

探索位置の予測に基づく時系列データの高速一致区間検出 - 参照区間自由時系列アクティブ探索法 (RIFAS 法) の改良 -

大盛善啓

株式会社 東芝 研究開発センター

本研究では 2 つの時系列信号列から任意の位置の類似区間を高速に探索する手法を提案する。本研究で扱う問題は 2 つの時系列信号による 2 次元探索空間を類似点か探索する必要がないスキップ可能領域で埋め尽くすこととして定式化できる。従来手法の RIFAS 法はある点の類似度が求めればその周辺にスキップ可能領域が存在することを示して高速化した。スキップ可能領域の重なりが広く効率が悪かった。提案手法は RIFAS 法を基本とするが、スキップ可能領域の重なりが少なくなるように探索点の位置を予測して効率を改善する。映像探索の実験では探索処理時間を大幅に短縮して 50 分の映像を約 1 秒で探索でき、番組コーナー検出への応用では高精度にコーナータイトル映像を検出できる。

A Fast Same Interval Retrieval for Time-Sequence Data Based on Predictions of Search Window Positions - Improvement of Reference Interval-Free Time-Series Active Search (RIFAS) -

Yoshihiro Ohmori

Toshiba Corporation Corporate Research & Development Center

In this paper, a fast same interval retrieval method between two time-series data is proposed. The task of this paper can be formulated by covering all over the search space with similar points or skips areas. Existing method (RIFAS) found the existence of skip area around the point which similarity is already known, but considerable parts of the skip area overlapped. Our method bases on RIFAS method, but can use the skip area more efficiently by predicting a position of search windows. Experimental results show that our method is faster than existing methods, and can search 50 minutes of movie in 1 second. Moreover our method can detect corner title shots accurately.

1. はじめに

記憶媒体の大容量化や信号圧縮技術の高度化によって映像や音楽などを大量に利用できるようになり、欲しい情報を高速に取り出す技術の重要性が高まっている。中でも、時系列信号の探索はきわめて重要な要素技術のひとつである[1]。

これまでに、時系列信号列からテンプレート信号と類似する部分を高速に探す時系列ア

クティブ探索法 (TAS 法) が柏野らによって提案され、コマーシャル (CM) の出現検出などに応用された[2] [3]。

一方で、テンプレートを与えられない問題も多い。例えば 2 つの映像から類似区間を検出して番組のコーナーを検出するにはテンプレートをあらかじめ用意できない。このような参照区間自由探索は膨大な処理を必要とするため実用化が困難だった。

参照区間自由探索の高速化手法として、RIFAS 法 (Reference Interval-Free Time-Series Active Search) が西村らによって提案された[4][5]。RIFAS 法は 2 つの時系列信号列からなる 2 次元探索空間上の 1 点の類似度が求まると、その点の周囲にスキップ可能領域があることを示し、その領域の探索処理を省略して高速化を実現した。しかし、スキップ可能領域のうち半分しか有効でない上に、スキップ可能領域の重なりが広いために効率が悪かった。

提案手法は RIFAS 法を基本とし、スキップ可能領域の重なりを低減して高速化する。提案手法を Square 法と呼ぶ。提案手法は探索処理時間を全探索の約 250 倍、RIFAS 法の約 4.6 倍に高速化できる。検出精度は番組コーナー検出に十分である。

本章の構成は、2 章で提案手法の基本である RIFAS 法を説明し、3 章で課題を明らかにする。4 章で提案手法を詳述し、5 章で映像を用いた実験結果を示し、6 章でまとめる。

2. 従来手法 (RIFAS 法) の概要

RIFAS 法は、音声や映像などの時系列信号を相互に探索して任意の長さの類似区間を高速に検出する[4][5]。

図 1 に RIFAS 法で利用するスキップ可能領域を示す。ここではしきい値 $\theta (0 \leq \theta \leq 1)$ を与えて、しきい値 θ 以下の探索点の処理を省略して探索を高速化することを考える。図 2 において、 $x-y$ 平面は 2 つの時系列信号 X および Y からなる 2 次元の探索空間を表し、探索点 $P_1(x_1, y_1)$ の類似度 $s_1 (0 \leq s_1 \leq 1)$ は、 x 軸上で位置 x_1 の探索窓 W_{1x} と、 y 軸上で位置 y_1 の探索窓 W_{1y} の類似度を表す。類似度は Histogram 差分で算出する。探索点 $P_1(x_1, y_1)$ の類似度 s_1 が求まると、探索点 P_1 を中心とする正方形が 45 度傾いたスキップ可能領域が求まる。 D を探索窓の長さ、 $\text{floor}(\)$ を小数点以下を切り捨てる関数とすると、正方形の中心から頂点までの長さ w は以下の式で表せる。

$$w = \text{floor}(D(\theta - s_1)) + 1 \quad \dots (1)$$

図 2 にスキップ可能領域発生の原理を示す。探索点 $P_1(x_1, y_1)$ において探索窓 W_{1y} をテンプレートとみなせば、時系列アクティブ探索法の原理により、探索窓 W_{1x} の類似度は $x_1 \pm (\text{floor}(D(\theta - s_1)) + 1)$ の区間内でしきい値 θ を超えない。ここで探索窓 W_{1y} を Ds_3 移動する場合を考えると、移動後の類似度は最大で s_3 上がり得る。移動後の探索窓 W_{1y} をテンプレート、移動後の類似度を $s_1 + s_3$ とみなせば、探索窓 W_{1x} のスキップ幅は以下の式で表せる。

$$\tilde{w} = \text{floor}(D(\theta - s_1 - s_3)) + 1 \quad \dots (2)$$

探索窓 W_{1y} を微小に移動して探索不要区間を描画すると図 2 に示した正方形のスキップ可能領域が求まる。

図 3 に RIFAS 法による探索処理の経過を示す。まず直線 $y=0$ 上で時系列アクティブ探索を行い、それぞれの探索点における直角二等辺三角形によって作られる谷を、探索済み領域を管理するための谷の初期値とする。次に、 y 軸方向に最も低い谷の類似度を算出し、二等辺三角形の新たなスキップ可能領域を設定して新たな谷を生成する。この処理を繰り返して探索空間全体をスキップ可能領域で埋め尽くす。ただし、谷の座標が整数にならない場合には隣接する整数座標の探索点を 2 つ設定し、探索点の類似度がしきい値 θ 以上の場合には類似点として座標値を出力する。

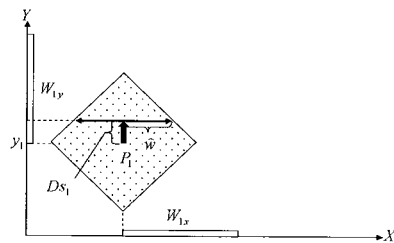


図 1 RIFAS 法が利用するスキップ可能領域

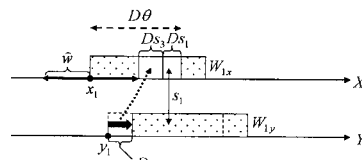


図 2 RIFAS 法の原理

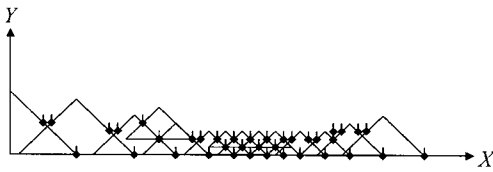


図 3 RIFAS 法の探索処理の経過

3. 従来手法の問題

西村らが示したスキップ可能領域の本来の形状は正方形だが、RIFAS 法では半分だけを利用し、しかもスキップ可能領域の重なりが広く無駄が多かった。

原典にはこれらの理由が明記されていないためここで考察する。参照区間自由探索は、2次元の探索空間をスキップ可能領域か類似点で埋め尽くす問題として定式化できる。もれなく探索するためにはスキップ可能領域を連続させなければならない。そのためには、次の探索点を探索済み領域の境界線を含む内部に設定すれば、次の探索点によるスキップ可能領域または類似点は探索済み領域と必ず連続する。次の探索点は探索済み領域の境界線上であればどこに設定してもよいが、探索済み領域の形状が複雑になると管理が困難になるため、RIFAS 法のように y 軸の正方向にのみ成長させるのが一般的だろう。この場合、スキップ可能領域のうち探索点から y 軸に関して負の領域は必ず探索済み領域と重なるため、有効なスキップ可能領域は正方形の半分の直角二等辺三角形となる。

RIFAS 法にはこのほかにも効率の悪い処理があった。例えば、谷の座標が整数にならない場合には隣接する座標に谷を 2 つ生成するため、管理点が増えて処理量を増加させた。また、新たなスキップ可能領域の内部に谷が含まれることがあるためこれを検索して削除したり、新たなスキップ可能領域が探索済み領域のどの辺と交差するかを検索したりする処理を含む。そのため、全探索や時系列アクティブ法と比較して、1 回の照合処理にかかる処理量が大きかった。

4. 提案手法

本章では、これらの問題を解決する新たな手法を提案する。この手法を Square 法と呼ぶ。Square 法では、探索点を探索済み領域の外部に設定すれば、正方形のスキップ可能領域の全体を有効に利用できることに着目する。

4.1. 提案手法の原理

図 4 に Square 法が利用するスキップ可能領域を示す。Square 法では、通常の探索点 $P_1(x_1, y_1)$ のほかに、探索済み領域の外部に探索点 $P_2(x_2, y_2)$ を設定する。探索点 P_2 の位置は、 P_1 によるスキップ可能領域と P_2 によるスキップ可能領域が重なるように、付近の類似度から予測する。もしも予測に失敗してスキップ可能領域が重ならない場合には、探索点 P_2 の位置を探索点 P_1 に近づけて重なりを再度試みるバックトラック処理を行う。

スキップ可能領域の境界線はスキップ可能領域に含まれないことに注意が必要である。境界線から 1 サンプル内部はスキップ可能領域に含まれる。そこで、探索点 P_1 によるスキップ可能領域と探索点 P_2 によるスキップ可能領域を 1 サンプル以上重ねてスキップ可能領域を連続させる。

同様の発想は柏野らによって提案されたオーバースキッピングによる時系列アクティブ探索法の高速度化手法 (OTAS 法と呼ぶ) で提案されている [6]。OTAS 法が扱う問題はテンプレートマッチングであり Square 法とは異なる。しかし、探索点を 2 つ使って探索済み領域の外部に探索点を設定する原理は参照区間自由探索にも応用できる。

図 5 に OTAS 法の原理を示す。探索点 T_0 におけるテンプレートとの類似度を s_0 とすると、スキップ幅は $D(\theta - s_0) + 1$ となる。探索点 T_p を適当な位置に設定してテンプレートとの類似度 s_p を求めると、スキップ幅は $D(\theta - s_p) + 1$ と求まる。2 つのスキップ幅が重なる場合は探索点 T_0 から探索点 T_p までの全区間で探索処理を省略できる。

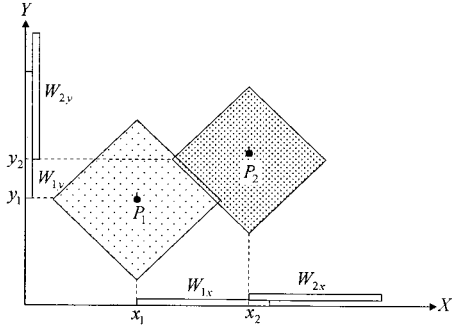


図 4 提案手法が利用するスキップ可能領域

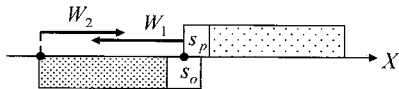


図 5 オーバースキッピングの原理

4.2. 提案手法のアルゴリズム

図 6 を用いて Square 法のアルゴリズムを説明する。 x - y 平面は 2 つの時系列信号 X および Y からなる 2 次元の探索空間を表し、探索窓幅を D 、しきい値を $\theta (0 \leq \theta \leq 1)$ とする。

(1) 谷の初期値の準備：直線 $y = 0$ についてオーバースキッピングによる時系列アクティブ探索法を行い、 N 個の探索点 $P_i(x_i, 0) (0 \leq i \leq N)$ 、スキップ可能幅 w_i および $N-1$ 個の探索点 P_{i+1} と P_i の距離 h_i を求める。

(2) 谷の初期値の算出：探索点 P_i および P_{i+1} を底辺の中心とする高さ w_i および w_{i+1} の 2 つの直角二等辺三角形が作る谷 $v_j(x_j, y_j)$ を以下の式で求める。ただし、 $\text{floor}(\)$ を小数点以下の切り捨て関数とする。

$$\begin{aligned} x_j &= w_i - \text{floor}((w_i + w_{i+1} - h_i)/2) + x_i \\ y_j &= \text{floor}((w_i + w_{i+1} - h_i)/2) \quad \dots (3) \end{aligned}$$

(3) 谷のソート：すべての谷を x_i の小さい順にソートする。

(4) y 軸上で最小の谷を検索：すべての谷の中から y_j が最小の谷 $v_{\min}(x_{\min}, y_{\min})$ を検索する。 $y_{\min} > N - D$ なら探索を終了し、そうでなければ次のステップに進む。 $y_j = y_{\min}$ の谷が複数ある場合にはどれかひとつを選ぶ。

(5) 類似度算出：谷 v_{\min} の類似度 s_{\min} を算出する。

(6) 類似点出力：類似度 s_{\min} がしきい値 θ を上回る場合には座標値と類似度を出力し、谷 $(x_{\min} - 1, y_{\min} + 1)$ 、谷 $(x_{\min}, y_{\min} + 1)$ および谷 $(x_{\min} + 1, y_{\min} + 1)$ を新たに生成してステップ 9 に進む。ただし新たに生成する谷が既存の谷と重なる場合は谷を生成しない。

(7) スキップ可能領域の計算：類似度 s_{\min} がしきい値 θ 以下なら探索点 $P_k(x_{\min}, y_k)$ (ただし $y_k > y_{\min}$) を設定する。谷 v_{\min} 周辺の類似度は一定と仮定して、探索点 P_k によるスキップ幅 w' を時系列アクティブ探索法の原理により以下の式で予測する[3]。

$$w' = \text{floor}(D(\theta - s_{\min})) + 1 \quad \dots (4)$$

探索点 P_k の y 座標値を $y_k = y_{\min} + w' - 1$ とする。次に、探索点 P_k の実際の類似度 s_k を算出して実際のスキップ幅 w を式(4)で求め、探索点 P_k と谷 v_{\min} の距離を w_0 とすると、 $w < w_0 + 1$ を満たすまで探索点 P_k を谷 v_{\min} に 1 ずつ近づけるバックトラック処理を行う。

(8) 新たな谷の生成：谷 $v_{\min-1}$ および谷 v_{\min} が生成する山と、探索点 P_k によるスキップ可能領域とが谷を作る場合には新たな谷 $v_{\text{new}}(x_{\text{new}}, y_{\text{new}})$ の座標値を以下の式で求める。

$$\begin{aligned} x_{\text{new}} &= x_{\min} - h_{\min} \\ y_{\text{new}} &= y_{\min} - h_{\min} \quad \dots (5) \end{aligned}$$

谷 $v_{\min+1}$ および谷 v_{\min} が作る新たな谷についても以下の式で同様に座標値を求める。

$$\begin{aligned} x_{\text{new}} &= x_{\min} + h_{\min} \\ y_{\text{new}} &= y_{\min} + h_{\min} \quad \dots (6) \end{aligned}$$

(9) 谷 v_{\min} を削除してステップ 3 に戻る。

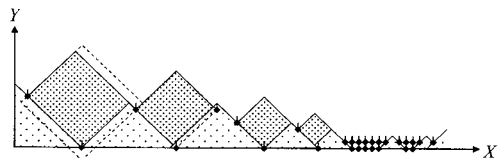


図 6 提案手法の探索処理の経過

5. 実験

提案手法を計算機に実装して探索処理時間と探索精度を測定する実験を行う。実験では2つの異なる映像信号列から類似区間を検出して探索処理時間を調べる。さらに番組コーナー検出に応用する場合の探索精度を調べる。

従来手法との差をあきらかにするため、全探索、時系列アクティブ探索法 (TAS 法)、参照区間自由時系列アクティブ探索法 (RIFAS 法) も実装して提案手法と比較する。全探索とはすべての探索点を照合する手法で、TAS 法は、 y 軸上の探索窓をテンプレートとみなして x 軸方向に時系列アクティブ探索法で探す処理を、 y 軸上の探索窓をひとつずつ進めながら行う手法である。RIFAS 法は 2 章で説明した手法である。各手法とも基本原理以外の最適化は施さない。

実験では別の日に録画した約 50 分の番組映像 2 つを使用する。映像には CM を含まない。映像は MPEG-2 形式で録画して、約 0.5 秒に 1 枚ある I-picture のみを抜き出して符号化する。符号化は文献[4]と同様に、フレーム画像全体の平均輝度を 16 段階で求めた値と、フレーム画像を 9 分割した各領域の平均輝度を、各領域の平均輝度の最大値で割った値を 4 段階で求めた値を使って、1 つの I-picture を $0 \sim 2^{22} - 1$ の値に符号化する。各ビンのしきい値は予備実験によって値が分散するように求める。実験環境は Intel Pentium 4, 3.06GHz, 2.00GB RAM, MS Windows XP であり、プログラムは C++ で実装する。

5.1. 特徴量

映像から特徴量を算出してベクトル量子化するのに必要な時間と、その符号を用いた場合の検出性能を調べる。

5.1.1. 符号化処理時間

約 50 分の MPEG-2 映像から I-picture を抜き出して特徴量を算出するのに約 2 分かかった。特徴量を算出する部分を除いて処理時間を測定したところ、差は検出できなかった。よって、映像の取り込み時に特徴量の算出を行っても処理負荷をほとんど上げない。

5.1.2. 特徴量の質

図 7 に実験映像の類似度分布の一例を示す。 x 軸と y 軸は映像 1 および映像 2 の I-picture のフレーム番号、 z 軸は類似度を表す。探索窓幅は 30 フレーム (15 秒に相当) とした。図示しないが 1 つの I-picture を 16 段階で符号化した場合の類似度分布と比較すると、実験で用いる特徴量は映像が類似しない領域では類似度が低く、類似する領域では類似度が急激に上昇した。この性質は類似領域を分離するのに適する。さらに、しきい値を 0.7 程度に設定すればほとんどの領域がスキップ可能領域となって高速化にも適する。

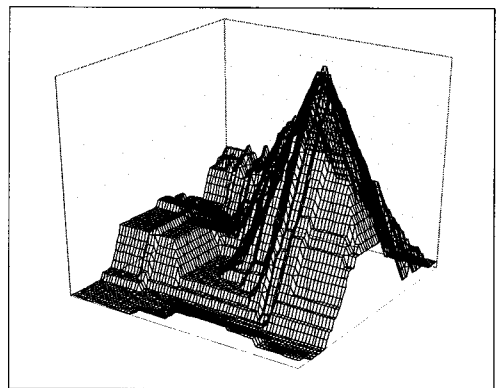


図 7 実験データの類似度分布

5.2. 探索処理時間

処理時間は、しきい値 θ 、信号列の長さ N 、探索窓の長さ D 、信号列の種類、の各パラメータに依存する。まず一般的な使用条件での探索処理時間を比較し、次にパラメータが処理時間に与える影響を調べる。評価にあたり、どの手法の検出結果も同一であることも確認する。

5.2.1. 一般的な使用条件での比較

表 1 に探索時間と照合回数を示す。しきい値 $\theta = 0.7$ 、信号列長 $N = 6000$ [frame]、探索窓長 $D = 30$ [frame] (15 秒に相当) とした。提案手法との処理時間比、照合回数比も示す。Square 法は、全探索の 253 倍、TAS 法の 13 倍、RIFAS 法の 4.6 倍高速だった。

全探索と TAS 法は、照合回数比ほど処理時間比が大きくなかった。Square 法は探索済み領域の管理や新たな谷の座標値を計算するために、1 回の照合にかかる時間が全探索や TAS 法より長いためと考えられる。逆に、RIFAS 法は照合回数比よりも処理時間比の方が大きかった。RIFAS 法は、新たに設定するスキップ可能領域に含まれる谷を削除する検索や、探索済み領域のどの辺と新たな谷を作るかの検索を含む。これらの処理が不要な Square 法のシンプルさが高速化に貢献したと考えられる。

図 8 に探索済み領域を管理する谷の数が探索処理中にどのように推移するかを示す。横軸は処理回数、縦軸は谷の数を表し、上側のデータ系列は RIFAS 法、下側のデータ系列は Square 法の谷の数を示す。Square 法の方が RIFAS 法より谷の数が大幅に少なく、谷の数の比の平均は 0.38 だった。谷の数の低減も高速化に貢献したと考えられる。

表 1 探索時間と照合回数の実験結果

	time [s]	time ratio	count [$\times 10^4$]	count ratio
exhaustive	308	253	3600	478
TAS	16	13	173	23
RIFAS	5.6	4.6	16.2	2.2
Square	1.2	1.0	7.5	1.0

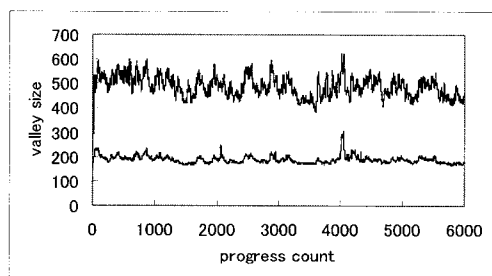


図 8 谷の数の比較

5.2.2. しきい値

図 9 にしきい値を変化させた場合の探索処理時間を示す。信号列長 $N = 6000$ [frame]、探索窓長 $D = 30$ [frame] (15 秒に相当) とした。横軸はしきい値、縦軸は処理時間 [秒] を対数目盛で表す。Square 法はしきい値によらず最速だった。全探索の処理時間はほぼ一定で、その他の手法はしきい値が大きいくほど処理時間が減少するが、しきい値が大きくなると処理時間の減少率が低下した。

図 10 にしきい値を変化させた場合の照合回数を示す。照合回数も処理時間と同様の曲線だった。しかし、RIFAS 法は他の手法と比較して照合回数の減少ほど処理時間が減少しなかった。

図 11 に Square 法で発生したバックトラックの回数を示す。バックトラックの回数は照合回数と比較して非常に小さかった。しきい値が大きくなるとバックトラックの回数と照合回数の比は 1 割程度まで上昇した。しかしバックトラックの発生しない他の手法よりも Square 法は高速であり、バックトラックが高速化に与えた影響は軽微だと考えられる。

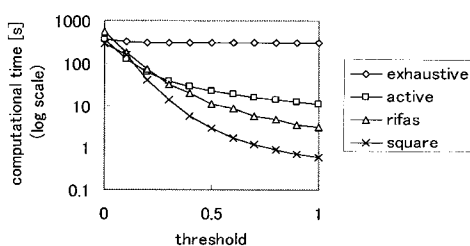


図 9 探索時間としきい値の関係

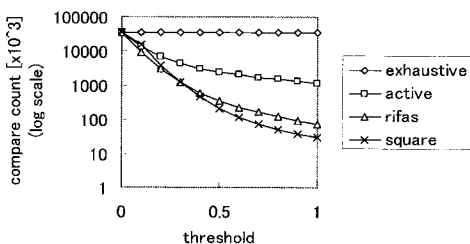


図 10 照合回数としきい値の関係

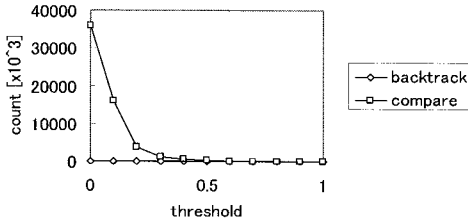


図 11 バックトラック数としきい値の関係

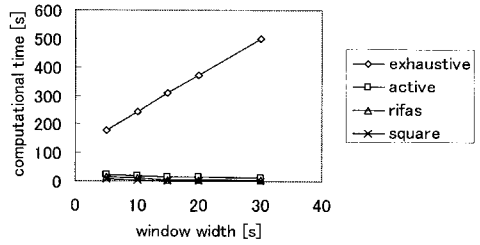


図 13 探索時間と探索窓幅の関係 - 全手法

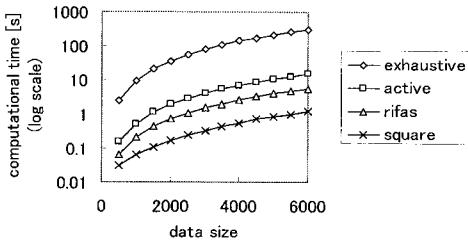


図 12 探索時間と信号列長の関係

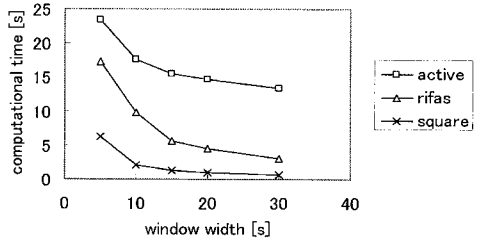


図 14 探索時間と探索窓幅の関係-全探索以外

5.2.3. 信号列長

図 12 に信号列長を変化させた場合の探索処理時間を示す。しきい値 $\theta = 0.7$ ，探索窓長 $D = 30$ [frame] (15 秒に相当) とした。横軸は信号列長，縦軸は処理時間を対数目盛で表す。どの手法も信号列長が大きくなれば探索時間が長くなるが，信号列長が 0~1500 では増加率が大きかった。信号列長が小さい領域では探索空間の境界処理の影響が大きいためと考えられる。

5.2.4. 探索窓幅

図 13 に探索窓幅を変化させた場合の探索処理時間を示し，図 14 に全探索以外の手法の拡大図を示す。しきい値 $\theta = 0.7$ ，探索窓長 $D = 30$ [frame] (15 秒に相当)，信号列長 $N = 6000$ [frame] とした。横軸は探索窓幅を，縦軸は探索処理時間を表す。全探索は探索窓幅と探索処理時間がほぼ比例した。ヒストグラム計算の時間が探索窓幅と比例するためと考えられる。一方，その他の 3 手法では探索窓幅が長くなると探索処理時間が低下した。

Square 法および RIFAS 法では式(1)に示したように，正方形または二等辺三角形のスキップ可能領域の中心から頂点までの長さは，探索窓幅に比例するから，探索窓幅が長い方が高速化できる。TAS 法も探索窓幅に比例してスキップ幅が長くなるから探索窓幅の長い方が高速化できる。探索窓幅が長くなるとヒストグラム計算時間は増加するが，高速化率がこれを上回ったと考えられる。

5.2.5. 信号列の種類

別の映像でも探索処理時間を調べたところ，処理時間は数%変化するが，他の手法との相対的な関係は変わらず，Square 法がパラメータによらず最速だった。探索処理時間は探索空間にどのくらいの類似領域があるかと類似領域の形状に影響される。探索空間と類似領域の比はしきい値によっても変化できるので，手法間の相対的な関係には影響を与えないと考えられる。この実験からは，類似領域の形状の影響を検出できなかった。

5.3. 番組コーナー検出への応用

Square 法を番組のコーナー検出に応用した場合の探索精度を評価する。番組コーナーの前後に含まれるタイトル映像を探索して番組コーナーを検出する。正解は目視により作成する。

図 15 に類似区間の検出結果を示す。縦軸は映像 1 のフレーム番号、横軸は映像 2 のフレーム番号を表し、類似度が 0.9 以上の点をプロットした。検出結果は傾きが 1 の直線として検出されたが、ごく短い区間であるために図では確認できない。x 軸方向または y 軸方向に複数の類似点がある場合には最も類似度の高い類似点を検出結果として採用した。

実験の結果、8箇所類似区間を検出した。5箇所ある正解のうち実線で囲んだ 4つを検出できた。破線で囲んだ残り 4箇所のうち、2つはコーナータイトル映像ではないが画面構成の酷似する映像であり、残りの 2つは誤検出だった。誤検出した映像区間は画面が均一な輝度の区間で発生しており、もともと特徴量のエントロピーが低かったために誤検出したと考えられ、改善が期待できる。また、検出できなかったコーナータイトル映像は 8秒で探索窓幅より短いために検出されなかった。

別の番組映像でも評価したが、探索窓より長い番組タイトル映像はほぼすべて検出できた。特徴量のエントロピーが低い箇所では誤検出が多かった。画面の動きが少なく特徴量のエントロピーの低い番組や、スポーツ番組のような類似した映像を多く含む番組には適さないと考えられる。

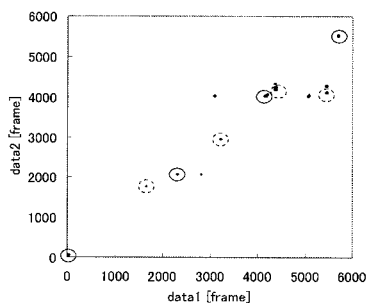


図 15 類似区間の検出結果

6. 結論

本報告では、2つの時系列信号から任意の位置の類似区間を高速に探す Square 法を提案した。Square 法では、アクティブ探索法に基づく参照区間自由探索法 (RIFAS 法) を改良して、スキップ可能領域を効率よく利用可能にして高速化した。実験では、Square 法は全探索の約 250 倍、TAS 法の約 13 倍、RIFAS 法の約 4.6 倍に高速化でき、約 50 分の映像を約 1.2 秒で処理できた。番組コーナー検出への応用実験では、探索窓幅よりも長いコーナータイトル映像を精度よく検出できた。

今後は音声信号などの別の種類の信号列での実験を行い、番組コーナー検出の誤検出抑制を検討する。

参考文献

- [1] 青木 恒, “映像対話検出によるテレビ番組コーナー構成高速解析システム,” 信学論 Vol.J88-D2 No.1 pp.17-27 Feb. 2005.
- [2] 村瀬 洋, V. V. Vinod, “局所色情報を用いた高速物体検索—アクティブ探索法,” 信学論 (D-II), vol.J81-D-II, no.9, pp. 2035-2042, Sept, 1998.
- [3] 柏野 邦夫, 村瀬 洋, “ヒストグラム特徴を用いた音響信号の高速探索方法,” 信学論(D-II), vol.J82-D-II, no.9, pp.1365-1373, Sept. 1999.
- [4] 西村 拓一, 関本 信博, 張 健新, 伊原 正典, 赤坂 貴志, 高橋 裕信, 岡 隆一, “時系列パターンの検索手法,” 信学技法, PRMU99-125, Nov. 1999.
- [5] 西村 拓一, 水野 道尚, 小木 しのぶ, 関本 信博, 岡 隆一, “アクティブ探索法による時系列データ中の一致区間検出—参照区間自由時系列アクティブ探索法—,” 信学論 Vol.J84-D-II No.8, pp. 1826-1837, Aug. 2001.
- [6] 柏野 邦夫, 村瀬 洋, “オーバースキッピングによる時系列アクティブ探索法の高速化,” 音響講義集, pp.445-446, 1999.