零ベクトル、また既に検出を終えた時間的に過去のブロックの中から、空間的または時間的に最も相関が強いブロックの動きベクトルを参照し、適応的に初期値を設定する。本稿では、以下に挙げる4つのベクトルを初期値の候補とする。

- 零ベクトル (静止領域)
 - 現在処理しているフレーム中の左隣のブロックの動きベクトル (水平方向の空間的な相関が最も高い)
 - 現在処理しているフレーム中の上隣のブロックの動きベクトル (垂直方向の空間的な相関が最も高い)
 - 一つ前に処理されたフレーム中で同じ位置にあるブロックの動きベクトル (時間的な相関が最も高い)
- この中で最も小さい予測誤差を与えるベクトルを選び、初期ベクトルとする。

2.2.2 最終ベクトルの修正

輝度値の空間勾配を利用する最急降下法では、画像に含まれる雑音の影響で正確な微係数を得られない場合があり、特に最適解の周辺ではその影響が強くなると考えられる。そこで最終ベクトルに対して、周囲8方向を含めた9通りのベクトルについて評価関数を計算し、それが最小となるベクトルを選択する。

2.3 階層化手法

階層画像を作成し、それに最急降下法を適用することで、大きいベクトルを高速に探索できるとともに、雑音の影響を回避する手法を提案する。この手法は、評価関数が滑らかな上位階層で大まかな動きを求め、下位階層で徐々に正確な動きへと修正していく手法である。

今回用いた階層化手法アルゴリズムについて、ステップごとに説明する。

ステップ A

処理画像に対して、画素値の平均化をとまなう4:1のサブサンプリングを行い、階層数 n のピラミッドを生成する。

ステップ B

最下位階層に対して前述した「適応的な初期値の設定」を行う。これにより得られたベクトルを最上位階層の画像サイズにあわせて縮小し、初期ベクトルとする。

ステップ C

最上位階層で基本アルゴリズムにより動きベクトルを検出し、それを2倍にスケールしたものを、一つ下の階層の初期ベクトルとする。これを各階層について行い、最下位階層まで繰り返す。

ステップ E

最下位階層で得られたベクトルに対して最終ベクトル

の修正を行う。

2.4 インターレースを考慮した動きベクトル検出法

2.4.1 順次走査とインターレース走査

動画像の走査方式には順次走査とインターレース走査の2種類がある。順次走査のフレームは全て同じ時間にサンプリングされた画素で構成されているのに対し、インターレース走査のフレームは、異なった時間でサンプリングされた画素が、1ライン毎に交互に繰り返される。この異なった時間にサンプリングされた画像を、それぞれトップフィールド、ボトムフィールドと呼ぶ。

2.4.2 フレーム予測とフィールド予測

ブロック単位の動き補償予測では、各対象ブロックにおける動きベクトルを算出し、動きベクトルでシフトした位置の前フレームのブロックにより予測が行われる。

順次走査の画像に対する動き補償予測には、フィールド構造を考慮しないフレーム予測が用いられる。それに対し、インターレース走査の場合、サンプリングされた時間によりフィールドに分割し、それぞれのフィールド毎に予測を行うフィールド予測も用いられる。

本論文では、一般的なフレーム予測方式、フィールド予測方式を以下の様に定義し、提案法のアルゴリズムを適用する。

- フレーム構造フレーム予測

対象フレームのマクロブロックを、動きベクトルによりシフトした位置の前フレームのマクロブロックで予測する方式。

- フレーム構造フィールド予測

フレームをフィールドに分割し、予測対象とするフィールドのマクロブロックに対して、前フレームのトップフィールド、ボトムフィールド両方からそれぞれ動きベクトルを検出する。検出した動きベクトルのうち予測誤差が小さいベクトルをそのフィールドの動きベクトルとする。決定した動きベクトルにより、各フィールド独立に予測し、合わせたものを最終的な予測マクロブロックとする。フィールドごとに動きベクトルを持つため、マクロブロック毎に2本の動きベクトルを持つことになる。

- フィールド構造フィールド予測

フレームをフィールドに分割し、フィールドをサンプリング時間順に並べ、1つのシーケンスとして動き補償予測を行う。したがって、トップフィールドは前フレームのボトムフィールドで予測し、ボトムフィールドは同フレームのトップフィールドにより予測することになる。この予測方式もマクロブロック毎に2本の動きベクトルを持つ。

3. シミュレーション実験

計算機によるシミュレーションを行い、予測特性と演算量を比較して提案法の特徴を示す。予測特性の比較には PSNR を用いる。演算量については、150 フレームの平均演算量を加算回数と乗算回数で見積もったものと、実測処理時間 (PC - Pentium4 2.4GHz) で比較する。

シミュレーション条件として、動きベクトル検出を行うブロックサイズは、一般的なマクロブロックサイズである 16×16 画素に設定する。ただし、提案法の階層化手法においては、階層画像のダウンサンプリングにあわせ、ブロックもダウンサンプリングしたサイズを用いる。提案法における階層数は 3 階層に設定した。また、最終的な動きベクトル精度は半画素単位とする。

提案法における次元探索の繰り返し回数は、それ以上行っても予測特性向上の効果が望めない値として、経験的に 7 回に設定する。また次元探索のステップは階層全てにおいて半画素単位とした。テスト画像には、“bus”(bus)、“flower garden”(flow)、“football”(ftbl)、“susie”(ssie) の 4 種類を用いる。(表中では括弧内の略称で表記)

3.1 提案法の特徴

まず CIF 画像 (順次走査, 352×288 画素) に対して提案法を適用し、その特性を示す。提案法の基本となる最急降下法と、それに適応的な初期値の設定、最終ベクトルの修正および階層化手法をそれぞれ適用した際の予測特性と演算量を示す。表 1 に各テスト画像 150 フレームにおいて、適応的初期値と最終ベクトル修正をそれぞれ適用した場合の予測画像の平均 PSNR を示す。また、図 1 にテスト画像 Football の各フレームにおける予測画像の PSNR を示す。

表 1 より、適応的な初期値を設定することで、全テスト画像における平均で 1.92 dB の大きな改善が得られている。これは初期値依存性の高い勾配法において、適応的初期値設定法を適用することにより、初期値に起因する局所解への陥りを回避していることを示している。また図 1 より、画像ごとの特徴をみても“Football”のような大きな動きを含んでいるシーケンスにおいて、特に動きの大きいフレームで適応的初期値設定の改善効果が大きいことがわかる。このことから、探索開始点が最適解から遠く離れている場合に、局所解に陥りやすい問題を、適応的初期値設定を行うことで回避する効果が得られていることがわかる。

また表 1 から、さらに最終ベクトルの修正を加えることにより平均値として 0.26 dB の改善が得られている。適応的初期値設定ほど大きな効果は得られていないが、

表 1 提案法の予測特性

Table 1 predictive performance of proposed methods

| 方式 | PSNR (dB) | | | | |
|-----------|-----------|-------|-------|-------|-------|
| | bus | flow | ftbl | ssie | 平均 |
| 最急降下法 | 23.57 | 26.41 | 26.59 | 37.99 | 28.64 |
| 適応的初期値 | 27.04 | 27.74 | 28.24 | 39.20 | 30.56 |
| 初期値+最終修正 | 27.29 | 27.99 | 28.38 | 39.63 | 30.82 |
| 提案法 (階層化) | 27.53 | 28.01 | 28.78 | 39.64 | 30.99 |

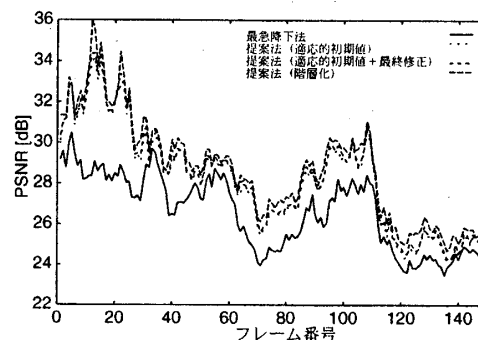


図 1 適応的初期値、最終修正および階層化手法の予測特性 (Football)

Fig. 1 predictive performance of adaptive initial vector setting, final vector adjustment and hierarchical algorithm (football)

全ての画像において平均的に改善されていることがわかる。これは雑音の影響により収束点付近で正確な微係数が得られない問題を改善している。

さらに適応的な初期値設定と最終ベクトルの修正に加え、階層化手法を適用した場合、平均値として 0.17 dB の改善が得られた。階層化手法を用いることにより、上位階層において雑音の影響が少ない大まかな勾配が得ることが可能となり、局所的な勾配がブロック全体の勾配と異なることによる局所解への陥りを回避している。

次にフレームあたりの平均演算量を表 2 に示す。表より、適応的に初期値を設定した方が、演算量が減少していることがわかる。この手法では、初期値を選択するための付加的な演算量が必要となるが、与えられた初期値が収束点に近くなれば、次元探索の回数を減らすことができる。特に大きな動きをもつ画像では、初期値を選択するためのオーバーヘッドよりも、探索回数を減少させるメリットの方が大きくなると考えられる。最終ベクトルの修正は多数の評価関数を余分に計算する必要があり、評価関数の計算は演算量が大きい処理であるため、演算量の増加が目立つ結果となった。

階層化手法では、階層化画像を生成する必要があるために演算量が増加する。しかし、上位レベルではブロックサイズが小さいため、一回の探索に要する演算量の負担

表 2 提案法の演算量

Table 2 computational complexity of proposed methods

| 方式 | 実測時間 (sec/フレーム) | | | | | | | |
|-----------|----------------------------------|-------|-------|-------|------|-----|------|-----|
| | bus | flow | ftbl | ssie | | | | |
| 最急降下法 | 0.012 | 0.012 | 0.014 | 0.011 | | | | |
| 適応的初期値 | 0.008 | 0.009 | 0.010 | 0.009 | | | | |
| 初期値+最終修正 | 0.012 | 0.012 | 0.014 | 0.013 | | | | |
| 提案法 (階層化) | 0.016 | 0.015 | 0.016 | 0.015 | | | | |
| 方式 | 加算・乗算回数 (×10 ⁶ /フレーム) | | | | | | | |
| | bus | | flow | | ftbl | | ssie | |
| 最急降下法 | 加 | 乗 | 加 | 乗 | 加 | 乗 | 加 | 乗 |
| | 3.1 | 1.6 | 3.1 | 1.6 | 4.1 | 2.1 | 2.6 | 1.4 |
| 適応的初期値 | 加 | 乗 | 加 | 乗 | 加 | 乗 | 加 | 乗 |
| | 2.2 | 1.3 | 2.1 | 1.3 | 2.8 | 1.6 | 2.3 | 1.4 |
| 初期値+最終修正 | 加 | 乗 | 加 | 乗 | 加 | 乗 | 加 | 乗 |
| | 3.7 | 2.1 | 3.7 | 2.1 | 4.3 | 2.3 | 3.8 | 2.1 |
| 提案法 (階層化) | 加 | 乗 | 加 | 乗 | 加 | 乗 | 加 | 乗 |
| | 4.5 | 2.5 | 4.4 | 2.5 | 4.9 | 2.7 | 4.5 | 2.5 |

表 3 提案法と他方式の予測特性の比較

Table 3 comparison of predictive performance between proposed method and other methods

| 方式 | PSNR (dB) | | | | |
|-------|-----------|-------|-------|-------|-------|
| | bus | flow | ftbl | ssie | 平均 |
| FS | 27.90 | 28.17 | 28.91 | 39.80 | 31.20 |
| 3SS | 27.21 | 27.72 | 28.89 | 39.28 | 30.78 |
| 1DGDS | 26.25 | 28.05 | 26.98 | 39.50 | 30.20 |
| 提案法 | 27.53 | 28.01 | 28.78 | 39.64 | 31.00 |

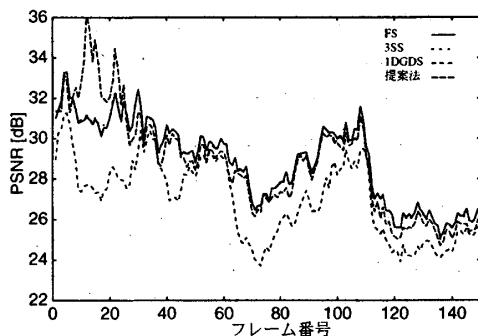


図 2 提案法と他方式の予測特性の比較 (Football)

Fig. 2 comparison of predictive performance between proposed method and other methods (Football)

が小さくなる。また、上位レベルではステップ幅が大きいため、大きい動きに対する探索回数も減少する。この結果、演算量の激しい増加はなかった。

3.2 他方式との比較

次に、提案法 (適応的初期値設定法、最終的なベクトル修正および階層化手法を適用した最急降下法) と他の手法の特性比較の結果を示す。比較する方式として、Full Search (FS)、Three Step Search (3SS)、および提案法と同じく勾配法を用いた高速な動きベクトル検出法である 1DGDS [4] を選択する。

FS の探索範囲は、テスト画像に含まれた動きのほとんどが検出が可能となるように 31 × 31 画素に設定する。したがって、比較する FS の示す予測特性はほぼ上限である。また 1DGDS についても、検出する動きベクトルを半画素精度にしている。

表 4 提案法と他方式の演算量の比較

Table 4 comparison of computational complexity between proposed method and other methods

| 方式 | 実測時間 (sec/フレーム) | | | | | | | |
|-------|----------------------------------|-------|-------|-------|------|-----|------|-----|
| | bus | flow | ftbl | ssie | | | | |
| FS | 1.318 | 1.318 | 1.318 | 1.318 | | | | |
| 3SS | 0.087 | 0.087 | 0.087 | 0.087 | | | | |
| 1DGDS | 0.011 | 0.012 | 0.012 | 0.012 | | | | |
| 提案法 | 0.016 | 0.015 | 0.016 | 0.015 | | | | |
| 方式 | 加算・乗算回数 (×10 ⁶ /フレーム) | | | | | | | |
| | bus | | flow | | ftbl | | ssie | |
| FS | 加 | 乗 | 加 | 乗 | 加 | 乗 | 加 | 乗 |
| | 681 | 341 | 681 | 341 | 681 | 341 | 681 | 341 |
| 3SS | 加 | 乗 | 加 | 乗 | 加 | 乗 | 加 | 乗 |
| | 44 | 22 | 44 | 22 | 44 | 22 | 44 | 22 |
| 1DGDS | 加 | 乗 | 加 | 乗 | 加 | 乗 | 加 | 乗 |
| | 3.1 | 3.3 | 3.3 | 3.5 | 3.5 | 3.7 | 3.1 | 3.3 |
| 提案法 | 加 | 乗 | 加 | 乗 | 加 | 乗 | 加 | 乗 |
| | 4.5 | 2.5 | 4.4 | 2.5 | 4.9 | 2.7 | 4.5 | 2.5 |

図 2 と表 3 に提案法と他方式の予測特性を示す。また、表 4 に実測演算時間と加算・乗算回数を示す。表 3、4 より、提案法は FS と比較して予測特性の劣化を平均値で 0.2 dB に抑えつつ、実測演算時間を約 1.2% に削減することができた。3SS と比較しても、ほぼ同等の予測特性で、実測演算時間を 20% に削減している。

また 1DGDS と比較すると実測演算時間で約 1.5 倍となるが、予測特性では平均で 0.8 dB 良いという結果が得られた。図 2 より、動きが大きい“Football”においては、1DGDS で予測特性の低下がみられる。これは 1DGDS が大きな動きを含む画像においては、局所解に陥る場合が多いことを示している。1DGDS では初期値設定と探索時に可変ステップを用いることで局所解への陥りを回避している。それに対して提案法は、さらに階層化を加えることにより、動きの平滑化と雑音の軽減を行い、局所解への陥りをより回避しているものと考えられる。

“Football”の 10 ~ 30 フレームでは FS の探索範囲を越える非常に大きな動きが含まれており、そのような場合でも提案法で動きの検出が可能となっていることがわかる。したがって、提案法は大きな動きを含むシーケンスにおいて、特に有効な手法といえる。

3.3 インターレース画像への適用

次に 422 画像 (インターレース走査、720 × 480 画素) に対して、提案法を適用し、その特性を示す。動き補償予測法として、フレーム構造フレーム予測、フレーム構造フィールド予測およびフィールド構造フィールド予測の 3 種類の手法に提案法を適用する。比較対象として、FS を適用した場合についてもシミュレーションを行った。このとき FS の探索範囲は 63 × 63 画素に設定した。

図 3 と表 5 にフレーム構造フレーム予測、フレーム構造フィールド予測、フィールド構造フィールド予測の 3 種類の予測方式に FS と提案法を適用した場合の予測特性を示す。また、表 6 に実測演算時間と加算・乗算回数を示す。

各予測方式において FS と提案法を適用した場合を比

表 5 予測方式による予測特性の比較

Table 5 comparison of predictive performance with prediction method

| 方式 | PSNR (dB) | | | | |
|-------------|-----------|-------|-------|-------|-------|
| | bus | flow | ftbl | ssie | 平均 |
| FS(fr.fr.) | 26.90 | 28.70 | 28.02 | 38.39 | 30.50 |
| FS(fr.fi.) | 28.64 | 29.46 | 31.07 | 39.60 | 32.19 |
| FS(fi.fi.) | 27.20 | 23.99 | 32.01 | 39.40 | 30.65 |
| 提案法(fr.fr.) | 25.97 | 28.33 | 27.26 | 37.95 | 29.88 |
| 提案法(fr.fi.) | 27.82 | 29.07 | 29.94 | 39.23 | 31.52 |
| 提案法(fi.fi.) | 26.00 | 23.56 | 30.88 | 39.03 | 29.87 |

fr.fr. : フレーム構造フレーム予測

fr.fi. : フレーム構造フィールド予測

fi.fi. : フィールド構造フィールド予測

表 6 予測方式による計算量の比較

Table 6 comparison of computational complexity with prediction method

| 方式 | 実測時間 (sec/フレーム) | | | |
|-------------|-----------------|--------|----------|--------|
| | bus | flower | football | ssie |
| FS(fr.fr.) | 18.607 | 18.607 | 18.607 | 18.607 |
| FS(fr.fi.) | 37.105 | 37.105 | 37.105 | 37.105 |
| FS(fi.fi.) | 19.132 | 19.132 | 19.132 | 19.132 |
| 提案法(fr.fr.) | 0.060 | 0.058 | 0.063 | 0.061 |
| 提案法(fr.fi.) | 0.096 | 0.097 | 0.107 | 0.100 |
| 提案法(fi.fi.) | 0.059 | 0.060 | 0.061 | 0.061 |

| 方式 | 加算・乗算回数 ($\times 10^4$ /フレーム) | | | | | | | |
|-------------|--------------------------------|-----|--------|-----|----------|-----|------|-----|
| | bus | | flower | | football | | ssie | |
| | 加 | 乗 | 加 | 乗 | 加 | 乗 | 加 | 乗 |
| FS(fr.fr.) | 994 | 497 | 994 | 497 | 994 | 497 | 994 | 497 |
| FS(fr.fi.) | 1920 | 960 | 1920 | 960 | 1920 | 960 | 1920 | 960 |
| FS(fi.fi.) | 960 | 480 | 960 | 480 | 960 | 480 | 960 | 480 |
| 提案法(fr.fr.) | 1.3 | 0.6 | 1.2 | 0.6 | 1.4 | 0.7 | 1.4 | 0.6 |
| 提案法(fr.fi.) | 2.7 | 1.3 | 2.6 | 1.3 | 2.9 | 1.4 | 2.8 | 1.4 |
| 提案法(fi.fi.) | 1.6 | 0.9 | 1.6 | 0.9 | 1.7 | 0.9 | 1.6 | 0.9 |

fr.fr. : フレーム構造フレーム予測

fr.fi. : フレーム構造フィールド予測

fi.fi. : フィールド構造フィールド予測

較すると、表 5 より予測特性の平均値では、FS と比較して提案法は、フレーム構造フレーム予測で 0.62 dB、フレーム構造フィールド予測で 0.67 dB、フィールド構造フィールド予測で 0.78 dB の PSNR の低下で抑えられた。実測演算時間では、表 6 より、3 種類の予測方式において、提案法は FS の約 0.3 % という非常に短い処理時間で動きベクトルの検出が可能となった。

また図 3 より、テスト画像の各フレームの予測特性をみても、特定のフレームで大きく落ち込むフレームはなく、安定した予測特性が得られることが確認できた。

4. む す び

本論文では、最急降下法を基とする高速な動きベクトル検出アルゴリズムを提案した。勾配法における一般的な局所解問題を回避する手法として、適応的初期値設定法と階層化手法を適用し、予測特性の劣化の抑制と処理の高速化を実現した。特に階層化手法は動きの大きい画像に有効であり、FS と比較しても予測特性の大きな劣化なしに、演算量を大きく削減することができた。また、他の勾配法による動き検出アルゴリズムである 1DGDS と

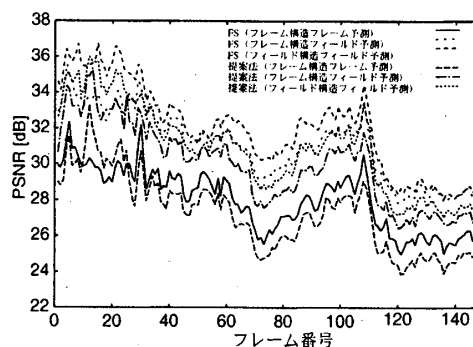


図 3 各フレームにおける予測特性の比較 (Football)

Fig. 3 comparison of predictive performance of each frame (Football)

比較した場合、提案法は演算量の点で一步譲るが、局所解への陥りが少なく、高い予測特性が得られるアルゴリズムであることを検証した。さらに、インターレース画像に対するフレーム予測およびフィールド予測に対する提案法の適用においても、安定した予測特性を示しつつ、大幅な演算量削減を実現することができた。尚、本提案法のアルゴリズムは低消費電力プロセッサとしてチップ製作され、現在は性能を評価中である [7]。

文 献

- [1] Mei-Juan Chen, Liang-Gee Chen, Tzi-Dar Chiueh, "One-Dimensional Full Search Motion Estimation Algorithm For Video Coding", IEEE Trans. Circuits and Syst. Video Technol., Vol.4, No.5, pp.505-509, Oct.1994
- [2] Lurug-Kuo Liu, Ephraim Feig, "A Block-Based Gradient Descent Search Algorithm for Block Motion Estimation in Video Coding", IEEE Trans. Circuits and Syst. Video Technol., Vol.6, No.4, pp.419-422, Aug.1996
- [3] R.Sirinivasan and K.R.Rao, "Predictive coding based on efficient motion estimation", IEEE Trans. Commun., Vol.COM-33, pp.888-896, Aug.1985
- [4] Oscar T.-C. Chen, "Motion Estimation Using a One-Dimensional Gradient Descent Search", IEEE Trans. Circuits and Syst. Video Technol., Vol.10, No.4, pp.608-616, June 2000
- [5] J. コワリック, M.R. オスポーン, "非線形最適化問題", 培風館 (1970)
- [6] 高林 将範, 今村 幸祐, 橋本 秀雄, "勾配法に基づく動きベクトル検出の高速化", 信学技報, IE2001-74, pp.1-6, SEP, 2001
- [7] M.Miyama, O.Tooyama, N.Takamatsu, T.Kodake, K.Nakamura, A.Kato, J.Miyakoshi, K.Imamura, H.Hashimoto, S.Komatsu, M.Yagi, M.Morimoto, K.Taki and M.Yoshimoto, "An Ultra Low Power, Realtime MPEG2 MP@HL Motion Estimation Processor Core with SIMD Datapath Architecture Optimized for Gradient Search Algorithm", Proc. of CICC2002 (Custom IC Conference 2002), pp.9.6.1-9.6.4, May 2002.