ヒューマンインタフェース 46－3
（1992. 1. 14）

ワークステーションにおける音声認識インタフェースの検討

橋本秀樹† 永田仁史†† 竹林洋一††

†東芝ソフトウェアエンジニアリング(株)
††(株)東芝 研究開発センター 情報・通信システム研究所

川崎市幸区小向東芝町一番地

あらまし 本報告では汎用ワークステーションにおける音声認識インターフェースについて述べる。マルチウィンドウ環境下で、マルチタスクへの音声入力を可能にするため、音声認識機能をサーバ化した。また、音声認識サーバは、汎用ワークステーションに標準装備されているCODECを利用し、外付けハードウェアなしで実現した。認識方式として複合類似度法を用いており、特定話者/不特定話者/話者適応型の単語認識を高精度に行う。音声認識サーバのクライアントとして電子メールシステムとDTPシステムを作成して使い勝手を調べ、クライアント・サーバモデルに基づく音声認識インターフェースの有効性を確認した。新開発の音声認識サーバとソフトウェア音声認識の利用により、ハードウェアに依存しない広範な応用が期待できる。

Speech Recognition Interface for General Purpose Workstation

Hideki Hashimoto† Yoshifumi Nagata†† Yoichi Takebayashi††

†Toshiba Software Engineering Co., Ltd.
††Toshiba Corporation, Research and Development Center,
Communication and Information Systems Research Laboratories

1 Komukai-Toshiba-cho, Saiwai-ku, Kawasaki 210, Japan

Abstract We have developed a speech recognition interface based on client-server model for general-purpose workstation. The speech recognition server enables applications to handle speech input under a multi-window environment using isolated word recognition without any additional hardware. High accuracy has been achieved for speaker-dependent, -independent, and -adaptive recognition systems using the Multiple Similarity method. We have developed an E-mail system and a DTP system using the speech recognition server. Experimental results have shown the effectiveness of the speech recognition interface based on the software speech recognition server. The speech recognition interface is applicable to various application systems utilizing the software speech recognition function under multi-task environment.

1.はじめに

ペン入力や音声入力など、キーボードやマウス以外の入力メディアが注目されてきており、音声メディアを利用したHIに関する研究が盛んになってきた[1-4]。音声は人間にとって自然な入力手段であるが、音声認識には多大の計算量が必要であり、また音声入力手段を既存のインタフェースに統合する枠組みが確立されていないなどの問題があるため、広範囲に応用されるに至っていない。

そこで我々は、既開発の高速アルゴリズムを応用した高精度なソフトウェア音声認識技術[5]に基づき音声認識機能をサーバ化することにより音声入力メディアの統合化の問題の解決をはかった。本報告では、ソフトウェア音声認識サーバとその応用について述べる。

2.ワークステーションにおける音声認識の利用

2.1.ワークステーションの高性能化

音声認識処理には多大な計算量を要するため、従来DSPなどの音声認識専用の特別なハードウェアを必要としていたが、我々はこれまで、汎用ワークステーションの計算能力を利用し、高速アルゴリズムによる実時間で動作するソフトウェア音声認識システムを開発した[5]。このシステムは、複合類似度法[6]をベースに、特定・不特定話者に対応した高精度の音声認識が可能である。我々のシステム以外にも、特定話者対応のソフトウェアのみによる音声認識システムが出現し、ハードウェアに依存しない音声認識を、ユーザが手軽に利用可能になってきた。

2.2.音声入力によるHIの改善

音声による高精度なテキスト入力や計算機との対話を目的に、連続音声認識の研究[7-9]が行われているが、連続音声の認識性能は、実験室環境で評価されているレベルにあり、現状のワークステーションに音声認識機能を組み込み、実環境における利用を考えると、孤立単語を認識対象とする方が現実的である[10,11]。孤立単語認識を利用した音声入力をキーボードやマウスのように標準的な入力手段とするためには、認識性能とともに、音声認識インタフェースの実現方法が重要である。特に、計算機パワーの増大に伴い、マルチタスクを支援するワークステーションが主流となりつつあるが、これまで、マルチタスク環境などの既存のインタフェースに音声入力手段を統合する枠組みは確立されていない。マルチタスクを前提とした音声認識インタフェースに関して、幾つかの試みがなされており、以下それらの特徴について述べる。

SchmandtらのXspeak[12]は、マルチウィンドウ環境でのウィンドウ操作に音声を応用した例である。マウスや音声によって指定した1つのウィンドウに対して、キーボード入力と音声入力を行うことが出来る。既存の複数のプログラムに音声入力を可能にする点では優れた方法である。しかし、音声入力は全てXウィンドウの介在によってXのイベントに翻訳されるため、応用プログラムが音声入力を直接受けとっているとはいえず、また、応用プログラムから音声認識機能に情報をフィードバックするチャネルが存在しないため、ユーザに対する肌理の細かい処理を提供できない等の問題がある。

Rudnickyらは、CMUの不特定話者連続音声認識システム(SPHINX)に基づいたネットワーク環境下における音声認識インタフェースの一般的なモデル[13]を提案している。このモデルによれば、複数の応用プログラムが連続音声認識機能を共用できるという利点があり、高価な音声認識装置の利用に関して有用な方法である。しかしリアルタイム処理や、パーソナルなワークステーション上での利用形態についての検討は十分ではない。

以上の点を考慮し、我々は高い認識性能を確保できる孤立単語を認識対象とし、応用拡大が図りやすいソフトウェア音声認識技術を採用し、音声認識機能のサーバ化によって、HIの質の向上を図った。以下では、ソフトウェア音声認識方式、音声認識サーバに基づく音声認識インタフェースの実装、および応用例とHIの改善について述べる。

3.ソフトウェア音声認識

3.1.複合類似度法によるソフトウェア音声認識

音声認識機能は図1に示すように音声認識部、単語登録部、辞書作成部から成る。音声認識部では音響処理によって得られた単語の特徴ベクトルを用いて認識辞書との複合類似度によるマッチングを行い、認識結果を出力する。単語登録部では、学習用の単語データを認識することにより不良な音声データの混入を防ぐことができる。辞書作成部では単語登録部で得られた単語特徴ベクトルを用いて複合類似度法の辞書を作成する。

最近の汎用ワークステーションには電話帯域のCODECが内蔵されており、サンプリング周波数8kHz、量子化8bitでオーディオ入出力が行える[14]。

音声認識部では、まず、単語の始終端を音声エネルギーの時間変化により検出した後、単語区間を等間隔に分割し、時間軸上のサンプル点を決めてFFTによる周波数分析を行い、周波数分析結果を平滑化し例えばフィルタバンク出力を得、時間周波数パターンを単語特徴ベクトルとして使用する。

本音声認識の最大の特徴は外付けのハードウェアを全く使用せずに全てソフトウェアのみによって認識処理を行う点である。従来の音声認識の処理を図2(a)に、高速化したアルゴリズムを図2(b)に示す。複合類似度法の認識アルゴリズムでは、時間方向のサンプル点(分析フレーム)すべてにおいて周波数分析を行った後に始終端を検出していたので、音声分析のために従来はDSP等の専用ハードウェアが必要であった。これに対して、高速化アルゴリズムでは、図3に示すように始終端検出された音声区間において等間隔にリサンプルしたデータについてだけ周波数分析を行うので大幅に計算量を削減できる。

音声入力がない場合にはパワー計算等を含む始端検出だけが働いているため、音声入力時以外は計算量が少なくてすむという長所がある。これらにより、ソフトウェアのみによる音声認識システムがマルチタスク環境下でもリアルタイムで動作可能となった。

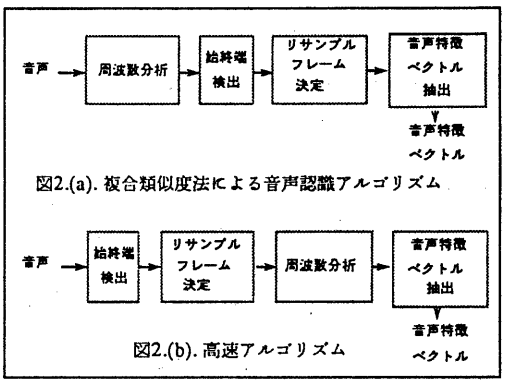

図2.(a). 複合類似度法による音声認識アルゴリズム

図2.(b). 高速アルゴリズム

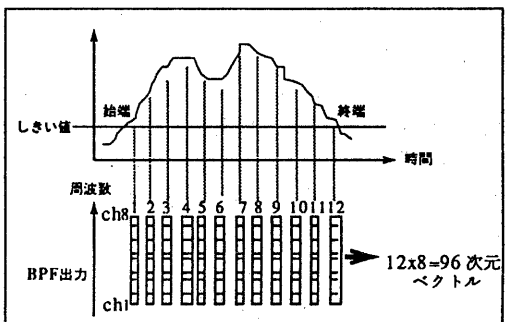

図3. リサンプリングによる特徴ベクトルの抽出

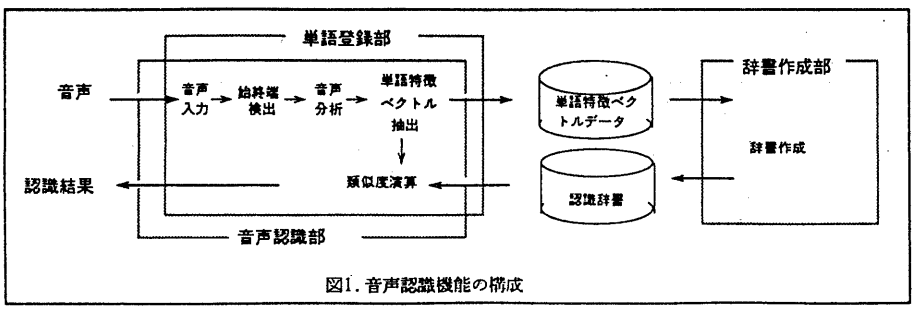

図1. 音声認識機能の構成

類似度演算部において、得られた単語特徴ベクトルと認識辞書との複合類似度を計算する。次式に複合類似度の定義を示す。

$$S^{(l)}[X] = \sum_{m=1}^{M} \frac{\lambda_m^{(l)} (X, \phi_m^{(l)})^2}{\lambda_1^{(l)} \quad \| X \|^2} \quad (1)$$

ここでS(l)[X]はカテゴリーlの類似度値、Xは入力単語特徴ベクトル、Mは軸数、λ(l)m、λ(l)1は固有値、φ(l)mは固有ベクトルである。上記φ(l)mは多数のパターン(単語特徴ベクトル)からK-L展開を用いて求められる。ソフトウエアにより各単語の複合類似度値を求め、単語認識を行う。

3.2 部分抽象化による認識性能の向上

本システムでは、単語全体の時間周波数スペクトルを単語特徴ベクトルとして複合類似度法により単語単位のマッチングを行う。単語単位の認識は、単語全体を一つのパターンとすることにより単語パターンの全体的かつ動的特徴を表現でき、雑音下の不特定話者を対象としたシステムに対しても高い認識性能を得られている。さらに、単語辞書を変更するだけで外国語にも適用可能であるという利点もある。

また、単語パターンは部分抽象化[15]により異なった視点から複数の次元数の異なる単語特徴ベクトルとして表現され、例えば図4(a)周波数分解能の高い表現、(b)前半分の詳細な表現、(c)後半分の詳細な表現、(d)詳細な時間周波数表現等が可能となる。

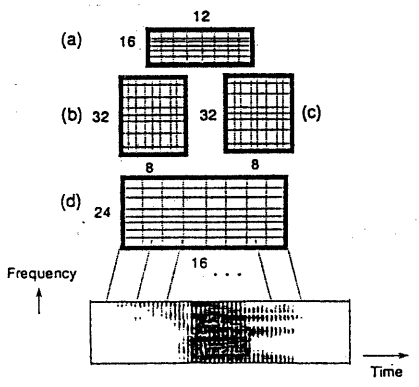

図4.単語特徴ベクトルの部分抽象化による表現

この部分抽象化により、異なった観点からの単語音声の照合を同一の複合類似度をベースに行え、一般に、各表現で誤認識の傾向が違うので、統合処理により認識性能の向上が可能である。特に、単語表記の一部が共通する単語ペアが認識語彙に存在する場合には、部分抽象化の効果が期待できる。ここでは、単語全体、単語の前半分、後半分の表現を用いて認識性能の向上を図った。

3.3 認識性能評価

本音声認識装置の不特定話者単語認識の性能評価を行った。認識対象語彙は表1に示すDTPシステムの操作等に必要な26単語である。単語音声データは男性100名から収録した。その100名のデータから5種類のデータセット(80名学習、20名評価)を作成し、5種の評価データに対する認識率の平均値を求めた。表2に実験条件を示す。5セット分平均の認識率は部分抽象化前が98.0%、導入後が99.2%であった。特に単語全体パターンを用いた場合には「グループ化」「グループ解除」或いは「グループ化」「スムース化」等の単語ペアにおいて認識誤りが見られたが部分抽象化の導入により大幅に改善された。ここで部分抽象化は160次元の単語の前半分、後半分のパターンを用いて行った。

また、特定話者単語認識の性能評価として、電子メールツールの操作コマンド40単語

表1. 認識実験に用いた語彙

取り消し	コピー	文書	編集
カット	ペースト	クリア	検索
文字書式複写	コンディション複写		大文字小文字変換
グループ化	グループ解除	フロント	揃え
均等配置	変形	スムース化	スムース解除
うえした反転	みぎひだり反転	回転	スケール
多角形火	属性	グラビティー	

表2 実験条件

サンプリング周波数	8kHz
フレーム周期	8msec
ブロック長	32msec
FFT ポイント数	256 点
単語特徴ベクトル次元数	256次元

を認識対象語彙として実験を行った。話者は男性1名女性3名で、それぞれ各単語45パタンを学習用に45パタンを評価用に用いた。実験条件は、上述した実験と同様である。結果は4名の平均で99.3%の認識性能であった。

また、上記の実験では不特定話者に関しては学習用のパターン数が不十分であったが、単語特徴ベクトルの次元数を増加させることによって、性能向上も可能であるが、次元数と処理時間とのトレードオフがある。メニュー選択の語彙程度ならば現状のワークステーションを用いれば処理速度としては十分であるが、計算機の処理速度が上がれば語彙数の増大も可能である。

以下では、上述した高精度のソフトウェア単語認識機能を効果的に使用し、マルチタスク環境下でワークステーションの対話性を向上させるための音声認識インタフェースについて述べる。

4.音声認識インタフェースの実装

4.1.クライアント・サーバモデルの応用

従来、音声入力の応用における音声認識インタフェースは、図5に示す構成をとっていた。図5(a)は、応用プログラムの中に認識機能を一体化して持つ方法、図5(b)は、音声認識ドライバを介在させる方法、図5(c)は、音声認識結果によってキー入力をエミュレートする方法である。図5(a)や図5(b)においては、応用プログラムによる音声認識機能の直接制御が可能であるが、特定プログラムによる認識機能の占有が問題となる。図5(c)においては、既存の応用プログラムを変更する必要がない利点があるが、情報の流れが一方向であり、入力可能な語彙の変更など、応用プログラム側の内部状態の変化を認識機能にフィードバックできないという問題がある。上記の構成はどれも、マルチタスク環境における利用を前提としてはいない。我々は、図5(d)に示す構成のような、クライアント・サーバモデルに基づく音声認識機能のサーバ化によって、これらの問題の解決をはかった。

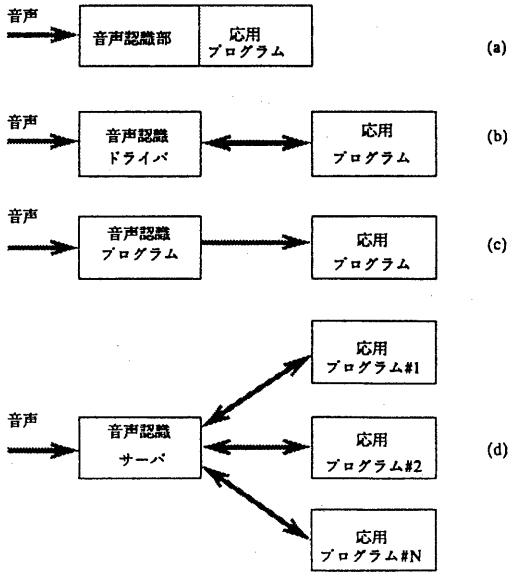

図5.音声認識インタフェースの構成

クライアント・サーバモデルとは、処理を要求するタスク(クライアント)と処理を実行するタスク(サーバ)とを分離し、その間で通信を行うことによって処理を進める方式である。音声認識機能をサーバ化してオーディオ入力装置や認識辞書を管理させ、音声認識の処理機能をサーバに集中させる。クライアントは、サーバと通信を行って音声認識のサービスを受ける。複数のクライアントがサーバの介在によって音声認識用の資源を共有し、音声認識機能を利用できる。クライアントと音声認識サーバは、通信によって情報の交換ができるため、クライアントの内部状態の変化に伴う認識対象語彙の変更などの情報を、サーバが利用可能である。

4.2.音声認識サーバ

我々が試作した音声認識サーバは、汎用のワークステーションSPARC Station上に、3章で述べたソフトウェア音声認識技術を利用して実装した。音声認識サーバとクライアントは、通信によって以下の手順で処理を行う。音声認識サーバは、マルチタスク環境において1つのプロセスとして常時動作し、クライアントからの要求を待つ。クライアントは、音声認識サーバに対して認識対象語彙を通知し、認識処理を依頼する。音声認識サーバ

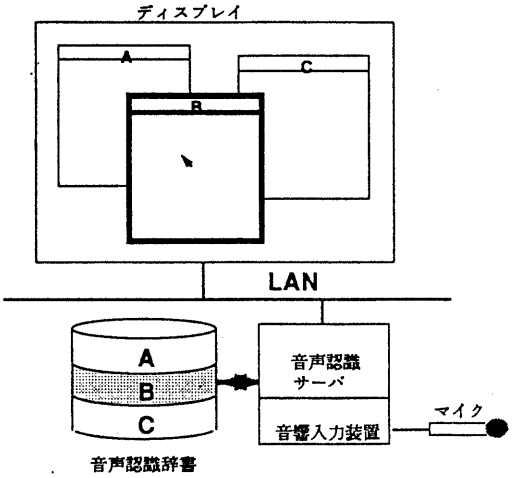

図6.ワークステーションにおける音声認識サーバの実装

は、依頼に従って認識処理を行い、結果をクライアントに送信する。クライアントは認識結果を受け、応用に従った任意の処理を行う。

認識結果の送信先は、音声フォーカスにより指定する。音声フォーカスの役割は、認識対象語彙を特定クライアントに関するものに限定して誤認識を防ぎ、また特定クライアント以外への音声認識結果の送信を防止することである。クライアントは、キーや音声などの外部入力や、プログラム自身の内部状態の変化をきっかけに、音声フォーカスを自分や他のクライアントに任意に設定できる。

図6は、音声認識サーバをウィンドウシステムと組み合わせた利用例である。A,B,Cのウィンドウが、音声認識サーバのクライアントであり、クライアントBに音声フォーカスが当たっている。音声入力に対し、サーバは、クライアントBが依頼した認識対象語彙を利用して認識処理を行い、結果をクライアントBに送信する。

Xspeakでは、キーボード入力と音声入力という2つの入力メディアを、常に1つのタスク(ウィンドウ)に結びつけていた。我々はこれに加えて、キーボード入力と音声入力を別個のタスクに結びつけ、複数の応用プログラムを別個の入力チャネルを通じて同時に制御可能にした。分離した入力対象タスクをユーザが判別できるようにウィンドウ環境におけるactiveなタスクを例えば図7の様に表示する。この図では、キーボードフォーカスをウィンドウ枠を太くして表現し、音声フォーカスをウィンドウタイトルの色の変更で表現している。音声フォーカスは、音声(ウィンドウ名)およびマウスにより設定する。

クライアントとサーバ間の通信プロトコルを、例えばXウィンドウシステムのXプロトコルの様に標準化すれば、特定のハードウェアに依存しない音声の応用が可能である。我々は音声認識サーバとクライアント間の通信プロトコルを策定し、C言語で呼び出し可能なクライアント・プログラミング・ライブラリを作成し、応用プログラムの試作に利用した。ライブラリを用いた応用プログラムからのサーバとの通信手順を図8に示す。

サーバ化に伴う音声認識機能の仮想化と、ソフトウェアのみによる音声認識機能の実現

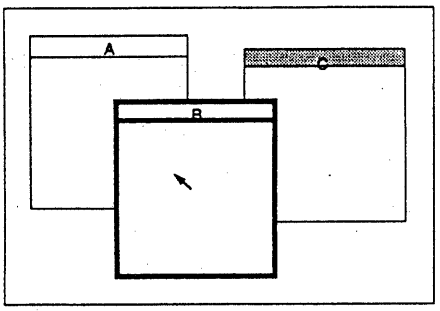

図7.マルチウィンドウ環境における
音声とキーボードの入力対象の表現

(1)音声認識サーバとの通信路を確保する。
(2)認識対象語彙をサーバに通知する。
(3)認識結果の送信をサーバに依頼する。
(4)音声フォーカスをクライアント自身に設定する。
(5)音声認識サーバからのメッセージを待つ。
(6)受信メッセージに従って処理を行う。
(7)(5)以降の処理を繰り返す。
(8)サーバとの通信路を閉じる。

図8.クライアントのサーバとの通信手順

によって、ハードウェアに依存しない標準的な音声認識の利用が可能となった。同時に、ワークステーションのマルチウィンドウ環境において、複数の応用プログラムに対する音声入力の利用が可能となった。

4.3.マルチタスクの支援

GUI(Graphical User Interface)とは独立したプログラムである音声認識サーバを、GUIを使って制御し、ユーザの音声入力を支援するために、音声インタフェースマネージャを作成した。特権クライアントの音声インタフェースマネージャは、GUIを利用して、オーディオ入力装置や音声フォーカスの制御手段をユーザに提供し、認識対象語彙の表示を行う。音声フォーカスや認識対象語彙の変更をサーバに通知させることで、認識対象語彙リストを動的に変更・表示する。音声インタフェースマネージャの提供する機能により、ユーザはクライアントを統一的に管理し、発声時の負担(応用プログラム毎の認識対象語彙を記憶する)が軽減できる。

5.音声認識の応用

音声認識サーバの評価を目的に、ワークステーション上でソフトウェア音声認識により制御が可能ないくつかの応用プログラムをクライアントとして試作した。

5.1.音声メールツール

音声入力の使用を想定し、図9に示すような表示画面を用いる電子メールシステム(音声メールツール)を作成した。音声操作により受信メールの内容を確認したり、返事を送信できる。

ツールの上部がリスト表示部、中央が受信メール表示部、下部が送信メール編集部である。リスト表示部において指定したメールを、受信メール表示部に表示する。送信メール編集部でメールを作成し送信する。音声メールツールの認識対象語彙を図10に示す。「上司」や「緊急」は、音声マクロコマンドとして実装されており、メールのヘッダを正規表現により照合した結果を用いて受信メー

オープン	クローズ	セーブ	終了
送信	返事	アンドゥ	コピー
カット	ペースト	クリア	引用
サイン	前	次	先頭
最後	機能	今日	全選択
アイコン化	同僚	上司	連絡会
会議通知	緊急	本社	研究所
大学	優先	海外	

図10.音声メールツールの認識対象語彙例

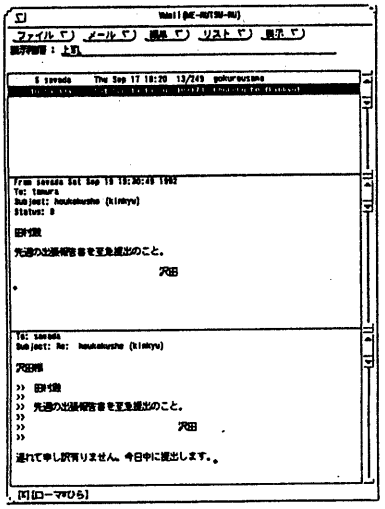

図9.音声メールツール

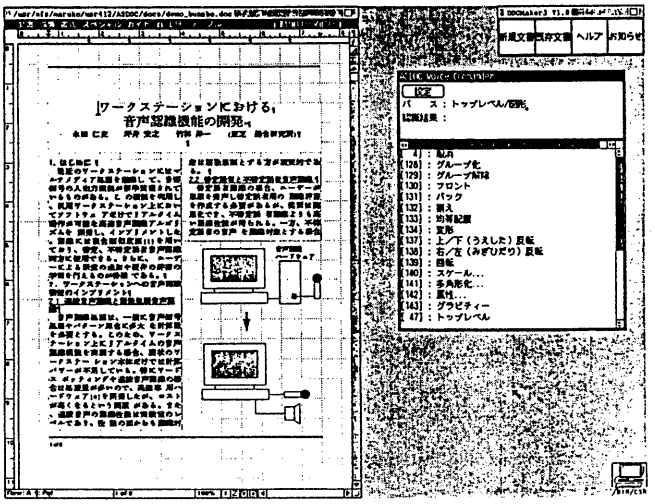

図11.DTPシステムへの応用

ルのリストを限定・表示するものである。

受信したメールを検索し、返事を送るといった繁雑な作業も、音声入力を使って簡単に行うことができる。

5.2.DTPシステムへの応用

既存の応用プログラムへの音声認識サーバへの利用の例として、DTPシステムへの応用を検討した。図11の左がDTPシステム、右がDTP制御プログラムである。

DTPの制御プログラムは、DTPシステムの提供するAPI(Application Programming Interface)を利用してキーボードコマンドを送信する。DTPシステムのメニュー階層を利用して認識対象語彙を限定する。メニュー階層のルートを「トップレベル」と呼び、トップレベルから単語を発声し、メニュー階層をたどることによってコマンドを実行していく。メニューの階層を移動する毎にウィンドウにメニューの各項目と、メニュー階層における現在位置をパスの形で表現し、ユーザに呈示する。

操作対象をマウスで選択し、「カット」「ペースト」「上下反転」などの対象に対する操作を音声で行い、2つのメディアを自然な形で役割分担できるため、図形編集操作が簡単に行える。

6.むすび

ワークステーションにおける標準的な音声認識機能の実装法として音声認識サーバを検討し、試作した。本稿で述べた音声認識機能のサーバ化によって、複数の応用プログラムの音声による操作が可能となった。また、実時間処理が可能な高精度のソフトウェア音声認識技術により、ハードウェアに依存しない広範な応用が可能である。今後、音声メディアを利用したより使い勝手のよいHIの確立を目指し、本システムの評価と改良を進めて行く予定である。

参考文献

[1] 竹林,永田,瀬戸,新地,橋本:「音声自由対話システムTOSBURGII-マルチモーダル応答と音声応答キャンセルの利用-」,情処学会HI研究会,HI-45-13 (1992)

[2] 佐藤,開,安西:「ロボットとの対話:センサ情報を利用した音声対話システム Lintaの設計と実装」,AI学会研究会,SIG-SLUD-9202-3 (1992)

[3] 中谷,守屋:「文書編集における音声制御の一方式」,情学論,Vol.33,No.2,pp.195-203 (1992)

[4] 河又,宮島,早川,並木,高橋:「表示一体型タブレットを用いた"未"ウィンドウシステムの設計と実現」,情処学会HI研究会,HI-45-17 (1992)

[5] 永田,竹林:「ワークステーションにおける音声認識機能の開発」,信学技報,HC-9119 (1991)

[6] 麻田,坂田,高橋,竹林,平井,篠田,新田,渡辺:「ハイブリッド構造マッチング法による電話音声の認識」,音講論,1-4-16 (1982-3)

[7] 竹沢,大倉,森元,嵯峨山,樽松:「日英音声言語翻訳実験システム SL-TRANS2」,音講論,1-5-24,pp.47-48 (1991-10)

[8] 畑崎,渡辺,磯谷,塚田,野口,坂井,古賀,吉田:「半音節を認識単位とする不特定話者連続音声認識システム」,信学技報,SP90-83 (1990)

[9] 中川:「文脈自由文法のフレーム同期型構文解析法による連続音声認識」,信学論,Vol.J70-D,No.5,pp.907-916 (1987)

[10] 竹林,金沢:「ワードスポッティングによる音声認識における雑音免疫学習」,信学論,Vol.J74-D-II,No.2,pp.121-129 (1991-2)

[11] 金沢,坪井,竹林:「不要語を含む連続音声からの単語検出」,信学技報,SP91-22 (1991)

[12] C.Schmandt, M.Ackerman and D.Hindus : "Augmenting a Window System with Speech Input", IEEE COMPUTER, No.23, pp.50-58 (1990)

[13] A.Rudnicky, J.Lunati and A.Franz : "Spoken language recognition in an office management domain", Proc. ICASSP'91, pp.829-832 (1991)

[14] Sun Microsystems : "Sun OS 4.1.1 Release & Install Manual", pp.1356-1360 (1990)

[15] 竹林,金沢,坪井:「単語特徴ベクトルの部分抽象化によるキーワードスポッティング」,信学技報,SP91-104 (1991)