2007－SLP－68 (3)
2007／10／19

音声入力・認識機能を有するWebシステムw3voiceの開発と運用

西村 竜一† 三宅 純平‡ 河原 英紀† 入野 俊夫†

† 和歌山大学 システム工学部
‡ 奈良先端科学技術大学院大学 情報科学研究科

あらまし 提案するw3voiceシステムは，Webシステムに対して，音声による入力インタフェースを拡張する．JavaアプレットとCGIプログラムから構成し，通信プロトコルには，HTTP POST methodとRedirection responseを応用した実装を行った．このため，事前に特別な専用プログラムのインストールを要求せず，普段のWebブラウザをそのままで使うことができる．また，音声認識，対話，ボイスチェンジャ，掲示板等の音声Webアプリケーションを作成し，Webサイトで公開した．本研究は，家庭や職場等での音声インタフェースの利用環境を調べることを目的とする．そのために，利用者からの入力発話を蓄積し，分析をはじめている．約7ヶ月で一日47.6個，合計で8,412の入力を得ることができた．本稿では，提案システムの概要を述べ，収集データの発話時間及びSNRに関する調査結果を報告する．

w3voice: Development of Speech Input Method for Voice-enabled Web Applications

Ryuichi NISIMURA† Jumpei MIYAKE‡ Hideki KAWAHARA† , and Toshio IRINO†

† Faculty of Systems Engineering, Wakayama University, Japan
‡ Graduate School of Information Science, Nara Institute of Science and Technology, Japan

Abstract We have developed a speech input method called "w3voice" to build practical and handy voice-enabled Web applications. It is constructed using a simple Java applet and CGI programs comprising free software. The mechanism of voice-based interaction is developed on the basis of raw audio signal transmissions via the POST method and the redirection response of HTTP. We have released a number of w3voice applications on our website for public uses. The system also aims at organizing a voice database obtained from home and office environments. We have succeeded in acquiring 8,412 inputs (47.9 inputs / day) over a period of seven months. This report describes an overview of the proposed system, and results of analyzing collected inputs to observe utterance lengths and SNR.

1 はじめに

本稿では，一般的なWebブラウザ上で動作するWebアプリケーションに，音声入力の機能を付加する新たな枠組みであるw3voiceシステムと、その活用を提案する．

提案システムは，音声認識・対話や音声合成等の音情報処理技術をWebのインタフェースに適用し，音声Webアプリケーションを構築することを可能にする．また，w3voiceシステムの利用に際しては，クライアントPCに対して特別なプラグインプログラムのインストールを要求しない．このため，私たちが普段から使用しているWebブラウザをそのまま使うことができる．

これまでに提案システムを用いて，音声認識，対話，ボイスチェンジャ，掲示板等の音声Webアプリケーションの実装を行った．本プロジェクトのWebサイト（`http://w3voice.jp/`）では，それらアプリケーションを一般に公開しており，だれでも利用可能である．この公開試験の結果，本システムの動作は安定しており，実用性を備えたアーキテクチャを持つことを確認している．また，Webデベロッパがw3voiceアプリケーションを開発することができるようにツールキットの配布を行っている．

音声Webアプリケーションのフレームワークを提供すること，それと同時に，利用者の発話を収集し，家庭や職場等での音声入力環境を調査することが本研究のもう一つの目的である．

実用的な音声認識・対話の技術開発を進めるに

は，人と機械との間に生じるインタラクションの観察と分析が不可欠である．そのためのデータ収集は，これまで，対話システムのフィールドテストという形で実施されてきた [1, 2]．例えば，著者らの開発による公共型音声情報案内システム「たけまるくん」を用いた試みでは，コミュニティセンターでの据え置き型対話システムの長期間フィールドテストを実施しており，収録された発話に基づくシステムの改良が続けられている [3, 4]．しかし，公共サービス向けにデザインされた据え置きシステムでは音声入力環境（使用するマイクやオーディオデバイスの選定，設定等）をある程度コントールできるが，家庭等ではユーザは自由にマイク等を選ぶことができ，制御された環境で録音できるという前提は守られない．また，プライベートな空間での利用では，対話システムに対する接し方が異なる可能性がある．これまでは，家庭や職場での音声インタフェースの利用実態の調査が十分であるとは言えず，今後の音声インタフェースの普及に向け，ユーザの利用環境や状況を確認することが不可欠になってきている．それに対する試みとして，原らは，PC 上で動作する楽曲検索の音声対話アプリケーションをインターネット上で公開し，データの収集を行っている [5]．しかし，独自のソフトウェアを利用しているため，そのセットアップに失敗し，正しくサービスを利用できないユーザが発生していることが報告されている．一方で，提案システムは，先に述べたようにインストール等の準備を必要としない．そのため，家庭での音声インタフェースの利用実態を調査するのに適していると考える．

本稿では，Web アプリケーション化された音声対話エージェントシステムを例に，提案システムである w3voice システムの概要について述べる．さらに，公開アプリケーションで集めた発話を分析し，提案システムの実用性について議論する．

2 w3voice システムの概要

w3voice と名付けた提案システムを用いて，音声対話型の Web アプリケーションを試作した．動作画面を図 1 に示す．この例では，Web ベースの果物通信販売サイトを想定している．例えば，「ミカンを 10 個ください.」「バナナとリンゴをお願いします.」等の発話による注文を認識し，発注処理する．

Web ブラウザ上には，一般的な HTML ドキュメントと Flash を用いたアニメーションムービ，そして，音声入力パネル（Recording Panel）が表示される．HTML ドキュメントは，通常の Web サイトと同様に，イメージファイルやハイパーリンク等が埋め込まれたものである．また，Flash ムービに関しては，キャラクタエージェントの表示に利用する．本システムでは，対話処理の応答となるエージェントの音声の再生にも，合成音声

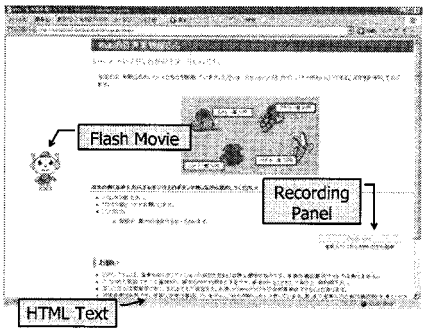

図 1: 音声対話 Web アプリケーションの動作画面（果物の通信販売用 Web サイトを想定）

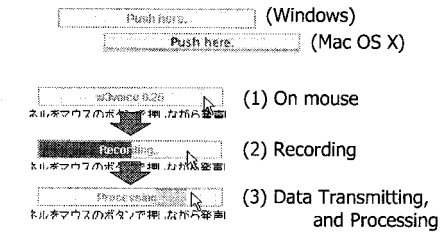

図 2: 音声入力パネル

が埋め込まれた Flash ムービを用いている．

図 2 に音声入力パネルの拡大写真を示す．この音声入力パネルは，提案システムを構成する重要なコンポーネントであり，発話の録音と，収録データを Web サーバに送信する役割を担っている．Java アプレットとして実装した．オブジェクト指向言語の Java[1]は，動作するプラットホームに依存しないことを特徴としている．すなわち，本システムは OS に関係なく，Java が動作する環境での動作が可能である．これまでに Windows（XP 及び Vista），MacOS X，Linux といった主要な OS での動作を確認した．また，Web ブラウザは，Internet Explorer（IE），Mozilla，Firefox，Safari 等で動作する．これにより，利用者は，普段利用している PC 環境そのままで，本システムを利用できる．

操作の手順は以下のようになる（図 2 を参照）．(1) マウスカーソルを音声入力パネルに移動する，(2) マウスボタンを押している間，発話を録音する，(3) マウスボタンを離すと録音を終了し，データを Web サーバに送信，処理待ちの状態になる．しばらくすると処理結果が Web ブラウザに表示される．(2) のとき，音声入力パネルは，正しく発話が録音できているかを利用者が視覚的に確認できるようにレベルメータとして動作する（赤く表示されたバーが入力レベルを示す）．また，(3)

[1] http://www.java.com/

の際も，送信・処理中の状態を示すような視覚的フィードバックを利用者に与えるように設計した．これらの工夫により，利用者の確実な発話の入力をサポートする．

3 関連研究と提案システムの特徴

Web 2.0 時代に登場した動画や画像をメディアとする Web サービスでは，動画や画像のマルチメディアデータをファイルとして交換することが多い．このため，既存の Web システムとの相性は良いと言える．しかし，音声に関しては，音を再生する Web ページは多いが，発話を入力とする Web サービスは依然として少ない．これは，ファイルアップロードが音声インタラクションの入力手段として適さないためである [6]．

このような状況において，発話を入力メディアとする Web システムを開発する試みは各所で進められている．W3C (World Wide Web Consortium) によって標準化が進んでいる VoiceXML[2]は，電話や Web システムにおいて，音声対話インタフェースを提供するための記述言語の規格である [7]．また，SALT (Speech Application Language Tags)[3]は，マイクロソフト社等によって提唱されている HTML (Hypertext Markup Language) の拡張規格である [8]．これらを利用するには，専用のボイスブラウザやプラグインプログラムをクライアント PC にセットアップする必要がある．その準備は，ユーザに負担を強いるものであり，強い動機を持つ者は別として，一般の人々に音声 Web アプリケーションに接する機会を与えることができるまでには至っていない．

楽曲検索サービス midomi[4]でも採用されているように，Flash[5]を使い，音声入力機能を Web ページに埋め込むことが可能である．この場合，利用に際しての事前のインストールは必要無い．しかし，サービスを提供する Web サーバ側に専用のプログラムが求められる等の理由で，開発者に負担を強いる．また，既存の CGI (Common Gateway Interface) や PHP の Web プログラムの資産や経験が利用できないことも問題となる．

MIT が開発している WebGalaxy[9, 10] は，Java を使った実装であり，提案システムと比較的似た構成を持つ．しかし，音声認識を背景とする自動対話システムの研究のための実装である．

これらの試みに対し，提案するシステムは以下のような特徴を有している．

1. Java アプレットを用いることでクライアント PC 側の事前インストール作業を不要する，
2. 既存の CGI プログラムやプロトコル等との統合が容易な実装を実現する（広く用いられている CGI の枠組みを土台とする），
3. 音声対話型を中心に，音声インタフェースの広い応用を可能とする汎用性を提供する，
4. オープンソースもしくはフリーソフトウェアを使いシステムを構築可能とする．

[2]http://www.voicexml.org/
[3]http://www.saltforum.org/
[4]http://www.midomi.com/
[5]http://www.adobe.com/products/flash/about/

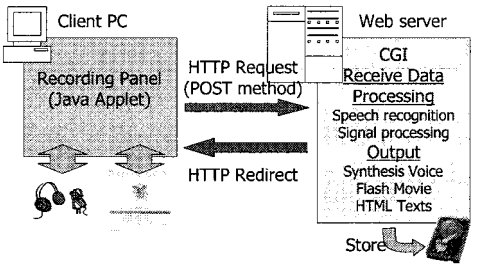

図 3: アーキテクチャの概略

本研究では，新たな規格を提案するのではなく，利用者の利便性を優先するとともに，既存資産（プログラム，プロトコル等）の拡張に基づくシステムの具現化を目指した．

しかし，本システムは，音声対話システムを実装するための抽象化記述言語を提供するものではない．基本的には，発話の入力と出力のフロントエンドのみを提供するものであり，自由は大きいが，入力データに対する処理や加工は Web 開発者自身が実装する必要がある．また，w3voice アプリケーションの実装には CGI プログラミングの知識が必要である．逆に CGI プログラムの経験があれば，開発キットに付属するサンプルプログラムを参考にすることでアプリケーションの開発は比較的容易である．例えば，音声掲示板システム[6]は，235 行の比較的小さい Perl スクリプトによって記述することができた．

4 w3voice のアーキテクチャ

以下では，提案システムの動作原理について概説する．図 3 で示すように，本システムは，大きくクライアント PC 側，Web サーバ側の二つの部分に分けることができる．また，処理の大半を Web サーバ側が担うサーバサイドアーキテクチャを採用している．

4.1 クライアント PC 側の構成

クライアント PC 側のプログラムは，前述の Java アプレットによる音声入力パネルである．Java アプレットに関する記述が埋め込まれた HTML ドキュメント（図 4）を Web ブラウザが読み込むことによって，自動的に起動する．録音が終

[6]http://w3voice.jp/keijiban/

```
<applet code="w3voice.class"
  codebase="http://w3voice.jp/applet/"
  archive="w3voice.jar"
  width="200" height="20">
<param name="SamplingRate" value="44100">
<param name="UploadURL"
  value="http://w3voice.jp/shop/upload.cgi">
</applet>
```

図 4: 音声入力パネルのために記述された HTML ドキュメント（抜粋）．UploadURL に送信先 CGI のアドレス，SamplingRate に録音時のサンプリング周波数を指定する．

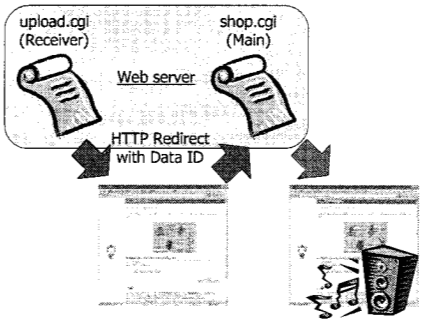

図 5: プログラム処理過程と HTTP Redirection

了した後に，音声入力パネルは，Web サーバに対して，HTTP (Hypertext Transfer Protocol)[11] の POST メソッドを用いて，データの送信を行う．HTTP は，Web のための通信プロトコルであり，その中でも POST メソッドは，画像等のファイルのアップロードのために一般的に使われる手法である．音声入力パネルは，そのプログラム自体が，Web ブラウザの動作をエミュレートし，POST リクエストを含む HTTP での通信を発行する．つまり，Web ブラウザの動作を真似ることで，データの送信を実現している．

このとき，送信するデータは，発話を AD 変換した無圧縮の WAV 形式オーディオデータである．16 bit の量子化波形信号であり，サンプリング周波数は音声入力パネルを呼び出す際にオプションで任意に指定可能とした．mp3 等の圧縮データはなく，raw audio data を扱うのは圧縮による波形の歪みの発生を抑えるためである．波形の歪みは，後処理となる音情報処理の際に精度の劣化につながるため避けることが求められる．

4.2 Web サーバ側の構成

Web サーバ側のプログラムは，Web サーバ内で動作する CGI プログラムである．開発には，CGI を記述できる大半の開発言語が使用でき，開発者の好みのものを選ぶことができる．一般に，Perl, Ruby, PHP 等の利用が想定される．

```
HTTP/1.1 302
Date: Sun, 18 Mar 2007 13:15:42 GMT
Server: Apache/1.3.33 (Debian GNU/Linux)
Location: http://w3voice.jp/shop/shop.cgi?q=ID
Connection: close
Content-Type: application/x-perl
```

図 6: HTTP Redirection を持つ HTTP レスポンス．ID はセッションを管理する固有の識別番号．

今回，この CGI プログラムは，さらに 2 つのプログラムから構成する（図 5）．そのうち一つは，音声入力パネルからのデータの受信を担い，もう片方は，音声の認識や加工，コンテンツの出力を行うシステム全体処理のメインプログラムである．図 5 の中で，"upload.cgi" と書かれたプログラムが受信を行うモジュールである．続いて，前述の果物通信販売の例では，"shop.cgi" が呼び出され，音声認識や合成，出力結果となるコンテンツの生成を行い，ブラウザに出力する．その過程は以下のようになる．

1. upload.cgi に対する HTTP の POST メソッドのリクエストパートとして Web サーバがデータを受信．受信した発話データは，ファイルとして保存される．
2. POST リクエストに対するレスポンスとして，HTTP Redirection を発行．shop.cgi のアドレスをブラウザに通知する．
3. HTTP Redirection を受けたブラウザが，転送先にアクセスを開始．
4. shop.cgi が実行され，認識・加工処理及び結果となるコンテンツを生成．
5. Web ブラウザに生成されたコンテンツが表示される．

Java アプレットと Web ブラウザは基本的には独立したプロセスである．そのため，収録データの送信を完了した音声入力パネルは，送信完了後に，結果出力を担う Web ブラウザへの処理の委譲を行う．ここで各プログラムの連動は HTTP Redirection を応用することで実装した．HTTP Redirection とは，HTTP の規格で定義されているレスポンスメッセージの一つであり，移転した Web サイトに対する自動転送を通知するのに広く用いられている．図 6 に，upload.cgi が生成した HTTP レスポンスを示す．最初の行の "302" は，HTTP Redirection を示すレスポンスコード（HTTP の規格で定義）である．HTTP Redirection の場合，転送先のアドレスがブラウザに通知される．"Location:" の行に書かれた URL が，この転送先のアドレスである．この例では，メインプログラムである shop.cgi がターゲットとなる．この行を参照することで，データの送信が完了した後に，ブラウザは shop.cgi にアクセスし，結果を得ることができる．

このように，本システムは標準的な通信規格の枠組みの中で実現されている．この結果，ファイ

表 1: w3voice.jp のアプリケーション一覧

音声認識・対話系

- Web Julius[12]，ディクテーション
  (`http://w3voice.jp/julius/`)
- Web たけまるくん [3]，音声対話
  (`http://w3voice.jp/takemaru/`)
- Web 通信販売のデモ（前述），音声対話
  (`http://w3voice.jp/shop/`)
- 音声認識ライブラリ w3voiceIM.js
  Web ページにある文字入力欄に音声認識機能を付与する Javascript ライブラリを配布．HTML に一行を追加するだけで通常の Web ページを音声入力対応にすることが可能．詳細は [13] を参照
  (`http://w3voice.jp/engine/`)

音声分析・合成系

- STRAIGHT[14] ボイスチェンジャ，音声分析・合成（変声器）
  (`http://w3voice.jp/straight/`)
- スペクトログラム，簡易音声分析・可視化
  (`http://w3voice.jp/specgram/`)

コミュニケーションツール

- おしゃべり写真 Voice Photo
  JPEG ファイルに発話を埋め込み，しゃべる写真（Flash の swf 形式）を生成
  (`http://w3voice.jp/VoicePhoto/`)
- 音声掲示板システム
  (`http://w3voice.jp/keijiban/`)

ヤウォールの内部からでも，Web をブラウジングできる計算機の大半からは，特別な設定無しで利用することができる．

5 公開試験と収集発話の分析

2007年3月9日より，`http://w3voice.jp/` の公開を開始した[7]．サービスの開始時期は異なるが，9月24日現在で，8種類のアプリケーションを提供している．表1にアプリケーションの一覧を示す．試験の目的は，提案システムの動作の安定性と有用性を確認することである．また，実環境の発話を集めることも目的とするため，Web サイトへの継続的なアクセスを得ることが必要である．

5.1 利用数

2007年9月24日までに，アップロードされた入力数は総計で9,758であった．各アプリケーションごとの内訳は，Julius 2,892，たけまるくん 2,052，通販デモ 722，音声認識ライブラリ 2,850，ボイスチェンジャ 387，スペクトログラム 445，おしゃべり写真 102，掲示板 209，その他（開発ツール付属のサンプル等） 99 であった．なお，この割

[7]入力された発話を蓄積し，研究活動の中で利用する旨を免責事項として Web サイト上に明示している．利用者には，それを承諾していただいた上での利用となる．

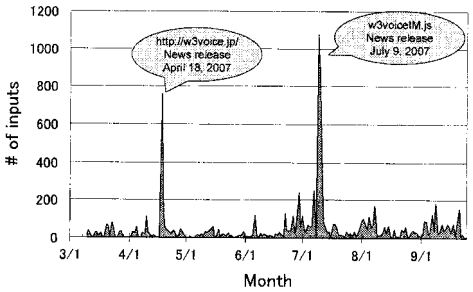

図 7: 日にちごとの入力数（0 秒入力は除く）

合は `http://w3voice.jp/` のページレイアウトが影響して変化することがわかっている．

9,758 の入力のうち，入力が 0 秒のもの（以下，0 秒入力）が 1,346 個であった．0 秒入力は，音声入力パネルのボタンを長押しせず，短いクリックをした場合に発生する．原因として，システムの利用方法がユーザに正しく伝わらなかったことと（利用方法の誤り），興味本意でのクリック（いわゆる冷やかし行為）のために生じたことが考えられる．

図 7 は，一日ごとの入力数の推移を示したグラフである（0 秒入力は除外）．4 月 18 日と 7 月 9 日は，それぞれ Web 上のニュースサイトに w3voice アプリケーションのリリースに関するアナウンス掲載されたため，爆発的に多くなっている．その影響も収まり，アクセス数が安定した 8 月以降，一日あたりの平均入力数は 47.6 であった．このように，継続的な利用を獲得しており，フィールドテストは成功を収めている．

5.2 発話時間とネットワーク負荷

次に，各入力が持つ発話時間の分布（0 秒入力は除外）を図 8 に示す．なお，本システムは push-to-talk のインタフェースであるため，録音された信号の長さを発話時間と見なした．

0 秒入力を除いたうちの 79.9%が 4 秒以下の発話であった．サンプリング周波数 44.1kHZ[8]の信号は 689k bit/sec. の容量を持つ．よって，大半は ADSL 等のブロードバンド回線では一秒以下で転送できる．つまり，本システムは，無圧縮の波形データをネットワーク転送するが，PHS 等のモバイル通信等の場合を除き，通信帯域による制限は，問題にならない．一方で，転送処理のストリーミング化が今後の検討課題として挙げられる．提案システムは，クライアント PC 側で録音された信号を一旦バッファする仕様であり，録音が終了するまで次の処理に進まない．このため，リアルタイム性は劣ることになり，改良が必要である．

[8]w3voice.jp では，いまのところ，44.1kHz サンプリングを標準の設定としている．JavaVM を通じたオーディオデバイスの動作において，他の周波数では不安定になる事例があり，それを回避したためである．

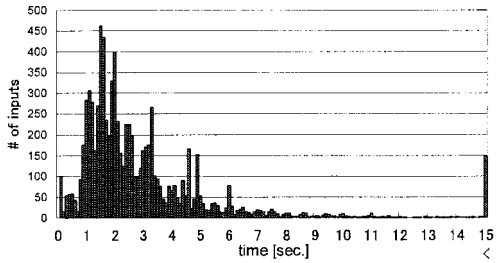

図 8: 入力の発話時間分布（0 秒入力は除く）

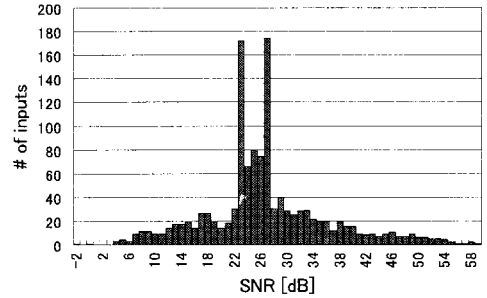

図 9: SNR の分布

5.3 録音環境（SNR）

利用者の録音環境の調査として，収録発話の SNR（Signal to Noise Ratio）を調査した．手動で音声および非音声区間の時間ラベルを設定した上で音声信号と非音声区間のパワーを求め，発話ごとの平均 SNR を算出した．対象とするデータは，0 秒入力を除外した収集データから抜き出した 1,250 個である．求めた SNR の分布を図 9 に示す．すべてを平均した SNR は 26.7 dB である．また，0 dB を下回ったのは 2 つの発話であった．

この結果からは，必ずしも録音の状態が音声認識に適しているとは言えない．しかし，SNR が 0 dB を下回るような発話は少なく，駅や街中等での劣悪な環境と比較して，Web システムでの音声入力は実現性が高い応用であると言うことができる．

なお，SNR が 0 dB 以下となった発話の原因を確認したところ，入力レベル不足とクライアント PC 内部で発生する悪質なノイズによる影響であった．前者に対しては，音声入力パネルにおいてゲインを調整する手法を検討したい．後者はシステム側からコントロールすることは難しい．今後，利用が増えると，このような事例は増える可能性があるから，調査を継続することで，その対策についても検討したいと考えている．

6 おわりに

本稿では，w3voice と名付けた，実用指向の音声 Web アプリケーションのフレームワークについて述べた．公開した w3voice アプリケーションによって集めた実環境発話の収集状況について報告した．今後は収集発話の分析を継続する予定である．

加えて，`http://w3voice.jp/` では，本システムのアプリケーション開発キットの配布を行う．研究成果を世に出すための一つの手段としてご利用いただき，作者へフィードバックをお願いしたい．

謝辞 本研究の一部は，文部科学省リーディングプロジェクト「e-Society 基盤ソフトウェアの総合開発」及び和歌山大学「平成 19 年度オンリー・ワン創成プロジェクト」の支援による．

参考文献

[1] A. Raux, et al., “Doing Research on a Deployed Spoken Dialogue System: One Year of Let’s Go! Experience”, *Proc. Interspeech2006*, pp.65–68, 2006.

[2] M. Turunen, et al., “Evaluation of a Spoken Dialogue System with Usability Tests and Long-term Pilot Studies: Similarities and Differences”, *Proc. Interspeech2006*, pp.1057–1060, 2006.

[3] 西村 他, “実環境研究プラットホームとしての音声情報案内システムの運用”, 電子情報通信学会論文誌, Vol.J87-D-II, No.3, pp.789–798, 2004.

[4] T. Cincarek et al., “Insights Gained From Development And Long-term Operation of A Real-Environment Speech-Oriented Guidance System”, *Proc. ICASSP2007*, 2007.

[5] 原 他, “汎用 PC 上で利用された音声対話システムによる音声収集と評価”, 情報処理学会研究報告, SLP-64-29, 2006.

[6] 西村 他, “Web ベースコースウェアのための音声入力システムの開発”, 情報処理学会論文誌, Vol.42, No.3, pp.605–613, 2001.

[7] S. McGlashan et al., “Voice Extensible Markup Language (VoiceXML) Version 2.0”, *W3C Technical Reports and Publications*, W3C, 2004.

[8] “SALT: Speech Application Tags (SALT) 1.0 Specification”, the SALT Forum, 2002.

[9] R. Lau et al., “WebGalaxy - Integrating Spoken Language and Hypertext Navigation”, *Proc. EUROSPEECH97*, pp.883–886, 1997.

[10] A. Gruenstein et al., “Scalable and Portable Web-Based Multimodal Dialogue Interaction with Geographical Database”, *Proc. Interspeech2006*, pp.453–456, 2006.

[11] R. Fielding et al., “Hypertext Transfer Protocol – HTTP/1.1”, RFC2616, The Internet Society, 1999.

[12] A. Lee et al., “Julius – An Open Source Real-Time Large Vocabulary Recognition Engine”, *Proc. EUROSPEECH2001*, pp.1691–1694, 2001.

[13] 西村 竜一, “音声入力 Web システムを用いた辞書共有型音声認識サービス”, 日本音響学会 2007 年秋季研究発表会講演論文集, pp.61–62, 2007.

[14] H. Banno et al., “Implementatioin of realtime STRAIGHT speech manipulation system: Report on its first implementation”, *Acoustic Science and Technology*, Vol.28, No.3, pp.140–146, 2007.