情報処理学会論文誌 Vol.54 No.2 484–494 (Feb. 2013)

音節継続時間を利用した直線検出に基づく音声検索語検出

大野 哲平[1,a)] 秋葉 友良[1,b)]

受付日 2012年5月31日, 採録日 2012年11月2日

概要：情報通信網の発展とデータ記録コストの低減により，音声を含むマルチメディアコンテンツが増大している．現在主流となっているマルチメディアデータに対する検索システムが検索の根拠としているファイル名やタグ情報等の人手によるメタデータ付与は，人的コストが非常に高い．そこで，音声データから求めたい情報になるべく早く，低コストでアクセスできる検索技術が求められている．音声検索語検出（Spoken Term Detection; STD）はある特定の検索語が音声データ中のどこで発話されたかを特定するタスクであり，現在活発な研究活動が行われている分野である．先行研究として，近似文字列照合を音節間距離平面上の直線検出問題ととらえる手法が提案されており，高速で距離順の検出が可能であることが示されている．しかし，認識誤りに対する対策に問題が残されていた．本研究では，直線検出に基づくSTD 手法に，音節継続時間情報を組み込むことにより検索性能の向上を試みた．提案手法は，音節の代わりに分析フレームを単位とした距離空間を構成することで，脱落・挿入誤りに頑健な検出を可能にする．評価実験の結果，高 Recall の領域で検索性能を改善することを確認した．

キーワード：音声検索語検出，直線検出，音節継続時間

Incorporating Syllable Duration into the Line Detection Based Spoken Term Detection

Teppei Ohno[1,a)] Tomoyosi Akiba[1,b)]

Received: May 31, 2012, Accepted: November 2, 2012

Abstract: Nowadays, multimedia contents including speech are rapidly increasing due to both the growth of the communication networks around the world and the decrease of storage cost. The current retrieval systems for such contents rely on the manually annotated metadata, which are too expensive to be obtained. Therefore, it is required the retrieval method that is not expensive but quick to access the desired information by using their speech data. Spoken term detection (STD) is one of the solution, which tries to find the positions that the given query term is uttered at in the spoken document, and recently has been actively studied in the context of speech processing. While conventional methods for STD are to apply approximate string matching against a subword sequence of spoken document obtained by speech recognition, there has been proposed a line-detection-based STD method, which regarded string matching as line detection in a syllable distance plane. While it demonstrated to enable fast and distance-ordered detections, it still suffered from the insertion and deletion errors brought by speech recognition. In this work, we try to improve the detection performance by employing the syllable duration information. The proposed method enables the robust detection by introducing the distance plane using frames as units, instead of using syllables as units. Our experimental evaluation showed that the incorporation of syllable duration improved its detection performance in high-recall regions.

Keywords: spoken term detection, line detection, syllable duration

1. はじめに

情報通信網の発展とデータ記録コストの低減により，テキストデータに加えて音声を含むマルチメディアコンテンツが増大している．現在主流となっているマルチメディ

[1] 豊橋技術科学大学
Toyohashi University of Technology, Toyohashi, Aichi 441–8580, Japan
[a)] ohnoteppei@nlp.cs.tut.ac.jp
[b)] akiba@nlp.cs.tut.ac.jp

アデータに対する検索システムでは，ファイル名や人手により付加されたメタデータを頼りに検索を行っているが，ファイル名ではデータの内容表現としてはいささか限定的であり，人手によるメタデータ付与は人的コストが非常に高い．そこで，音声データから求めたい情報になるべく早く，低コストでアクセスできる検索技術が求められている．

音声検索語検出（Spoken Term Detection; STD）はある特定の検索語が音声データ中のどこに出現するのか，その位置を特定するタスクであり，2006 年に NIST を中心としてタスクが設定され，以降様々な研究が行われている．日本においても 2008 年に情報処理学会音声言語情報処理研究会の音声ドキュメント処理ワーキンググループによって日本語話し言葉コーパス（Corpus of Spontaneous Japanese; CSJ）を対象とした STD 評価用テストコレクション [1] が構築されており，現在活発な研究活動が行われている分野である．

様々な STD 手法が提案されている中で，本研究では直線検出に基づく STD 手法 [2], [3], [4] に焦点を当てる．この手法に対して，部分距離空間上の索引を用いて高速かつ距離順に検出結果を出力する STD 手法が提案されている [2]．STD の一般的な手法では，閾値をあらかじめ設定しておき，その閾値を超える候補を検出結果として出力するのに対し，部分距離空間上の索引を用いる STD 手法では，閾値を用いずにもっともらしい順に検出を行う．この特長により，従来では不可能であった利用法が可能となる．たとえば，システム全体の処理時間に対して検索に割ける時間が制限されるような応用場面において，その時間内で見つかる候補だけをもっともらしい順（距離順）に出力するということができる．あるいは，最初の候補が出力されるまでの時間は非常に短くなることを利用して，一番もっともらしい候補だけを高速に検出する場合にも有用である．また，最適な閾値は，対象音声ドキュメントの質，音声認識の性能，アプリケーションの要求，等によって異なるため，閾値の設定は難しい問題であるが，金子らの手法では検出の際に閾値を用いる必要がないため，この問題を避けることができる．

一方，直線検出に基づく STD 手法には，音声ドキュメント中の音声認識誤りに対する対策が十分ではなく，一般的な連続 DP マッチングを用いる手法に比べて検出性能が劣るという問題点があった．そこで本研究では，直線検出に基づく STD 手法の性能改善に焦点を当てる．

音節単位の直線検出 [2] では，検索語を発話している音声ドキュメント区間中の音節が脱落誤りや挿入誤りを起こした場合，検索語と対応する音声ドキュメント区間の対応が直線にならず，検出が難しくなる．そこで，検索語を発話している区間中の音節が脱落・挿入しても，正しい音節が他の音節に置き換わり，音声ドキュメント中での検索語の発話時間自体は変わっていないという仮定に基づくと，音節継続時間を利用し処理単位を音節からより細かい単位である分析フレームに変更することで，検索語と対応する音声ドキュメント区間の対応を直線近似できると考えられる．このように，処理単位を分析フレームに変更し，音節継続時間情報を利用することで音声認識誤りに対応し検索性能の改善を目指す．ただし，本論文では直線検出による STD の枠組みの中で検索性能を改善することに焦点を置き，文献 [2] での索引付けによる高速化については扱わず今後の課題とする．

CSJ を対象とした STD の評価実験において，継続時間情報を用いない直線検出に基づく STD 手法 [2] と比較し，高 Recall（50%以上）領域での検索性能に改善を確認した．このことから，高 Recall 領域での検索性能に難のあった直線検出に基づく STD 手法においても，継続時間情報を導入することにより高 Recall 領域での検出が可能になることが確認できた．

本論文の構成は以下のとおりである．まず 2 章で，本研究の関連研究について述べる．次に 3 章で，部分距離空間上の索引を用いる STD 手法 [2] の概要と問題点について述べる．4 章では，本論文で提案する音節継続時間を利用した直線検出 STD について述べる．5 章では，提案手法の実験結果について述べる．最後に 6 章で，結論と提案手法の今後の方針について述べる．

2. 関連研究

STD に対する典型的な手法は，音声データに対して音声認識を行い，得られたテキスト表現に対して検索語の検出を確認するというものである．音声認識誤りや認識語彙外語に対処する種々の手法が提案されている．たとえば，認識語彙外語の検出のためにサブワード系列の音声認識結果に対して検索語検出を行う手法 [5], [6]，ラティスやコンフュージョンネットワークで表現された複数認識候補に対して検索語検出を行う手法 [6], [7], [8]，認識結果に対し誤りを許した近似照合を行う手法 [9], [10]，等が提案されている．また複数認識結果を用いる手法の発展として，複数の音声認識器の認識結果を組み合わせて利用することで高い検索性能が得られることが報告されている [11]．これらは，STD の性能向上を目指した研究であるといえる．

一方，STD の高速化を目指した研究として，検索対象ドキュメントの索引付け手法が提案されている．これまで，STD に対する索引付け手法としては，転置インデックス [12], [13], [14] およびサフィックスアレイ [10] が主に用いられてきた．転置インデックスは，単語等のあらかじめ決められた単位をエントリとして，そのドキュメント中での出現位置のリストを記録したハッシュ表（あるいは，同様の機能を持つ任意のデータ構造）を用いる手法である．検索時は，検索語を索引付けの単位に分割し，それぞれをエントリとしてハッシュ表を調べることで出現位置を特定する．一方，サフィックスアレイは，検索対象文書をツリー状に圧縮し，文書中の共通部分文字列をまとめて検索

可能とすることで，検索の効率化を行う．転置インデックスに比べて，あらかじめ単位を決めることなく対象ドキュメントの全部分列を対象にした索引を作ることができるのが大きな特徴である．転置インデックスやサフィックスアレイは，索引自体は一致/不一致の 2 値情報しか含んでいない．これらの手法では，誤りを含む音声ドキュメントから近似照合を行うために，あらかじめ検索語と候補間の距離に対する閾値を設定しておき，その閾値以内の候補を検出結果として出力する手法であった．最適な閾値は，対象音声ドキュメントの質，音声認識の性能，アプリケーションの要求，等によって異なるため，閾値の設定は難しい問題である．そのため，閾値の設定によっては，まったく結果が得られなかったり，大量の検出結果が 1 度に得られたりする場合がある．また，1 度検出を行った後，新たな検出結果が必要な場合は，再度閾値を設定して検出処理を最初から再実行する必要があった．

これに対して，金子ら [2], [15] は部分距離空間上の索引付けを用いた直線検出による STD 手法を提案している．この手法では音声認識結果の音節列と検索語の音節列から音節間距離平面を構成し，そのうえで直線検出を行うことで STD を実現するということを基本的なアイデアとしている．音節間距離平面の構成に関して，音節間距離情報を用いた効率的な索引付けを行うことで，従来手法のように検索の際に閾値を設定する必要がなく，距離順に結果を出力できるという特長がある．

単純な音節単位の直線検出では，音声ドキュメント中で脱落誤りや挿入誤りが発生した場合，検索語と対応する音声ドキュメント区間の対応が直線にならず，直線以外の対応も許すような連続 DP マッチングに比べ検索性能は劣る．そこで，何らかの認識誤り対策を施す必要がある．金子らは認識誤り対策として，音声ドキュメント中の隣接音素（または音節）の特徴を考慮した距離尺度を導入している．Noritake ら [3] は金子らと同様に，音節間距離平面上での直線検出による STD を提案しており，その手法中で認識誤り対策として音節間距離平面の直線強調と雑音除去を行う画像処理フィルタを用いている．しかし，これらの認識誤り対策法を用いても，認識誤りに対する対策が十分とはいえない．脱落誤りや挿入誤りを起こした音声ドキュメント区間と検索語を直線で対応付けするのには，処理単位が音節の場合認識誤り時の変動が大きく，検索性能の改善には限界があると考えられる．

音声ドキュメント処理に直線検出を利用した関連研究として，音声データ中から言語知識なしで語の発見を行う研究 [16] がある．この研究では，音声データから何度も繰り返し出現する類似区間を発見するために，音響特徴量ベクトルの自己類似度平面に対して直線検出アルゴリズムを適用している．しかし，索引付けによる高速化に用いているのではないこと，音声データ間の類似度計算に用いていること（提案手法はテキストと音声データ間の類似度）等，提案手法とは動機や目的が大きく異なる．

3. 部分距離空間上の索引付けに基づく STD

音声認識結果の音節列を x 軸，検索語の音節列を y 軸にとり，平面上の格子点には音節どうしの誤りやすさを反映する何らかの距離を与え音節間距離平面を構成する．ここで，音節間距離を画素濃度に対応付けて，音節間距離が小さいほど（音節間類似度が大きいほど）濃い画素で表示することを考える．たとえば，音声ドキュメント音節列が “wa ta shi wa shi ze N ge N go sho ri ke N” であり，検索語音節列が “shi ze N ge N go” であったとすると，**図 1** のような平面が得られる．この平面上の黒い格子点が並ぶ傾き 1 の直線は，音声認識結果中で検索語が時系列順に対応する区間を表しているので，この直線を検出することが STD を実行していることに対応する．

今，検索語の長さ（音節数）を M，検索対象音声ドキュメントの長さ（音節数）を N とし，音節間距離平面上の格子点に与える音節間距離を $D_{i,j}$（$0 \le i < M$，$0 \le j < N$）で表現する．

このとき，STD 問題，すなわち傾き 1 の直線検出問題は，以下の累積音節間距離 T_j の小さい位置 j（$0 \le j < N$）を求める問題として定式化できる．

$$\begin{aligned} T_j &= D_{0,j} + D_{1,j+1} + \cdots + D_{M-1,j+M-1} \\ &= \sum_{i=0}^{M-1} D_{i,j+i} \end{aligned} \tag{1}$$

検索対象の音声ドキュメントは検索前に既知であると仮定できるので，音節間距離 $D_{i,j}$ は音節ごとにあらかじめ計算しておくことができる．ここで，ある音節 s と検索対象音声ドキュメント音節列の位置 j の音節との音節間距離を $D(s)_j$ とする．各音節ごとに求めた音声ドキュメント長の長さを持つベクトル（音節間距離ベクトル）$[D(s)_0, D(s)_1, \ldots, D(s)_j, \ldots, D(s)_{N-1}]$ は，検索前にあらかじめ求めておくことができる．また，検索語音節列 $Query = s_0 s_1 \ldots s_{M-1}$ が与えられたとき，音節間距離ベクトルを y 軸方向に検索語の音節順に並べることで（すなわち $D_{i,j} = D(s_i)_j$ とすることで），$D_{i,j}$（$0 \le i < M$，

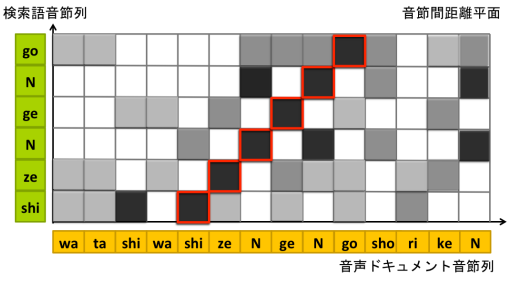

図 1 直線検出による STD
Fig. 1 STD as straight line detection.

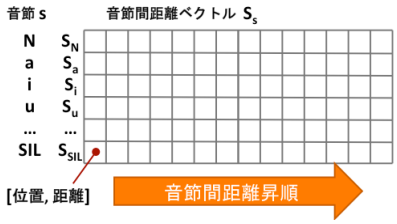

図 2 距離空間上の索引付け
Fig. 2 Metric space indexing.

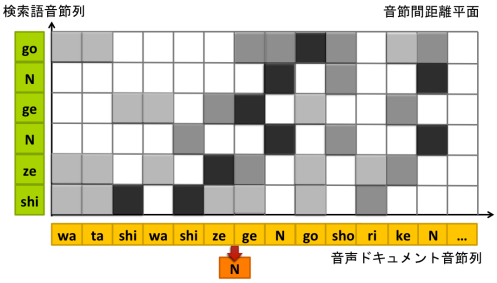

図 3 音節単位の直線検出 STD で脱落誤りが発生した場合
Fig. 3 An example of deletion error in line-detection-based STD using syllable as a processing unit.

$0 \leq j < N$）を求めることができる．

さらに，音節間距離ベクトルはあらかじめ距離順にソートしておくことができる．検索対象音声ドキュメントの音節位置ベクトル $[0, 1, \ldots, j, \ldots, N-1]$ を，音節間距離ベクトル $[D(s)_0, D(s)_1, \ldots, D(s)_j, \ldots, D(s)_{N-1}]$ に従って昇順ソートした，音声ドキュメント中での位置 j とその位置での音節間距離 $D(s)_j$ を要素として持たせたベクトルを S_s とする．S_s は，第 1 要素側（距離の小さい側）をスタックトップとするスタックとして使用する．S_s は音節 s と文書（ここでは，音声ドキュメント音節列）の各位置 j の音節との間で定義される距離に基づいた距離空間上の索引と見ることができる（**図 2**）．この S_s を検索に用いると，累積距離が小さい位置から結果を出力する，高速な STD アルゴリズムが実現できる．

3.1 問題点

音声ドキュメント音節列に誤りが含まれており，それが問題となる場合を考える．直線検出による STD で特に問題となるのは，音声ドキュメント音節列中から音節が抜け落ちてしまう脱落誤りと，誤った音節が挿入されてしまう挿入誤りである．これらの認識誤りが発生した場合，検索語音節列と音声ドキュメント音節列中の検索語発話部分が時系列順に対応しなくなるため，直線検出では検索語発話位置が検出できなくなってしまう．

たとえば，検索対象音声 “私は自然言語処理研……” から自然という単語中の “ん” が脱落し，“wa ta shi wa shi ze ge N go sho ri ke N……” と認識されてしまった場合に STD を実行すると，**図 3** のようになり直線上の累積距離が小さくならないため，発話位置の検出が行われない．

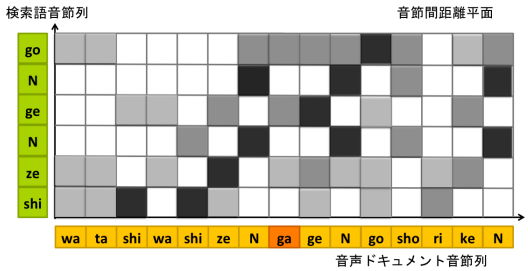

図 4 音節単位の直線検出 STD で挿入誤りが発生した場合
Fig. 4 An example of insertion error in line-detection-based STD using syllable as a processing unit.

また，検索対象音声 “私は自然言語処理研……” の “自然” と “言語” の間に “が” が挿入されてしまい，“wa ta shi wa shi ze N ga ge N go sho ri ke N……” と認識されてしまった場合に STD を実行すると，**図 4** のようになる．この場合も同様に発話位置の検出が行われない．

4. 提案手法：音節継続時間を考慮した STD

音声ドキュメントに対する検索の性能を上げるためには，認識誤り対策が不可欠であるが，音節を単位とした金子らの手法では音声認識の際の脱落誤りや挿入誤りに対応するのが難しいという問題点があった．そこで本研究では，音声ドキュメント音節列，検索語音節列両方の音節列の各音節に音節継続時間情報を組み込み，音節継続時間を持っている音節間距離平面上で直線検出を行うことによって脱落誤りや挿入誤りといった音声認識の際の誤りに対応する手法を提案する．

4.1 検索語音節列における音節継続時間長の推定

音節継続時間情報を持つ音節間距離平面を構成するためには，音声ドキュメント音節列と検索語音節列両方について音節継続時間情報が必要である．これらのうち音声ドキュメントの音節列については，音声ドキュメントを音声認識する際に，分析フレーム長を最小単位とした各音節区間の時刻情報が得られる．一方，検索語音節列はテキストで与えられるため，継続時間情報を持っていない．そこで，音声ドキュメントの音声認識結果から各音節の継続時間を推定し，その推定値を検索語各音節の継続時間情報として利用する．

ある音節の発話時間は，前後の文脈（たとえば，音素や音節）に依存すると考えられる．そこで，音声ドキュメントの音声認識結果から音素–音節–音素の並びのように，前後の音素の違いを考慮した系列（音素文脈系列と呼ぶ）を抽出し，各音素文脈系列について中央の音節のフレーム数の平均を計算し，音素文脈中のある音節の音節継続時間とした．検索語を音素文脈系列に分解し，検索対象音声ドキュメントから求めた音素文脈系列の音節継続時間を付与した

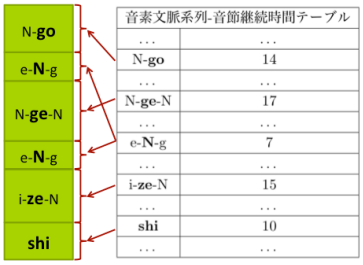

図 5 検索語音節列への音節継続時間の付与
Fig. 5 Adding syllable durations to syllables in a query term.

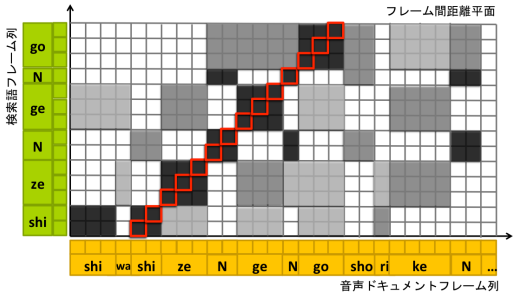

図 6 処理単位をフレームにした場合の直線検出による STD
Fig. 6 Line-detection-based STD using frame as a processing unit.

(図 5)．音声ドキュメント中に検索語中に現れる音素文脈系列が存在しない場合，音素–音節のように前の音素だけを考慮した音素文脈系列での音節継続時間の平均値を付与し，音素–音節の系列も存在しない場合は音節単体での音節継続時間の平均値を付与する．

4.2 STD の定式化

3 章と同様に，音声ドキュメント音節列を x 軸，検索語の音節列を y 軸にとる．ここで，各軸の音節列を各音節の継続時間長に従ってフレーム単位に分割し，分析フレーム長を単位とする距離平面を構成する．音節間距離を画素濃度に対応付けて，音節間距離が小さいほど濃い画素で表示することを考えると，この平面上の黒い格子点が並ぶ正の傾きを持つ直線は音声ドキュメント中で検索語が時系列順に対応する区間を表している（図 6）．したがって，STD 問題はこの直線を検出する問題として定式化できる．

まず，従来法と同様に傾き 1 の直線検出を定式化する．検索語の長さ（フレーム数）を m，検索対象音声ドキュメントの長さ（フレーム数）を n とし，音節間距離平面上の格子点に与えるフレーム間距離を $D_{i,j}$（$0 \leq i < m$, $0 \leq j < n$）で表現する．音節間距離平面上での傾き 1 の直線検出は，累積距離 T_j を検索語フレーム数 m で正規化した正規化累積距離 $\overline{T_j}$ の小さい位置 j（$0 \leq j < n$）を求める問題として定式化できる．式 (2) は式 (1) と同じ形だが，式 (1) は発話位置の検索単位が音節単位なのに対し，式 (2) はフレーム単位である点に注意されたい．

$$T_j = D_{0,j} + D_{1,j+1} + \cdots + D_{m-1,j+m-1}$$

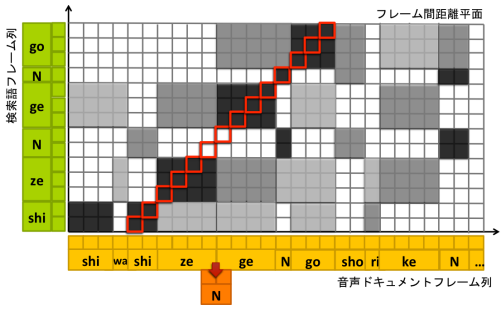

図 7 フレーム単位の直線検出 STD で脱落誤りが発生した場合
Fig. 7 An example of deletion error in line-detection-based STD using frame as a processing unit.

$$= \sum_{i=0}^{m-1} D_{i,j+i} \tag{2}$$

$$\overline{T_j} = \frac{T_j}{m} \tag{3}$$

格子点に与えるフレーム間距離 $D_{i,j}$ には，以下のいずれかを利用した．

- フレームが属する音節の間の距離をそのまま用いる場合．

$$D_{i,j} = d(a(i), b(j)) \tag{4}$$

- フレームが属する音節の間の距離を，検索語側の音節継続時間長（フレーム数）で正規化した場合．

$$D_{i,j} = d(a(i), b(j)) \cdot \frac{m}{len(a(i)) \cdot M} \tag{5}$$

ここで，$a(i)$ は検索語の位置 i のフレームが属する音節，$b(j)$ は音声ドキュメントの位置 j のフレームが属する音節，$d(a, b)$ は音節 a，b 間の距離，$len(a)$ は音節 a の継続時間長（フレーム数）である．

認識誤りが発生した場合，たとえば脱落誤りが発生した場合を考えると，脱落した音節は他の音節に置き換わっている．したがって，検索語を発話している区間中の音節が脱落しても，検索語の発話時間自体は変わっていないと考えられる．これは挿入誤りについてもいえ，検索語を発話している区間中に誤った音節が挿入されても，脱落誤りの場合と同様に検索語の発話時間は変わっていないと考えられる．この性質により，音節継続時間情報を持つ音節間距離平面で直線検出を行えば，音声ドキュメント音節列中の検索語に対応する区間に黒い格子点が多く並び，発話位置を検出することができる．

図 7 に検索対象音声ドキュメント音節列 “shi wa shi ze N ge N go sho ri ke N” から 5 番目の音節 “N” が脱落した場合の音節継続時間情報を持つ音節間距離平面での直線検出の模式図を，図 8 に 5 番目の音節 “N” と 6 番目の音節 “ge” の間に音節 “ga” が挿入されてしまった場合の音節継続時間情報を持つ音節間距離平面での直線検出の模式図を示した．どちらも誤った音節部分は検出直線上に白い（音

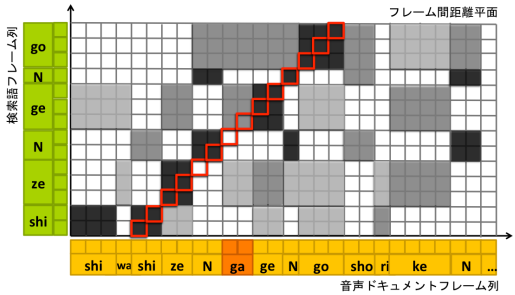

図 8 フレーム単位の直線検出 STD で挿入誤りが発生した場合
Fig. 8 An example of insertion error in line-detection-based STD using frame as a processing unit.

節間距離が大きい）格子点が現れてしまっているが，誤った音節以外の音声ドキュメント音節列と検索語音節列で一致している音節部分が直線上に並ぶことで，正規化累積距離 $\overline{T_j}$ は小さくなり発話位置を検出できると考えられる．

4.3 傾きを可変とした直線検出による推定誤差補償

検索語の音節継続時間は，音声ドキュメントの認識結果に基づいて推定するため，推定誤差が発生する．そこで，検出直線の傾きを様々に変更して，最も累積距離の小さい候補を音声ドキュメントのその点での候補とすることで推定誤差を補償する．これは式 (3) の代わりに式 (6) を用いて適当な閾値より小さい $\overline{T_j}$ を求めることで実現する．

$$\begin{aligned}\overline{T_j} &= \frac{1}{m} \cdot \min_{l \in Ratio} \left(D_{0,j} + \cdots + D_{m-1,j+\lfloor (m-1)l \rfloor}\right) \\ &= \frac{1}{m} \cdot \min_{l \in Ratio} \sum_{i=0}^{m-1} D_{i,j+\lfloor il \rfloor} \end{aligned} \tag{6}$$

たとえば，傾き 1 の直線に加えて，検索語の推定音節継続時間よりも音声ドキュメント側の発話区間が 10%短くなっている場合（傾き 1.1）と，10%長くなっている場合（傾き 0.9）を考慮し，3 つの正規化累積距離のうち一番小さい値を音声ドキュメント音節列の位置 j の候補とするならば，$l \in \{0.9, 1.0, 1.1\} = Ratio$ となる．

4.4 検索語音節境界を可変とした推定誤差補償

4.3 節の推定誤差補償に加えて，さらに検索語の各音節の境界を可変とすることで推定誤差を補償する．傾きを可変とした直線検出では，検索語全体の音節継続時間の推定誤差を補償している．しかし，それに加えてさらに検索語の各音節ごとの継続時間にも推定誤差が生じている可能性がある．この誤差を補償するため，検索語の音節境界付近のフレームは境界前後の音節両方に属しているものとし，両者のうちフレーム間距離が小さくなる音節を採用してそのフレーム間距離とする．この処理は，フレーム間距離の定義として式 (4) の代わりに以下の式 (7) か式 (8)，式 (5) の代わりに式 (9) か式 (10) を用いることによって実現する．

- フレームが属する音節の間の距離をそのまま用いる場合．

$$D_{i,j} = \min_{-A \le k \le A} d(a(i+k), b(j)) \tag{7}$$

$$D_{i,j} = \min_{-\alpha len(a(i)) \le k \le \alpha len(a(i))} d(a(i+k), b(j)) \tag{8}$$

- フレームが属する音節の間の距離を，検索語側の音節長（フレーム数）で正規化した場合．

$$D_{i,j} = \min_{-A \le k \le A} d(a(i+k), b(j)) \cdot \frac{m}{len(a(i)) \cdot M} \tag{9}$$

$$D_{i,j} = \min_{-\alpha len(a(i)) \le k \le \alpha len(a(i))} d(a(i+k), b(j)) \cdot \frac{m}{len(a(i)) \cdot M} \tag{10}$$

ここで，A と α は境界付近で何フレームにわたり音節境界を可変とするかを決定する定数であり，可変とする幅を A フレームで固定する場合は式 (7) か式 (9)，可変とする幅を各音節の継続時間の α%とする場合は式 (8) か式 (10) を用いる．

4.5 累積距離に対する音節数ペナルティ

音節間距離平面上で直線検出を行った結果，音声ドキュメント音節列の各位置 j での正規化累積距離 $\overline{T_j}$ が得られる．この $\overline{T_j}$ が適当に設定した閾値よりも小さければ，j を音声ドキュメント中の検索語の発話位置として検出する．直線検出で STD を実現する場合，検索語の音節数によって最適な閾値が変化するということが Noritake ら [3] によって報告されている．このことは，音節数が少ない検索語は湧き出し誤りが多く発生するため検出と判定する正規化累積距離に対する基準を厳しめに，音節数が多い検索語は直線検出では直線近似による誤差が増えるため正規化累積距離に対する基準を緩めにするのが妥当であるという考えに基づく．そこで，式 (11) により検索語の音節数に基づき正規化累積距離 $\overline{T_j}$ に重み付けを行った．式 (11) では，基準の検索語長（音節数）より短い検索語は正規化累積距離が大きくなるように，長い検索語は正規化累積距離が小さくなるような重み付けが行われる．ここで，$\overline{T_j}$ は正規化累積距離，M は検索語長（音節数），$\overline{M}$ は基準の検索語長（音節数），β は重みの変化をなだらかにするための定数である．

$$\tilde{T}_j = \frac{\overline{M} + \beta}{M + \beta} \cdot \overline{T_j} \tag{11}$$

5. 評価実験

5.1 テストコレクション

評価実験には，CSJ を対象とした検索語検出テストコレクション [1] の，コア講演（177 講演，計 44 時間）用既知語検索セット（50 語）を用いた．このテストコレクションでは，音声ドキュメントは Inter-Pausal Unit（IPU）とい

うポーズで数秒間ごとに区切った単位に分割されており，検索語が発話されている IPU を正解発話とする．各実験を通して同じテストコレクションを使用しており，検索対象の講演音声，検索語セットは共通である．

5.2 評価尺度

検索性能の評価尺度には NTCIR-9 Core Task “Spoken Doc” で使用されている Recall-Precision 曲線，および最大 F 値を用いる．最大 F 値は，以下のように計算された F 値について，閾値を動かしたときの最大値である．

$$\text{Recall} = \frac{\text{検出した正解発話数}}{\text{正解発話数}} \tag{12}$$

$$\text{Precision} = \frac{\text{検出した正解発話数}}{\text{検出した発話数}} \tag{13}$$

$$\text{F 値} = \frac{2 \cdot \text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}} \tag{14}$$

5.3 ベースライン手法

ベースライン手法には，文献 [2] の音節単位での直線検出による STD の最適解アルゴリズムを用いた．この最適解アルゴリズムは，式 (15) で表現される音節間距離平面上での傾き 1 の直線上の累積距離を計算し，距離の小さい箇所から順に発話位置候補を出力するアルゴリズムである．ここで，$d(a,b)$ は音節 a と b 間の距離，$D_{i,j}$ は傾き 1 の直線上の累積距離である．M は検索語長（音節数），N は音声ドキュメント長（音節数）である．

$$\begin{cases} D_{0,j} = d(a(0), b(j)) & (i = 0,\ 0 \le j < N) \\ D_{i,0} = D_{i-1,0} + d(a(i), b(0)) & (0 < i < M,\ j = 0) \\ D_{i,j} = D_{i-1,j-1} + d(a(i), b(j)) & (0 < i < M,\ 0 < j < N) \end{cases} \tag{15}$$

5.4 音節間距離

フレーム間距離を決定するための音節間距離 $d(a,b)$（a，b は音節）には，文献 [17] を参考に音響モデル間の Bhattacharyya 距離を使用した．

5.5 フレーム間距離の正規化の有無

4.2 節で提案した 2 種類のフレーム間距離を用いて STD を実行し，フレーム間距離として音節間距離をそのまま使用する場合（式 (4)）と検索語の各音節の累積距離に対しての重みが一定になるように正規化する場合（式 (5)）とを比較した．検出直線の傾きは 1 で固定した．

F 値の最大値を比べると（**表 1**），正規化を行った場合の方が行わなかった場合よりも小さくなった．しかし，実験結果のグラフ（**図 9**）を見ると，正規化を行った場合が必ずしも行わなかった場合よりも検索性能が悪くなるわけではないことが分かる．正規化を行った場合は行わなかった場合に比べて，高 Recall 領域では Precision が下回っているが，低 Recall 領域では上回っている．

これは，正規化を行わなかった場合，音節継続時間が短い音節は軽視され，長い音節を重要視して発話位置の検出が行われる．低 Recall 領域，すなわち音声認識率が比較的高いと考えられる音節列のみに対しての照合時にはどの音節も一律の重みで評価した方が検索性能が良くなると考えられる．しかし，高 Recall 領域，すなわち音声認識率が比較的低いと考えられる音節列も含む場合での照合時には音節継続時間が短い音節は脱落誤りにより脱落していたり，挿入誤りが存在したりすると考えられるため，音節継続時間が長い音節を信頼して検索に用いる方が検索性能が良くなると考えられる．

表 1 フレーム間距離の正規化の有無と検索性能
Table 1 Detection performances with and without normalization of distance between frames.

傾き幅	Ratio	フレーム間距離	最大 F 値
1	{1}	式 (4)（正規化なし）	0.370
1	{1}	式 (5)（正規化あり）	0.365

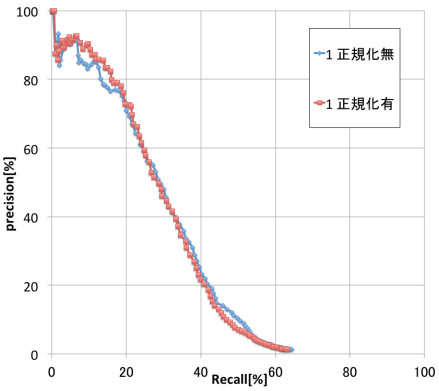

図 9 Recall-Precision 曲線（フレーム間距離の正規化）
Fig. 9 Recall-Precision curves with and without normalization of distance between frames.

表 2 傾きを可変とした推定誤差補償結果
Table 2 Effect of the compensation for estimation errors by varying slope angle.

傾き幅	*Ratio*	最大 F 値	誤差補償率
1	{1}	0.365	-
1 ± 10%	{0.9, 0.95, 1, 1.05, 1.1}	0.441	33.9%
1 ± 20%	{0.8, 0.85, . . . , 1.15, 1.2}	0.476	57.6%
1 ± 30%	{0.7, 0.75, . . . , 1.25, 1.3}	0.481	73.2%
1 ± 40%	{0.6, 0.65, . . . , 1.35, 1.4}	0.481	83.8%

5.6 傾きを可変とした直線検出による推定誤差補償の効果

検出直線の角度を変化させ，単純な傾き 1 の直線検出から，傾きを ±40%変化させた直線検出までの 5 種類の傾き幅について検索性能を調べた．**表 2** は式 (6) 中の傾きの値集合 *Ratio* を示している．格子点に与える距離 $D_{i,j}$ には，

表 3 音節境界を可変とした推定誤差補償結果
Table 3 Effect of the compensation for estimation errors by varying boundaries between neighboring syllables.

傾き幅	音節境界可変	フレーム間距離	最大 F 値
1	無	式 (5)	0.365
	有	式 (9)	0.404
1 ± 30%	無	式 (5)	0.481
	有	式 (9)	0.503
	有	式 (10)	0.521

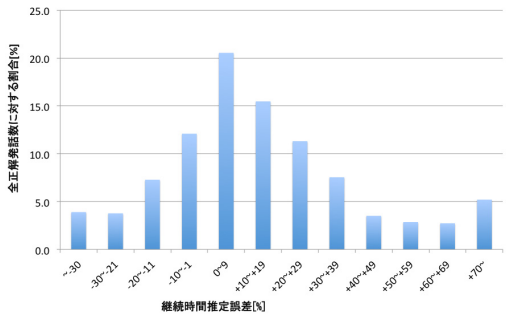

図 10 検索語音節継続時間推定誤差
Fig. 10 Distribution of estimation errors of syllable duration in queries.

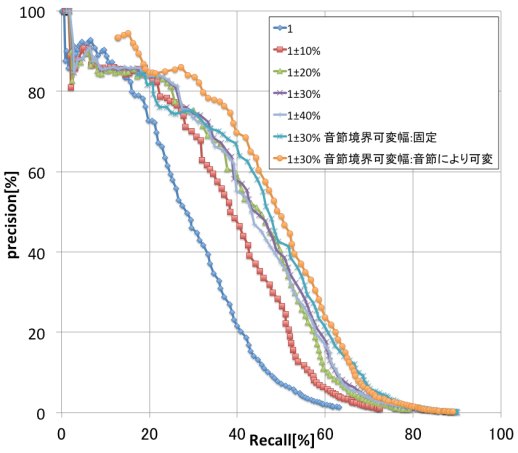

図 11 Recall-Precision 曲線（傾きを可変とした直線検出）
Fig. 11 Recall-Precision curves by varying slope angle.

検索語側の音節長で正規化した音節間距離（式 (5)）を用いた．結果のグラフを図 **11** に示す．また，傾きを 1 で固定した場合と，±30%変化させた場合の比較を特に**表 3** に示した．

単純な傾き 1 の場合に比べ，検出直線の傾きを可変とした場合の方が検索性能が向上していることから，この処理により検索語音節継続時間の推定誤差を補償することができているといえる．検出直線の角度変更の範囲が広がるほど検索時の計算量は増えるため，計算量と検索性能はトレードオフの関係になる．しかし，傾きを 20%以上変化させても検索性能に大きな違いが見られない．

推定がどの程度正確に行われているかを確認するため，音声認識結果から正解発話区間の継続時間を調査した．今回使用したテストセットには正解発話が 769 発話含まれている．そのうち酷い音声認識誤り（“国立国語研究所” が “え” と認識されている等）を起こしている発話を除く 740 発話の継続時間を調べ，推定した検索語音節継続時間と本来の検索語音節継続時間にどの程度誤差が発生しているかを図 **10** に示した．提案手法での推定では，検索語音節継続時間は実際より長く推定される傾向にある．

また，傾きを可変とした直線検出（4.3 節）によってどの程度検索語音節継続時間の推定誤差を補償できているかを表 2 の誤差補償率の列に示した．誤差補償率とは，ある傾き変更幅内で検出直線の角度を変更したとき，推定音節継続時間が正解音節継続時間と一致する割合である．実際には，検出直線の傾き幅を広げるにつれ湧き出し誤りも増加する．また，推定に大きな誤差のある発話は認識誤り率が高いことが予想されるため（脱落誤りや挿入誤りが多く発生している場合，検索語全体の継続時間は推定から大きくはずれる），検索性能が飽和すればそれ以上に検出直線の傾き幅を広げる必要はない．

5.7 検索語音節境界を可変とした推定誤差補償の効果

4.3 節の推定誤差補償に加えて，さらに検索語の各音節境界を可変とすることで推定誤差を補償した場合について検索性能を調べた．境界付近で何フレームにわたり音節境界を可変とするかを決定する定数 A の値は 3，α の値は 0.2 とした．

傾き幅を 1 に固定，つまり検出直線の傾き変更による推定誤差補償は行わない場合でも，検索語音節境界を可変とした推定誤差補償のみによって検索性能は改善された．傾き幅を変更した場合，つまり検出直線の傾き変更による推定誤差補償と検索語音節境界を可変とした推定誤差補償の両方を行った場合も検索性能は改善された．検索語音節境界を可変とした推定誤差補償には，各音節の継続時間を一定の比率で伸縮させるだけである傾き可変の推定誤差補償にはない，音節ごとに継続時間の変動を吸収する働きがあり，この効果により検索性能が向上したと考えられる．

5.8 累積距離に対する音節数ペナルティの効果

式 (11) により，正規化フレーム間距離に対して検索語の音節数に基づき重み付けを行った場合の検索性能について調べた．検出直線の傾きを変化させる範囲は 1 ± 30%，フレーム間距離の正規化は行う（式 (9) および式 (10) を使用する）という条件のもとで，基準の音節数 $\overline{M}$ は 5 とし，音節数ペナルティの変化をなだらかにするための定数 β を変化させて検出を行った．

実験結果のグラフ（図 **12**，図中の大きなラベルが最大 F 値を示す点）を見ると，音節数ペナルティを行った場合が行わなかった場合に比べ全面的に向上しており，音節数ペ

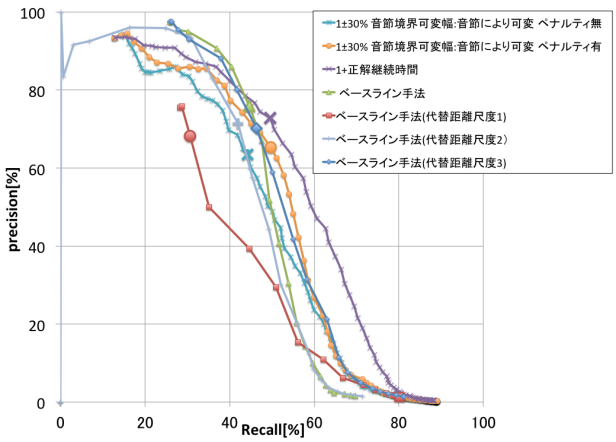

図 12 Recall-Precision 曲線（音節数ペナルティと正解継続時間）
Fig. 12 Recall-Precision curves by introducing penalty on number of syllables and by supplying oracle durations.

表 4 音節数ペナルティと正解継続時間を与えた場合の結果
Table 4 Detection Performances by introducing penalty on number of syllables and by supplying oracle durations.

手法	傾き幅	フレーム間距離	音節数ペナルティ	最大F値
提案手法	1 ± 30%	式 (10)	無	0.521
			有 ($\beta = 0$)	**0.564**
	1+正解継続時間	式 (9)	有 ($\beta = 2$)	0.590
ベースライン手法	-	-	-	0.562
（代替距離尺度 1）	-	-	-	0.422
（代替距離尺度 2）	-	-	-	0.528
（代替距離尺度 3）	-	-	-	0.558

ナルティが非常に効果的であることが分かる．音節数ペナルティを行った最大の最大 F 値を示した**表 4** の 1 ± 30%のペナルティありの場合とペナルティなしの場合を比較すると，検索性能は最大 F 値で 8.3%向上した．ベースライン手法の最大 F 値に比べてもわずかだが（0.4%）向上した，特に，比較的高 Recall 領域（50%以上の領域）での検索性能に関して大きな改善が見られた．

次に，音節継続時間の推定が正確に行われるという仮定のもとでの検索性能を調べた．傾き幅は 1 で固定し，検出直線の傾きを可変とする推定誤差補償，検索語音節境界を可変とした推定誤差補償の両方を行う．さらに音声ドキュメント中の正解発話が存在する区間（正解 IPU）では，5.6 節で調査した正解継続時間に基づいて，各 IPU 内の検索語発話時間と推定音節継続時間が一致するように検出直線の傾きを調節する処理を行った．結果は図 12 と表 4 の “1+正解継続時間” に示した．最大 F 値ではベースライン手法に対して 5.0%向上し，特に高 Recall 領域での検索性能が大きく改善された．高 Recall 領域では認識誤りが多く発生していることが予想され，その領域で検索性能が改善されたことは，提案手法がベースライン手法に比べて認識誤りに頑健であることを示す．

また，ベースライン手法の拡張として，隣接する音節を考慮した代替距離尺度を導入することで認識誤りに対処することが提案されている．以下に文献 [2] で提案されている代替距離尺度 $d_{extend1}$，$d_{extend2}$ と，文献 [15] で提案されている代替距離尺度 $d_{extend3}$ を示す．これらの代替距離尺度は式 (15) の音節間距離 $d(a, b)$ の代わりに用いる．式 (17) 中の ρ，σ と式 (18) 中の γ は隣接する音節の影響を調節する近傍ペナルティである．近傍ペナルティは各文献を参考に $\sigma/\rho = 0.1$，$\gamma = 0.5$ とした．

図 12 で比較すると，文献 [2] で提案されている代替距離尺度 1，2 を用いた拡張はベースライン手法よりも全体的に検索性能が低下しており，音節数ペナルティなしの場合の提案手法の方が高 Recall 領域での性能が高い．新しく文献 [15] で提案された代替距離尺度 3 を用いると，ベースラインに比べて最大 F 値は低下しているが，高 Recall 領域での検索性能は向上する．代替距離尺度 3 と比較すると，提案手法は音節数ペナルティを与えることで，Recall が 50%から 60%の範囲で性能向上しており，最大 F 値も上回った．

- 代替距離尺度 1

$$d_{extend1}(a(i), b(j)) = \min\{d(a(i), b(j-1)), d(a(i), b(j)), d(a(i), b(j+1))\} \quad (16)$$

- 代替距離尺度 2

$$d_{extend2}(a(i), b(j)) = \frac{\rho d(a(i), b(j-1)) + \sigma d(a(i), b(j)) + \rho d(a(i), b(j+1))}{\sigma + 2\rho} \quad (17)$$

- 代替距離尺度 3

$$d_{extend3}(a(i), b(j)) = \min\{d(a(i), b(j-1)) + \gamma, d(a(i), b(j)), d(a(i), b(j+1)) + \gamma\} \quad (18)$$

5.9 提案手法による改善の例

提案手法により得られた改善について述べる．提案手法とベースライン手法の比較として，検出に用いる閾値を一致させたとき，ベースライン手法で検出できなかった発話が提案手法によって検出できた場合，逆にベースライン手法で検出できた発話が提案手法によって検出できなくなった場合について調べた．比較対象とする提案手法は表 4 の傾き幅が 1 ± 30%，フレーム間距離が式 (10)，音節数ペナルティなしの結果を用いた．一致させる閾値はベースライン手法が最大 F 値（0.562）を示したときの 0.3 に設定した．

提案手法が検出できた正解発話数は 354，ベースライン

手法は 328 であった．このうち，共通の発話は 298，ベースライン手法で検出できなかった発話が提案手法によって検出できた数は 56，またベースライン手法で検出できた発話が提案手法によって検出できなくなった数は 30 であった．

提案手法により新たに検出できた例をあげる．クエリ “chi ka te tsu”（地下鉄）に対して，“chi ka q te q tsu” と促音の挿入誤りが発生している正解発話箇所を検索する場合，ベースライン手法では検出できていなかったが，提案手法によって検出できている．同様に，クエリ “ze q ta i o N ka N”（絶対音感）に対して，“ze q ta i o ka i mo” と音節 “N” の脱落誤りや音節 “i” の挿入誤りが発生している正解発話箇所を検索する場合も提案手法によって検出できている．これらの例から，提案手法によって挿入誤りや脱落誤りに対処できていることが分かる．ここではベースライン手法が最大 F 値を示す閾値に設定して比較しているが，より高 Recall 領域で比較すれば，さらにベースライン手法で検出できなかった発話を提案手法によって検出できるということが考えられる．

逆に，ベースライン手法では検出できていた箇所が提案手法により検出できなくなってしまった例をあげる．クエリ “e be re su to ka i do”（エベレスト街道）に対して，“e bi ri su to ka i do” と 2 カ所の置換誤りが発生している正解発話箇所を検索する場合，ベースライン手法では検出できていたが，提案手法では検出できていない．また，クエリ “wa N pa su to ra i gu ra mu”（ワンパストライグラム）に対して，認識誤りが発生していない正解発話箇所でも提案手法により検出できていない場合があった．提案手法では，特に音節数が多い検索語の場合に，検索語中に音節継続時間推定に大きな誤差が発生したとき，正解発話箇所に認識誤りがない，もしくは少ない場合でも検出が難しくなることが問題といえる．

5.10 計算量の増加

提案手法はベースライン手法である音節単位の直線検出 STD に比べて，(1) 処理単位を音節からフレームに変更，(2) 傾きを可変とした直線検出による推定誤差補償（4.3 節）の処理の影響により計算量が増加している．提案手法に対してナイーブに文献 [2] の索引付けを適用する場合，ベースライン手法に比べそれぞれの処理での計算量は，

(1) テストセットの検索対象音声ドキュメントでは，1 音節は平均 12 フレームで構成されているため，索引から要素を取り出す処理回数が 12 倍増加．
(2) 傾き幅 $1 \pm 30\%$では 13 段階の傾きに対応する各投票箱に投票を行うため，投票回数が 13 倍増加．

となり，両者の積を考えるとおおよそ 150 倍の計算量の増加が見込まれる．しかし，(1) は処理を行うフレームを数フレームごとに間引いて実行する，または，フレームの粒度を粗くする，(2) は傾き幅の可変段階数を減らす，等の処理により計算量を削減できると考えられる．これらの最適な処理単位の粒度の決定は今後検討すべき課題である．

また，提案手法をフレーム単位にナイーブに実装するのではなく，手法独自の性質を利用する高速化も考えられる．提案手法で利用する距離平面（図 6）を見ると，音節間距離は音節間の長方形領域内で同一の値が決まるため，音節間距離への投票は音節間の長方形領域に対して行えばよい．このような手法の規則性を利用することで，高速な索引付け手法が構築できる可能性があり，今後の課題である．

6. 結論

部分距離空間上の索引付けに基づく STD 手法の直線検出部分について，処理単位を音節からより細かい単位である分析フレームに変更し音節継続時間情報を組み込むことで検索性能を向上させることを試みた．

検索対象音声ドキュメント音節列と検索語音節列両方に音節継続時間情報を持たせ，音節間距離平面上で直線検出を行った．直線検出の際に，(1) 検索語音節継続時間に基づいた音節間距離への重み付け，(2) 検出直線の角度を可変とするとともに検索語音節境界を可変とすることで音節継続時間推定誤差を補償，(3) 正規化累積距離の音節数によるペナルティの各処理を行った．検索性能はベースライン手法の最大 F 値を上回り，特に高 Recall 領域ではより多くの正解発話位置を検出できていることが特長であり，これは特に検索単位を音節からフレームに変更し (2) の処理を行ったことによる効果だと考えられる．また，正解継続時間を与えた場合はベースライン手法に比べて最大 F 値で 5.0%の改善を達成した．あくまで検出直線の正しい傾きを人手で指定した場合の検索性能であるが，検索語音節継続時間の推定精度，推定誤差補償が改善されれば検索性能が向上する余地が十分にあることを示している．たとえば，(2) の推定誤差補償の処理について，音節境界の可変幅を決定するパラメータ α は今回 0.2 で固定していたが，これを可変として適切な値に設定するというような工夫が考えられる．

提案手法は高 Recall 領域で使用されるとき，応用先としては聞き逃しが許されないような場面での利用が考えられる．文献 [18] ではコールセンタでの会話音声に対する検索があげられている．近年のコンプライアンス意識の高まりにより，「言った・言わない」のトラブル回避のため，コールセンタでの全会話音声を保存しておくケースが増えている．「言った・言わない」のトラブル回避のために会話音声を検索する場合，漏れのない検出が重要であるため，提案手法の高 Recall 領域での性能改善の利点が生かされるものと考えられる．

今回は直線検出による STD の枠組みの中での検索性能の改善に焦点を当てたため，距離順の索引付けによる高速化は行わなかった．今後は音声ドキュメントに対する索引付けを行い検索の高速化を図るとともに，検索語音節継続時

間の推定精度，推定誤差補償についても改善を行い検索性能を向上させたい．

参考文献

[1] Itoh, Y., Nishizaki, H., Hu, X., Nanjo, H., Akiba, T., Kawahara, T., Nakagawa, S., Matsui, T., Yamashita, Y. and Aikawa, K.: Constructing Japanese Test Collections for Spoken Term Detection, *Proc. International Conference on Speech Communication and Technology*, pp.667–680 (2010).

[2] 金子泰輔，秋葉友良：部分距離空間上の索引付けに基づく音声中の高速検索語検出手法，電子情報通信学会論文誌，Vol.J95-D, No.3, pp.608–617 (2012).

[3] Noritake, K., Nanjo, H. and Yoshimi, T.: Image Processing Filters for Line Detection-based Spoken Term Detection, *Proc. International Conference on Speech Communication and Technology*, pp.2125–2128 (2011).

[4] 西　宏之，横林優貴，トランリハイエン，木村義政，柿木稔男：距離マトリクス画像中からの直線検出によるワードスポッティング（LD-DMI 法）を用いた電話会話ログ検索，Vol.2011-SLP-89, No.6 (2011).

[5] 岩田耕平，伊藤慶明，小嶋和徳，石亀昌明，田中和世，李時旭：語彙フリー音声文書検索手法における新しいサブワードモデルとサブワード音響距離の有効性の検証，情報処理学会論文誌，Vol.48, No.5, pp.1990–2000 (2007).

[6] Pan, Y., Chang, H., Chen, B. and Lee, L.: Subword-based Position Specific Posterior Lattices (S-PSPL) for Indexing Speech Information, *Proc. International Conference on Speech Communication and Technology*, pp.318–321 (2007).

[7] Saraclar, M. and Sproat, R.: Lattice-Based Search for Spoken Utterance Retrieval, *Proc. Human Language Technology Conference* (2004).

[8] Hori, T., Hetherington, L., Hazen, T.J. and Glass, J.R.: Open-Vocabulary Spoken Utterance Retrieval using Confusion Networks, *Proc. International Conference on Acoustic, Speech, and Signal Processing*, pp.73–76 (2007).

[9] Jansen, A., Church, K. and Hermansky, H.: Towards Spoken Term Discovery At ScaleWith Zero Resources, *Proc. International Conference on Speech Communication and Technology*, pp.1676–1679 (2010).

[10] Katsurada, K., Teshima, S. and Nitta, T.: Fast Keyword Detection Using Suffix Array, *Proc. International Conference on Speech Communication and Technology* (2009).

[11] Nishizaki, H., Furuya, H., Natori, S. and Sekiguchi, Y.: Spoken Term Detection Using Multiple Speech Recognizers' Outputs at NTCIR-9 SpokenDoc STD subtask, *Proc. 9th NTCIR Workshop Meeting*, pp.236–241 (2011).

[12] Iwami, K. and Nakagawa, S.: High speed spoken term detection by combination of n-gram array of a syllable lattice and LVCSR result for NTCIR-SpokenDoc, *Proc. 9th NTCIR Workshop Meeting* (2011).

[13] Chelba, C. and Acero, A.: Position Specific Posterior Lattices for Indexing Speech, *Proc. Annual Meeting of the Association for Computational Linguistics*, pp.443–450 (2005).

[14] Zhou, Z.-Y., Yu, P., Chelba, C. and Seide, F.: Towards Spoken-Document Retrieval for the Internet: Lattice Indexing For Large-Scale Web-Search Architectures, *Proc. Human Language Technology Conference*, pp.415–422 (2006).

[15] 金子泰輔，秋葉友良：部分距離空間上の索引を用いたSTD における距離順計算の厳密化と非直線検出への拡張，第 6 回音声ドキュメント処理ワークショップ講演論文集 (2012).

[16] Wang, D., King, S., Frankel, J. and Bell, P.: Stochastic pronunciation modelling and soft match for out-of-vocabulary spoken term detection, *Proc. International Conference on Acoustics Speech and Signal Processing*, pp.5294–5297 (2010).

[17] 山本一公，中川聖一：発話スタイルによる話速・音韻間距離・ゆう度の違いと音声認識性能の関係，電子情報通信学会論文誌，Vol.J83-D-II, No.11, pp.2438–2447 (2000).

[18] 大淵康成，神田直之：音声検索実用化の現状と課題，情報処理学会研究報告，Vol.2010-SLP-88, No.5 (2011).

大野 哲平

平成元年生まれ．平成 22 年沼津高専電気電子工学科卒業．平成 24 年豊橋技術科学大学工学部情報工学課程卒業．同年豊橋技術科学大学大学院工学研究科情報・知能工学専攻入学，現在在学中．

秋葉 友良

昭和 40 年生まれ．平成 7 年東京工業大学大学院システム科学専攻博士課程修了．同年通産省電子技術総合研究所入所．平成 13 年独立行政法人産業技術総合研究所に組織移行．平成 16 年より豊橋技術科学大学工学部助教授．現在，豊橋技術科学大学工学部准教授．自然言語処理，音声言語処理の研究に従事．博士（工学）．電子情報通信学会，人工知能学会，日本音響学会，言語処理学会各会員．