

H-024

# アクティブセグメンテーション法：2つの隣接する窓間における類似度の下限に基づく信号分割の高速化手法：

## Active Segmentation: A Fast Segmentation Method Based on the Lower Limit Similarity Between Two Adjacent Windows

大盛 善啓†  
Yoshihiro Ohmori

### 1. まえがき

本稿では、2つの隣接する探索窓をスライドさせながら信号列を分割する処理を高速化する新たな手法であるアクティブセグメンテーション法 (Active Segmentation) を提案する。提案手法では、探索窓間の類似度があらかじめ定めたしきい値を下回り得ない区間で検出処理を省略する。この原理は村瀬らの提案したテンプレート照合の高速化手法であるアクティブ探索法で初めて提案されたが、境界検出にも応用できる。テキスト分割の実験で、提案手法はしらみつぶし処理と比較して数倍に高速化できることを示す。提案手法は、映像ショット分割、話者交代検出、画像エッジ検出などの高速化にも適用できる。

### 2. 境界検出の高速化に関する従来手法

記憶媒体の大容量化や情報圧縮技術の高度化によって映像、音楽、文書などを大量に利用できるようになり、膨大な量の情報から必要な情報を取り出す技術の重要性が高まっている。その中でも、テキスト分割、映像ショット分割、話者交代検出などの情報を意味のある単位に区切る境界検出技術は重要な要素技術のひとつである。本稿では、対話的なパラメータの調整や、多様な条件下での多数回の試行を可能にするために、検出処理の高速化に取り組む。

境界検出手法は、あらかじめ記録された特定の信号を検出する手法と[1]、特徴の急激な変化を検出する手法に大別できる[2][3]。Hearstのテキスト分割手法 (TextTiling 法) は後者の手法の代表例である[3]。TextTiling 法では2つの隣接する探索窓に含まれる単語についてベクトル間の余弦を類似度として求めて、類似度が急激に低下して極小になる位置を話題の境界とした。本研究では、TextTiling 法のように2つの隣接する探索窓をスライドさせながら類似度を計算する境界検出手法を高速化する。

Barnea らの提案した残差逐次検定法 (SSDA: Sequential Similarity Detection Algorithm) は、累積計算による類似度を用いた照合において、類似度があらかじめ定めたしきい値を超えると、類似しないと判定して途中で計算を止めて次の照合に移る高速化手法とした[4]。SSDA 法は TextTiling 法に適用すると1回の検出処理時間を低減できるが、すべての位置で検出するため、高速化率は限定的だった。

村瀬らのアクティブ探索法 (Active Search) は、ヒストグラムインタセクションによる類似度を用いて、類似度があらかじめ定められたしきい値を下回り得ない位置での照合を省略してテンプレート照合を高速化した[5][6]。アクティブ探索法では移動する探索窓はひとつであり、TextTiling のような2つの探索窓が移動する処理にそのまま適用できなかった。

### 3. アクティブセグメンテーション法

図1にアクティブセグメンテーション法の原理説明を示す。矢印線は例えばテキスト列などのベクトル列を表す。(a)は2つの隣接する長さ  $L$  の探索窓  $W_a(T_0)$  と  $W_b(T_0)$  を位置  $T_0$  に設定した場合を表す。 $W_a(T_0)$  と  $W_b(T_0)$  間の類似度  $S_0$  は、以下の式で表されるヒストグラムインタセクション  $S(H_a, H_b)$  により求める。ただし、 $H_a$ 、 $H_b$  を  $W_a(T_0)$ 、 $W_b(T_0)$  に含まれるベクトルのヒストグラムとし、 $H_a = (h_{a1}, h_{a2} \dots h_{aN})$ 、 $H_b = (h_{b1}, h_{b2} \dots h_{bN})$  とする。

$$S(H_a, H_b) = \frac{1}{D} \sum_{n=1}^N \min(h_{an}, h_{bn}) \quad (1)$$

(b)は  $W_a(T_0)$  は固定したままで、探索窓  $W_b(T_0)$  を前方に  $w (w \geq 1)$  サンプル進めた場合を表す。 $W_b(T_w)$  には新たに  $w$  サンプルのベクトルが追加されるが、これらがすべて  $W_a(T_0)$  に含まれない場合には、類似度は  $S_1 = S_0 - w/L$  まで下がり得る。この様子をグラフ  $G_1$  に示す。(c)は探索窓  $W_a(T_0)$  を前方に  $w$  サンプル進めた場合を表す。類似度は  $S_2 = S_1 - 2w/L$  まで下がり得る。この様子をグラフ  $G_2$  に示す。よって探索窓が移動した後の類似度  $S_2$  の下限は、探索窓が移動する前の類似度  $S_0$  を用いて以下の式で表せる。

$$S_2 = S_0 - 2w/L \quad (2)$$

ここで、 $S_2 \geq \theta$  となる探索窓のスキップ幅  $w'$  を考える。式(2)の  $S_2$  に  $\theta$  を代入すると  $\theta \geq S_0 - 2w'/L$  となり、これを变形すると  $w' \leq L(S_0 - \theta)/2$  となる。 $\text{floor}(x)$  を  $x$  を超えない整数を表す関数とすると  $w'$  は以下の式で表せる。

$$w' = \text{floor}(L(S_0 - \theta)/2) + 1 \quad (3)$$

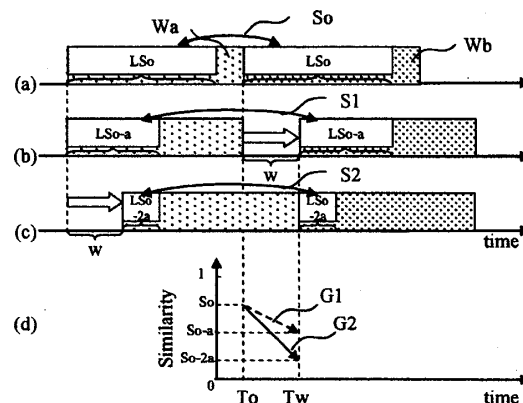


図1 アクティブセグメンテーションの原理

## 4. テキスト分割の実験

### 4.1 実験方法

提案手法で日本語の特許明細書をテキスト分割する実験を行なった。特許明細書は文章の区切りが明示されない場合が多く、テキスト分割の要求が高いと考えられる。実験ではまず日本語の特許明細書を茶笥で構文解析し[7]、名詞や動詞などの自立語のみを取り出して単語列を作成した。単語列は番号をインクリメントしながら先頭から対応付けることによって数値列に変換した。ただし、単語と番号を対応付けたテーブルを作成しておいて、新たに入力した単語が既にテーブルで対応付けられていた場合にはその番号を割り当てて、新たな番号を対応付けないものとする。

まず、提案手法 (active) と TextTiling 法 (cosine) の検出精度を比較した。TextTiling 法は類似度に余弦を、提案手法はヒストグラムインタセクションを用いる。この差がどの程度検出性能に影響するかを調べた。正解は手作業で作成した。

次に、提案手法の高速化の効果を調べた。余弦による検出手法 (cosine)、cosine を SSDA 法で高速化した余弦による検出手法 (ssda)、ヒストグラムインタセクションによる検出手法 (histogram)、アクティブセグメンテーション法 (active) の合計 4 種類の手法を比較した。窓幅は  $w=80$  とし、処理時間は 10 回の試行を平均した。

実装には MS Visual C++ を用い、実行環境は Pentium 4, 3.06GHz, 2.00GB RAM, MS Windows XP とした。

### 4.2 実験結果

図 2 に提案手法と TextTiling 法の検出結果の一部を示す。横軸が時系列、縦軸が類似度、実線が提案手法 (active)、破線が TextTiling 法 (cosine) を表す。類似度の絶対値は異なるが、どちらの手法も極大、極小となる位置は同じだった。TextTiling 法では谷の深さも考慮するが、テキスト分割の検出性能はほとんど変わらないと考えられる。

図 3 にしきい値を変化させた場合の処理時間を示す。cosine と histogram ではしきい値に関わらず処理時間はほぼ一定だった。ssda では  $\theta < 0.4$  で処理時間が低減されたが、高速化率は最大でも約 6% でほとんど効果がなかった。これに対して提案手法の active では、 $\theta$  が 0.7 付近から処理時間を大幅に低減できた。実用的な検出精度を保てる  $\theta = 0.5$  では、他の手法と比較して約 1.2 倍に高速化できた。

### 4.3 議論

提案手法では、しきい値より大きな極小点は検出できないため、分割点を見逃す可能性があることに注意が必要である。提案手法は映像ショット分割、話者交代検出、画像エッジ検出にも適用できる。柏野らのオーバースキッピングによる高速化手法を適用すると、提案手法はさらに高速化が可能である[8]。

## 5. 結論

本稿では 2 つの隣接する探索窓をスライドさせながら信号列を分割する処理を高速化するアクティブセグメンテーション法を提案した。提案手法では探索窓間の類似度があらかじめ定めたしきい値を下回り得ない区間で検出処理を省略した。テキスト分割の実験では、提案手法は従来手法と比較して約 1.2 倍に高速化できた。

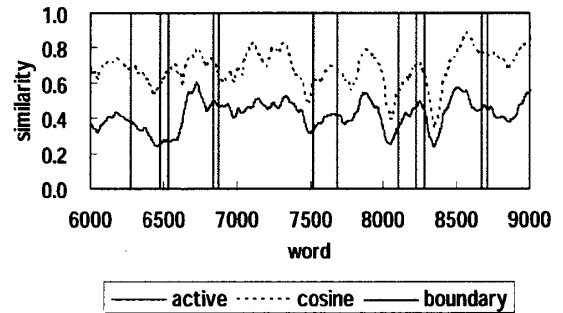


図 2 検出精度

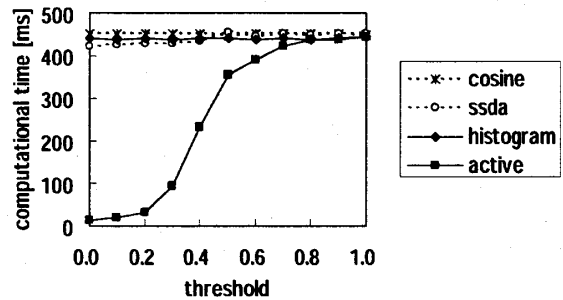


図 3 しきい値と処理時間の関係

## 参考文献

- [1] Fernando W. A. C., Canagarajah C. N., Bull D. R., "Scene change detection algorithms for content-based video indexing and retrieval," *Electronics & Communication Engineering Journal*, Vol.13, No.3, pp. 117-126, Jun. 2001.
- [2] P. Delacourt, C. J. Wellekens, "DISTBIC: A speaker-based segmentation for audio data indexing," *Speech Communication*, Vol.32, pp.111-126, 2000.
- [3] Hearst, Marti A, TextTiling: Segmenting Text into Multiparagraph Subtopic Passages, *Computational Linguistics*, Vol. 23, No 1, pp. 33-64, 1997.
- [4] Barnea, D. I. And Silverman, H. F.: "A Class of Algorithms for Fast Digital Image Registration," *IEEE Trans. Comput.*, Vol.c-21, No.2, pp.179-186(Feb.1972).
- [5] 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.
- [6] 柏野 邦夫, ガビンスミス, 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.
- [7] 松本 裕治, 北内 啓, 山下 達雄, 平野 善隆, 松田 寛, 高岡 一馬, 浅原 正幸, "日本語形態素解析システム『茶笥』 version 2.2.1 使用説明書", Dec. 2000.
- [8] 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," *日本音響学会講演論文集*, pp.445-446, Sept. 1999.