2004－ＨＩ－109　(8)
2004－SLP－ 52　(8)
2004／7／16

言語情報を用いない音声による直接操作インタフェース

五十嵐　健夫[1]　　John F. Hughes[2]
[1]東京大学　[2]ブラウン大学

音声入力を利用したシステムでは主に音声の持つ言語情報を利用している．すなわち，音声認識エンジンが音声入力をテキストやコマンドといった離散的なシンボル列に変換し，それを入力としてシステムが動作する．このような方法は，連続音声認識によるテキスト入力や，自然言語によるエージェントとのインタラクションといった目的には適しているが，ユーザの発声から結果のフィードバックまでに時間がかかるために，画面をスクロールするとった連続的かつ機械的な操作には適していない．本論文では，音程や発声の有無といった，音声のもつ音響信号としての側面に注目することによって，機械的な直接操作を効率的に実現する手法を提案する．

Voice as Sound: Using Non-verbal Voice Input for Interactive Control

Takeo Igarashi[1]　　John F. Hughes[2]
[1]The University of Tokyo　[2]Brown Univeristy

We describe the use of non-verbal features in voice for direct control of interactive applications. Traditional speech recognition interfaces are based on an indirect, conversational model. First the user gives a direction and then the system performs certain operation. Our goal is to achieve more direct, immediate interaction like using a button or joystick by using lower-level features of voice such as pitch and volume. We are developing several prototype interaction techniques based on this idea, such as "control by continuous voice", "rate-based parameter control by pitch," and "discrete parameter control by tonguing." We have implemented several prototype systems, and they suggest that voice-as-sound techniques can enhance traditional voice recognition approach.

1　はじめに

音声入力を利用したシステムでは主に音声の持つ言語情報を利用している．すなわち，音声認識エンジンが音声入力をテキストやコマンドといった離散的なシンボル列に変換し，それを入力としてシステムが動作する．このような方法は，連続音声認識によるテキスト入力や，自然言語によるエージェントとのインタラクションといった目的には適しているが，ユーザの発声から結果のフィードバックまでに時間がかかるために，画面をスクロールするとった連続的かつ機械的な操作には適していない．本論文では，音程や発声の有無といった，音声のもつ音響信号としての側面に注目することによって，機械的な直接操作を効率的に実現する手法を提案する（図１）．なお、本手法を説明したデモビデオが www-ui.is.s.u-tokyo.ac.jp/~takeo から入手可能である．

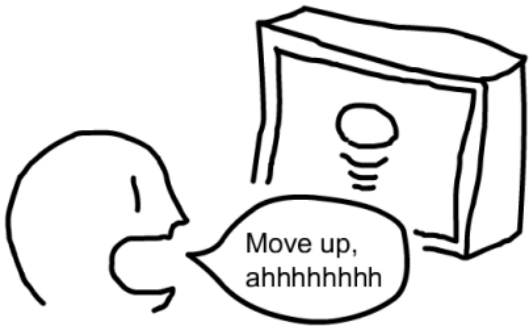

図１　概念図

2　関連研究

非言語情報を利用した例としては，まず従来型の音声認識の認識率向上のために利用したものが挙げられる[2][4][5]．また，会話型のインタフェースシステムにおいて，相槌のような非言語情報からユーザの状態を推定することによってインタラクションを支援する方法が提案されている[7]．後藤らは，音声入力中の有声休止をトリガ

ーキーとして残りの単語を補完する「音声補完」を提案している[1].

音声によって機械的な直接操作を実現した例として SUITEKey システムがあげられる[6]. これは音声のみによってウィンドウシステムを操作するシステムで, たとえばマウスを動かす場合には「上へ移動…ストップ」というように間に時間を空けて発話すると, 「ストップ」と発話されるまでの間マウスカーソルが移動を続ける. 本論文では, このような連続的・機械的操作をより効率よく行う方法をいくつか提案する.

3　提案するインタラクションテクニック

1）声の有無によるオン・オフ操作

本手法を用いたインタフェースでは, ユーザの声がオン・オフボタンとして動作する. すなわち, ユーザが連続的に声を出しつづけている間はボタンが押され続けているものとみなされ, 逆に発声がない間はボタンから手が離されているものとみなされる（図２）. 例えば, ユーザが「ボリュームアップ, あーーーー」と発声すると, 「あーーーー」と発声している間ボリュームが上がりつづける. これによって, ユーザはシステムのフィードバックを連続的に観察しながら入力を行うことができる. なお, この手法を利用する際には, 有声音だけでなく息をマイクに吹きかけるといった無声音を利用することができる. 有声音にくらべて無声音は喉への負担が少ない他, 他人への迷惑が最小限に抑えられるといったメリットがある.

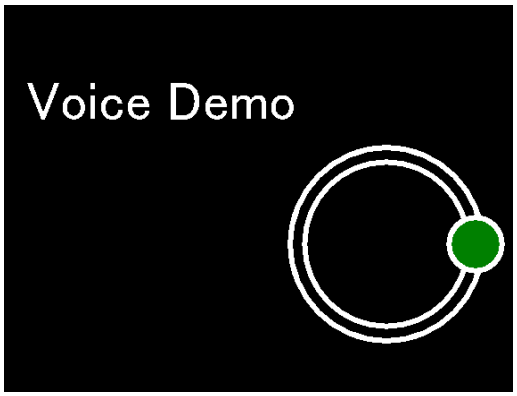

図２：オン・オフ操作のデモ。声を出している間、ノブが回転を続ける。気に入った場所で声を止めれば回転が停止する。「右３６度」などと言って認識結果を待ってさらに言い直す、といった通常の音声認識を利用した方法よりもスムーズな操作が可能になる。

2）　音程によるパラメータの増減操作

本手法は, 前出の手法を拡張するもので, 連続的な発声の間に音程を変化させることによって補助パラメータの制御を可能とする. 言い換えると, 音声が一つのパラメータを制御するジョイスティックやスライダーなどと同様の動作をするものである（図３）. 例えば, ユーザが「上へ移動, あーーー」と発声して「あーーー」の発声が続く間だけ移動が起こるような場合に, 途中で音程をあげることで速度を上げたり逆に音程を下げることで速度を落としたりできる. 移動速度に基づく自動ズーミングなどと組み合わせることによってより効率的な動作を行うことも可能である[3].

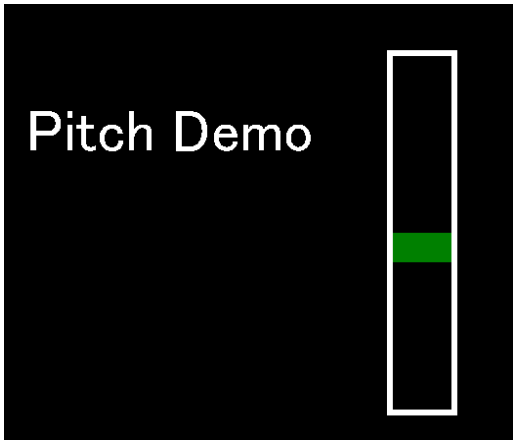

図３：音程を利用した操作のデモ。音程が徐々に上がるように発声するとスライダーのノブが上に移動し、徐々に下がるように発声するとノブが下へ移動する。絶対的な音程をとるのでなく、音程の相対的な変化を見ることで安定した操作が可能になる。

3）タンギングによる離散的パラメータの増減操作

前出の２手法はテレビの音量のような連続的なパラメータの制御に使われるものであるが, 本手法はチャンネルの切り替えのような離散的パラメータの制御に使用するものである. 例えば, ユーザが「チャンネルアップ, たったったったっ」と発声すると, テレビのチャンネルが４つ上に切り替わる（図４）. この手法も前出の手法と同様, 音響信号を見ているだけなので, 声でなく, 手を叩いたり指を鳴らしたりすることでも同様の効果を得ることができる.

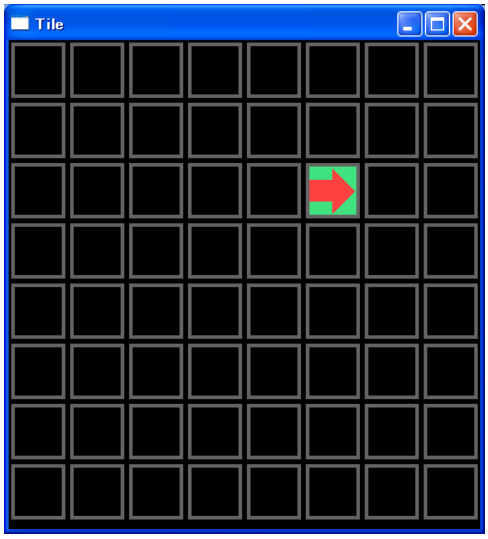

図４：タンギングを利用した操作のデモ。方向を音声で指示した後、タッタッタッと発音するとその方向へ、タンギングした数だけカーソルが移動する。

4　応用例

１）地図のナビゲーション

「上」「右」と発音すると、それぞれの方向へ移動するモードに入る。その後で、「あーー」と発声を続けると、発声している間その方向へ移動を続ける。さらに移動中に音程を上げ下げすることで移動速度を調整することができる。また、「ズームイン」と言ってから「あーー」と発声することによりズームレベルを連続的に調整することができる。

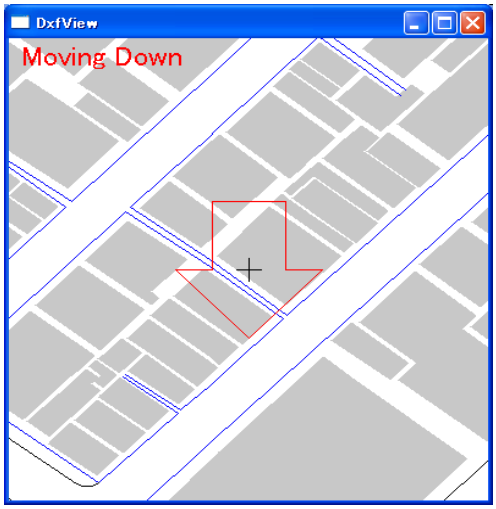

図５：地図のナビゲーション

２）音の有無によるゲーム

音声を発生するとジャンプする。発声を続けると高くジャンプできるが、上限が決まっているのでそこまで上がってしまうと下降を始める。敵を避けるためには、タイミングよく発声を繰り返さなくてはいけない。

図６：音の有無によるゲーム

３）音程によるゲーム

主人公のキャラクタの位置が、ユーザの発する音声の音程によって上下する。敵は右から左へとやってくるので、それを上下操作で避けなくてはならない。

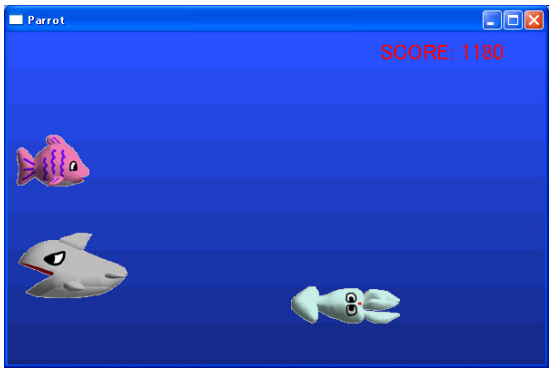

図７：音程によるゲーム

４）タンギングを利用したゲーム

「あたたたた」と発声すると、「た」を発声した数だけパンチが当たる。規定回数発声した後に「たーーー」と伸ばすと敵を退治することができる。

図８：タンギングを利用したゲーム

5　実装

提案手法の有効性を確かめるためにいくつかのプロトタイプシステムを実装した．マイクからの信号からスペクトルを計算する部分は C++で実装し，音声認識や音程の計算，およびグラフィカルな表現部分は Java で実装している．現在の実装では，入力音響信号を量子化ビット数 16bit，標本化周波数 22kHz でサンプリングし，窓関数として 2024 点からなるハニング窓を用いた短時間フーリエ変換を，高速フーリエ変換(FFT)によって計算している．FFT のフレームは 256 点づつシフトするため，処理の時間単位となるフレームシフト時間は 12msec である．

発声の有無の検出は，スペクトルの合計を閾値にかけることで判断している．ノイズの影響を除去するため，低周波領域(375Hz 以下)はカットして計算している．音程の変化は，1 フレーム前(-12ms)の元のスペクトルと，1 フレーム後でかつ周波数を上下にシフトさせた(±43Hz)スペクトルとの間の内積を比較することで検出している．すなわち，元のスペクトルと後で周波数を上にシフトさせたものとの内積が，周波数を下にシフトさせたものとの内積よりも大きい場合には，音程が上がったと判断し，逆の場合には下がったと判断される．ユーザの発声する音程が一定の場合には，上昇分と下降分が同程度出現するために，大局的には一定に保たれる．以上のように，本手法では，絶対的な音程を計算しているのではなく相対的な変化のみを検出している．タンギングは，入力音声ストリーム中に見られる短い発声部分を数えることで検出される．音声の有無，音程変化，タンギングの検出は，単純な実装ながら，複数のユーザに対して比較的頑健に動作することを確認している．

音声認識部分は，音素の識別に簡単なテンプレートマッチングを，音素列からの単語認識には直接コーディングされたヒューリスティックスを用いており，あくまでも簡単なプロトタイプとしての実装となっている．現状では，実装者の声に対して限定された数語を認識するのみである．本格的な音声認識エンジンと組み合わせて，より実際的な試験を行うことは今後の課題である．

6　議論

通常の音声認識による入力と比較した場合の本手法のメリットしていは，1) 発話してから結果を待つのなく，システムの出力を見ながら連続的に操作を行うことができること，2) 日本語や英語といった特定の言語に依存しないこと，3) 声だけでなく拍手や指のスナップといった身体的な音も同様に利用できること 4) 実装がシンプルであること，が挙げられる．第 4 の点については，大量の語彙を聞き分けなければならない通常の音声認識に比べて，低レベルの音響信号を見るだけの単純なアルゴリズムであるため，ノイズに強く，話者に依存しない動作が実現しやすいと期待できる．

本論文で提案した手法は，身体的な障害，あるいは特定の作業のために両手が使えない場合の計算機インタフェースとして利用することを想定している．具体的な例としては，自動車の運転中のカーナビゲーションシステムの操作 [8]，CAVE のような没入型のバーチャルリアリティシステムで両手がふさがっている場合の操作，あるいはロッククライミングや乗馬など両手がふさがっている状況での携帯端末の利用などが考えられる．また，これ以外のアプリケーションとして，エンターテイメント向けの利用が考えられる．例として，本手法を利用した簡単なゲームを実装してみた結果，非常に好評であった．個人的な利用だけでなく，劇場などでの観客参加型のエンターテイメントに応用することも検討している．

本手法の問題点としては，通常の会話と異なる不自然な発声が必要となる点が挙げられる．従来型の音声認識でも計算機に認識させるために多少不自然な発声が要求される場合があるが，本手法の場合は，音声を連続的に出しつづけなければならないなど，ユーザへの負担がさらに大きいものと考えられる．また，不自然な発声は，周囲

の人間にとっても迷惑となるので，共用のオフィスや公共の場での利用は困難であろう．

本手法は，従来型の音声認識に基づく入力と組み合わせることによって一層の効果をあげるものと期待される．例えば，３節で例にもあげたような，音声認識によってコマンド情報を入力し本手法によってパラメータを調節する，といった組み合わせが有効な使い方と考えられる．今後は，非言語情報を活用したインタフェースについて本論文で提案した３手法以外のものを考えていく他，各種のモダリティを組み合わせたインタフェースにいて探っていきたいと考えている．

謝辞

高速フーリエ変換部分に大浦拓哉氏による FFT Package を利用させていただいた．

http://momonga.t.u-tokyo.ac.jp/~ooura/index-j.html

参考文献

[1] 後藤真孝，伊藤克亘，秋葉友良，速水悟，音声補完:音声入力インタフェースへの新しいモダリティの導入，インタラクティブシステムとソフトウェア VIII，日本ソフトウェア科学会 WISS2000, pp.153-162，近代科学社, 2000.

[2] Hirose,Y., Ozeki,K., Takagi,K., Effectiveness of prosodic features in syntactic analysis of read Japanese sentences, Proceedings of ICSLP2000, Vol.3, pp.215-218, 2000.

[3] Igarashi,T., Hinckley,K. Speed-dependent automatic zooming for browsing large documents, Proceedings of UIST'00, pp.139-148, 2000.

[4] Iwano,K., Hirose,K., Prosodic Word Boundary Detection Using Statistical Modeling of Moraic Fundamental Frequency Contours and Its Use for Continuous Speech Recognition, Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol.1, pp.133-136, 1999.

[5] Lieske,C., Bos,J., Emele,M., Gamback,B., Rupp,CJ., Giving prosody a meaning, Eurospeech97 vol3 pp.1431-1434, 1997.

[6] Manaris,B., McCauley,R., MacGyvers,V., An Intelligent Interface for Keyboard and Mouse Control--Providing Full Access to PC Functionality via Speech, Proceedings of 14th International Florida AI Research Symposium (FLAIRS-01), 2001, (to appear).

[7] Tsukahara,W., Ward,N., Responding to Subtle, Fleeting Changes in the User's Internal State. Proceedings of CHI 2001, pp.77-84, 2001.

[8] Westphal,M., Waibel,A. Towards Spontaneous Speech Recognition For On-Board Car Navigation And Information Systems, Proceedings of the Eurospeech 99, 1999.