

手話 CAI のための手話記述言語と GUI の開発

三船 智明†, 江刺 宏恭†, 千種 康民†

†東京工科大学 工学部 情報工学科

最近、聴覚障害者と健常者とのコミュニケーション手段としての「手話」に対する注目度は社会的要求とマスメディアの影響により高められてきている。現状では、手話の学習には本やビデオを用いた手段が一般的である。本の場合、項目が構造化されているため文章の検索は容易であるが、手話において最も重要である手の動作がわかりづらい。また、ビデオによる学習の場合、実際の手の動作が観察できて効果的であるが、そのメディアの特性上、目的の手話の検索に時間がかかる。さらに、どちらのメディアの場合も、受動的な学習といった傾向が強いため、これらにはいくつかの例文が用意されているが、学習者の意図する表現にぴったりの文章を見つけるのは困難である。このように、現状では手話の学習には相当な努力が必要である。

そこで著者らは、手話の学習が能動的にかつ手軽にできる「手話の学習支援システム」の開発を行った。このシステムでは次の3点を要求仕様とし、実現した。1. ユーザは、単語のボタンを使用して、実現したい手話を簡単かつ自由に入力できる。2. 新たに開発した「手話記述言語」を3次元描画するインタプリタ上で、入力した文章の手話はアニメーションとして描画され、何度でもプレイバックできる。3. 手話のアニメーションを詳細に学習するため、観察する視点を、正面、斜め上、斜め後ろ、に変更できる。

Gesticulation CAI using Gesture Programming Language and GUI

Tomoaki MIFUNE, Hiroaki ESASHI and Yasutami CHIGUSA
Dept. of Information Engineering, Tokyo Engineering University
E-mail chigusa@cc.teu.ac.jp

It is very important to help handicapped persons and to develop many helpful tools for them. These requirements are much grown in our world. We focused in learning tools for gesticulation speaking. It is popular to learn gesticulation speaking with books or video tapes. In the case of using books, we can easily search something to learn, but we cannot always learn the motion of the gesture as any word. In another case, we can easily learn the motion of the gesture, but it takes much time for searching the gesture to learn.

To realize easily learning and searching, we developed a gesticulation CAI using gesture programming language and the GUI. There are three main point: 1. user-friendly interface, 2. playing gesture animation and playback, 3. view point approach for detail learning.

1 はじめに

人間の生活にコミュニケーションは不可欠である。人と人の日常的なコミュニケーションの主な手段として健常者は「音声によることば」を使用し、聴覚障害者は「手話」を使用する。この、聴覚障害者にとって日常的に使用する「ことば」である手話を必要とする社会的要求も次第に高まってきた。現状では手話の学習には本かビデオを用いた手法が一般的である。しかし本による学習は、本の内容が構造化されているため、検索は容易であるが、手話の手の動作の表現方法に限界がある。また、ビデオによる学習は手の動作が見えて効果的ではあるが、目的とするシーンの検索に時間がかかる。また、どちらの場合も、いくつかの例文が用意されてはいるが、学習者の学びたい文章に合致した手話の表現の検索は困難である。

そこで著者らは、図1に示すような、手話の学習を容易にする学習支援システムの開発を行なった。

この開発は以下の手順で行った。

1. 手話の表現の中から「単語」と「動作」の出現頻度を調べる。

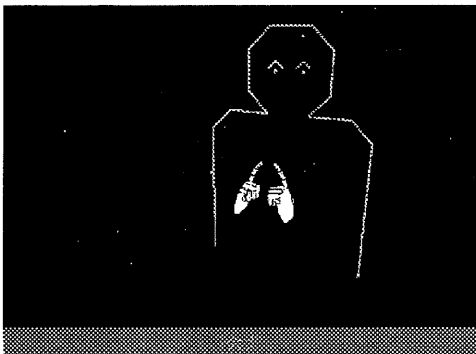


図1: 手話のアニメーションの一コマ

2. 単語と動作の関係を記述する手話記述言語の仕様を決定する。
3. 手話記述言語で記述されたプログラムをX11 Window上にアニメーションとして描画するインタープリタをで開発する。
4. ユーザの入力を容易にするため「単語」のボタン群を有するパネルをTCL/TKにて開発する。

ただし、対象とした手話の表現は文献[1]と文献[2]から抽出した。以上の手順に従って実現された機能は、

1. ユーザは単語を組み合わせて、実現したい手話を簡単かつ自由に入力できる。
2. 入力した文章の手話はアニメーションとして描画され、何度でもプレイバックできる。
3. 手話のアニメーションを詳細に学習するため、観察する視点を変更できる。

である。

2 手話記述言語

2.1 手話の分析と手話記述言語の基本動作の抽出

手話記述言語の仕様を定めるに当たって、単語と手話の対応を図解している文献[1]と、文章と手話の対応を図解している文献[2]を調査した。文献[1]の約500単語の分類について調べると、単語は、人間、政治・経済・法律、生産・労働、教育・福祉、文化・歴史、自然・自然科学、心理、動作・活動、疑問詞・時・場所・形・量、抽象、スポーツ、数詞、地名、の13分野に分類され

ている。文献 [2] の約 150 の文章の分類について調べると、あいさつ分野 1, 教育・文化, 社会福祉・ろうあ者問題, 職業, 医療・衛生, の 5 分野に分類されている。

これらの中から動作を抽出し、手話記述言語の仕様とするために、文章表現において主語、目的語として使用され、かなり頻度が高い文献 [1] の「人間」に関する単語 39 単語からその動作を抽出した。単語を手話するアクション数によって分類すると、ワンアクション (19), ツーアクション (15), スリーアクション (5), となる。() の数字は頻度である。これにより、39 単語は $19*1+15*2+5*3=64$ の動作数となる。これをさらに手の動作 (左, 右, 対称, 平行, それ以外) によって分類すると、左のみ (8) 右のみ (30) 対称 (14) 非対称 (12) となる。左右とも動作させる場合、左右別々に計数すると、全動作数は $(8+30)*1+(14+12)*2=90$ となる。さらに、動作の種類によって分類すると、平行移動/44, 静止 (23), 回転 (11) 指を曲げる (5), 脈動 (5), 微動 (2), の 6 動作となる。これを手話記述言語の基本動作とし、実際の文章 (文献 [2] のあいさつ) の 31 の文章 (動作数 227) に当てはめてみる。この結果、前述の 6 種類の動作に含まれない動作数は 5 となり、前述の動作を実現することにより、98% の動作が記述可能であると予測できる。

2.2 手話と日本語文章との相違

日本語の文章は、複数の単語を助詞により組合せ、構文を構成し、1つの文章をなしている。また、動詞、形容詞などは活用形を持つため文章の構成方法はさらに複雑になる。これに対し、手話の特徴を以下に示す。

1. 単語は活用形を持たない。

2. 基本的に単語と単語の間に助詞は不用である。
3. 未知語を指文字で表現できる。

日本語の文章を表現する際、手話の話者は単語を次々と手話としてジェスチャーすればよい。また、表現したい手話の単語を知らない場合は、指文字を使用して、単語の音をジェスチャーすればよい。指文字は、4 6 音のひらがなと濁音, 半濁音, 促音, 長音の手話を実現する。例えば、「みふね」が未知語である場合は、指文字「み」「ふ」「ね」を順に手話すれば良い。

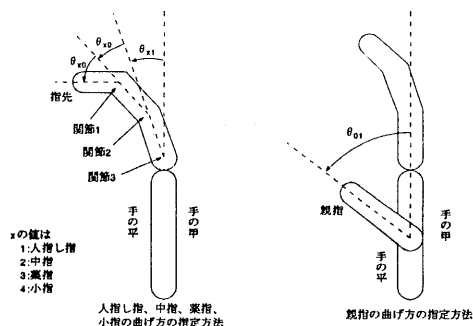
3 手話インタープリタ

手話インタープリタは以下の手順したが、い単語をアニメーション表示する。

1. ワールド座標系内に視点を決定する。
2. 手話の開始における両手の形状, 向き, 位置を与える。
3. 手話の終了における両手の形状, 向き, 位置を与える。
4. アニメーションの一コマにおける両手の形状, 向き, 位置を決定する。
 - モデリング座標系内の両手の形状の決定する。
 - モデリング座標系内の両手の向きを決定する。
 - ワールド座標系内の両手の 3 次元位置を決定する。
 - 手, 指の各部を視点からの距離により並び換える。
 - 視平面座標系に距離の遠い順に手指の各部を描画する。

3.1 手の座標系

両手の座標系はそれぞれ異なるモデリング座標系において定義される。話者の右手は、右手座標系、左手は、左手座標系を使用する。これを図2に示す。この状態を両手の初期状態とする。この形から指や手首を曲げる角度を指定することにより任意の手の形状を実現する。



3.2 手の形状

手首から先を定義を図3に示す。処理速度を考慮して、指は太さのある線（骨）、手の平は厚さのある1枚の平面、で構成する。指は各関節ごとに曲げる角度を指定することにより、自由に指を曲げ、開くことができる。これにより自由度の高い手の形状を実現する。

左右の手の指は各関節の角度 θ_{ab} を指定することにより、曲げたり開いたりすることができる。ただし、本研究に於いては手の形のデータの量を減らすために、図中の関節1と2を曲げる角度は同じ値を用いる。また、人指し指および小指を開く角度 θ_{12} と θ_{42} は、薬指を開く角度 θ_{32} に応じて自動的に決定される。

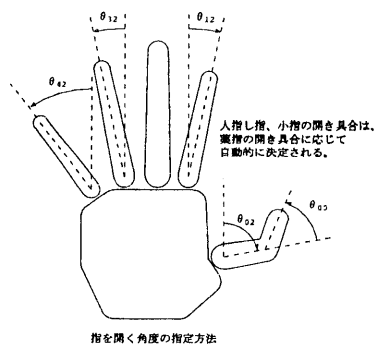


図3: 指の曲げ方, 開き方の指定方法

3.3 手の向き

指定した角度の指を曲げた手の形状を図2の座標軸まわりの回転を施すことにより、手の向きを任意に決定できる。実際には、手をモデリング座標系内で、x軸、y軸、z軸まわりに回転させることによって、手首を曲げることを実現している。回転の順番は、x軸まわり→y軸まわり→z軸まわりとする。ただし、右手用のモデリング座標系には右手系、左手用には左手系を用いているため、手首の回転方向は左右対称となる。これにより、左右対称の向きを決定する際、同じ回転角度を指定すれば良く、非常に都合がよい。

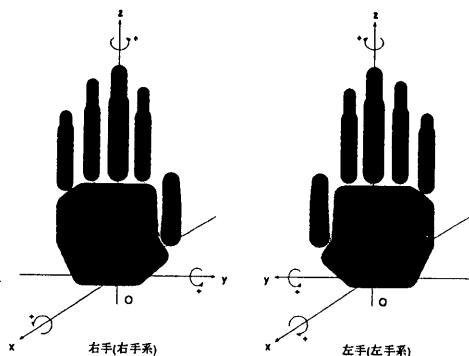


図2: 両手の座標系

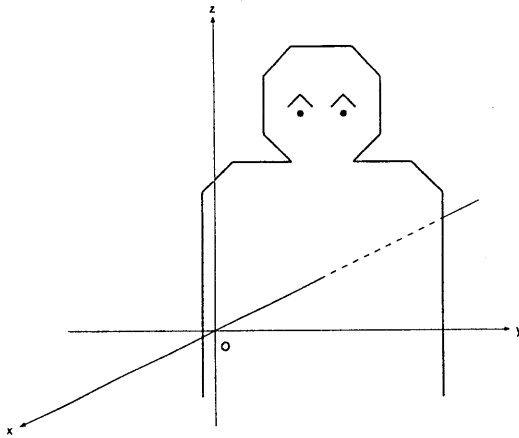


図 4: 話者の座標系

3.4 話者の座標系と手の位置

指を曲げ向きを変えた手のモデルに平行移動を施し、3次元の世界座標に変換する。世界座標系内に設定した人間の体の前に配置する。

世界座標系内に人間の形を定義しておき、それに合わせて左右それぞれの手の位置を世界座標系の値で指定することにより、手の位置が決定される。なお、世界座標系の原点が、おおむね人間の胸の中央のやや前方になるように、人間の位置を調整してある。これを図4に示す。

3.5 隠線処理

本研究で用いる隠線処理は、描画しようとする全ての物体の奥行き距離を求め、物体を奥行き距離の大きい順に並び換え、視点から遠い順に物体を描画するというものである。距離は視点と物体の距離で、直線の場合は中点との距離、手の平の場合はその重心との距離である。このアルゴリズムの場合指と指を複雑に絡み合わせる場合に正しく描画されないが、本研究においては、

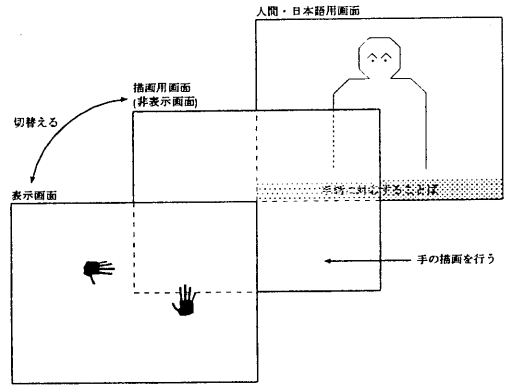


図 5: 画面構成

手指の表示のためだけに用いており、複雑な図の表示は行わないので、実用上特に問題は無いと思われる。

3.6 視平面座標系への描画

視平面（投影面）は世界座標系内に任意に設定でき、視点から視平面を通して人間と手を見た時の視界を、透視変換により視平面座標系に変換し、奥行き情報を利用して隠線処理を行い、アニメーションの1コマとして描画する。これを図5に示す。

3.7 手話のアニメーション

両手の始めの状態（手の形、向き、位置）と終りの状態を指定すると、始めの状態から終りの状態に向かう途中の状態を直線補間により定める。つまり、画面上でスムーズなアニメーションとなるように、指を曲げる角度、手の向き、手の位置および隠線処理、座標変換のそれぞれについて、リアルタイムに計算し、途中の状態を次々と描画し、手話のアニメーションを実現する。

4 システムの構成

4.1 Tcl/Tk を用いた日本語入力 インタフェース

Tcl(Tool Command Language) は言語として2つの側面を持ち、1つは、単独で動作し、簡単に実行結果を得ることができる独立した言語としての側面であり、もう1つは、他のアプリケーション用の組み込みマクロ言語としての側面である。前者はBASICのようなインタプリタ言語としての側面で、この側面から見たTclはテキスト処理言語としての性格を持つ。もう一つの側面は組み込みマクロ言語としての特徴を持っている点である。TkはTclにGUIインタフェースを提供するTclの拡張である。Tcl/Tkを用いるとGUIの構築を容易に行うことができる。

実際にTcl/Tkを用いて、例題、単語、指文字(50音)が書かれたボタンのパネル作成した。これを図6に示す。実際に用意したボタンは、例題：3文、名詞：4、動詞：6、副詞など：13、指文字：46、である。マウスでボタンをクリックすることにより文章を作成する。「正面から見ます」「斜めから見ます」「斜め後ろから見ます」と書かれたボタンのいずれかをクリックすると、対応する視点から見た時の手話がアニメーション表示される。

4.2 手話インタプリタ

手話インタプリタでは、手話CAIとして、手話の学習に役立つように学習者の視点を正面、斜め前、斜め後ろ、と変更できる。これにより、学習者が与えた手話の時空間のイメージを理解するのに大きく貢献する。

Tcl/Tkを用いて作成した日本語入力インタフェースに文章を入力し、手話インタ



図6: Tcl/Tkを用いて作ったボタン

リタのアニメーション表示を実行中の画面を図10に示す。画面右側は日本語入力インタフェースのパネルであり、左下は入力された文章の表示画面である。

5 おわりに

東京工科大学の1995年10月の学園祭で、本研究の実演展示を行った。日本語を手話に翻訳するというテーマの珍しさに加え、マウスの操作のみで行える簡単な入力インタフェースとグラフィカルな画面構成のため、大変な注目を集めた。3次元の特性を生かして視点を変更できる点は大きく評価された。特に斜め後ろからの視点、すなわち「手話をしている自分の手を自分で見る視点」は、手話CAIとして大いに役立

ることがある。スピードを制御するパラメータが必要である。

・手話の表現力

実際の手話では、表情や体の動きが意思の伝達において重要な要素となる。そのため、少なくとも手話に応じた表情をつけられると、よりいっそう表現力が高まる。

参考文献

- [1] 手話研究委員会編, "わたしたちの手話(1)", 全日本ろうあ連盟, 1995.
- [2] 手話研究委員会編, "わたしたちの手話(会話編1)", 全日本ろうあ連盟, 1994.
- [3] 石井繁夫, "続C言語による3Dグラフィックス入門", 技術評論社, 1994.
- [4] 井門俊治, "X-Window 実用グラフィックス入門", 日刊工業新聞社, 1992.
- [5] 木下凌一, "X-Window Ver.11 プログラミング", 日刊工業新聞社, 1990.

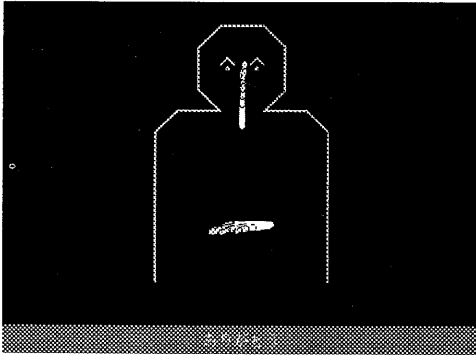


図 7: 正面から見た図

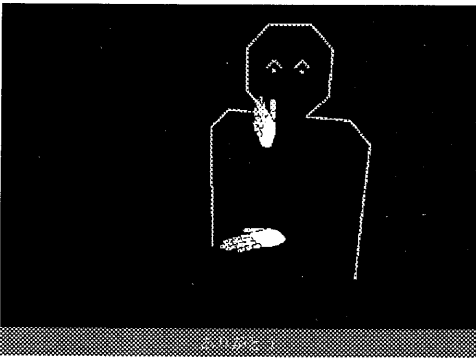


図 8: 斜め前から見た図

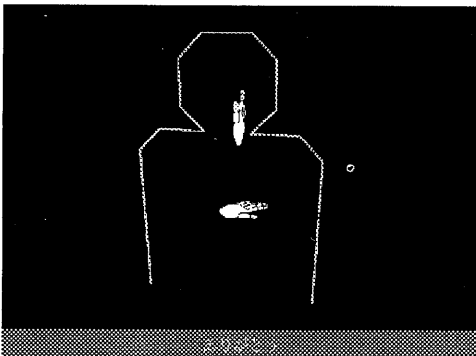


図 9: 斜め後ろから見た図