モバイルピクトグラミングへの音声による自然言語文入力機能の実装

石井幹大[†1]　　伊藤一成[†2]

ピクトグラムとは，日本語で絵記号，図記号と呼ばれるグラフィックシンボルであり，意味するものの形状を使ってその意味概念を理解させる記号である．我々の研究グループでは，人型ピクトグラムを用いたコンテンツ作成環境「ピクトグラミング」を開発しており，人型ピクトグラムの形状のシンプルさや，命令セットの単純さを生かして，そのスマートフォン版である「モバイルピクトグラミング」も開発・公開している．しかし，現状では，「モバイルピクトグラミング」では，ソフトウェアキーボードによる入力のみをサポートしており，入力インタフェースの改良が喫緊の課題であった．そこで，スマートフォンに標準でサポートされている音声入力機能を利用した入力方式を構築したので，報告する．

1. はじめに

ピクトグラムとは，日本語で絵記号，図記号と呼ばれるグラフィックシンボルであり，意味するものの形状を使って，その意味概念を理解させる記号である[1]．ピクトグラムは，世界共通の記号表現として，世界中で用いられており，グローバル化やその流れに伴う外国人観光客客の増加などの理由で，一例を挙げるだけでも，感性工学[2]，異文化コミュニケーション[3]，メディア情報処理[4]，記号論[5]など，様々な学問領域で研究が盛んになっている．

ピクトグラムの共通化は，ISO（国際標準化機構）を中心に規格の審議や策定が進められており，中でも，ISO 3864 の付録では，人間のピクトグラムに特化した作成ガイドラインが提示されている．以後，本稿では，このピクトグラムを人型ピクトグラムと呼称する．

我々の研究グループでは，人型ピクトグラムを用いたコンテンツ作成環境「ピクトグラミング[a]」を開発している．「ピクトグラミング」は，ピクトグラムとプログラミングを組み合わせた造語であり，初学者向けプログラミング学習環境の用途でも用いられている[6]．「ピクトグラミング」は Web アプリケーションであり，PC での利用を想定した開発に重きを置いてきたが，人型ピクトグラム自体の形状のシンプルさや，命令セットの単純さを生かして，スマートフォン上で動作する「ピクトグラミング」も開発・公開し，「モバイルピクトグラミング[b]」と呼称している[7]．しかし，「モバイルピクトグラミング」は，PC のブラウザ上で動作する版と同じく，キーボード入力による命令の記述を旨としており，スマートフォンの特徴を活かしたアプリケーションの開発が急務であった．

そこで，スマートフォンに標準でサポートされている，音声入力機能に着目し，音声の入力によりピクトグラムのコンテンツを作成が可能な環境を構築したので報告する．

以下，2 章で音声入力インタフェースについての関連研究を示し，3 章でピクトグラミングおよびモバイルピクトグラミングについて解説をする．4 章で実装した内容を解説し，5 章で本稿をまとめる．

2. 関連研究

音声操作インタフェースに関する研究は，多く行われている．非言語情報を利用した例として，音声の持つ音響信号に注目した研究が挙げられる[8]．ユーザの発声量によって，画面上のキャラクターの移動量を調整したり，画面のスクロール量を調整したりすることで，エンターテイメント分野で有効性を示したが，通常の会話と異なる不自然な発声が必要であり，ユーザに負担がかかる点が課題であった．一方，言語情報を利用した研究として，医療でのトリアージにおける活用[9]や，テレビ操作における活用[10]，PC におけるキーボードやマウス操作への活用[11]などが挙げられる．音声操作インタフェースを用いる利点として，複雑な操作を簡便にすること[10]や，運動障害者に対する操作の支援[11]，ハンズフリーで操作が可能になること[9]が示されている．しかし，これらの研究では，発話された音声を認識した文字列がそのまま操作に用いられていることが多く，音声認識率の低さを問題に感じるユーザが多数存在する[12]．また，認識率が低いにも関わらず，発話が終了するまで認識結果が表示されない，他の変換候補が表示されない，といったことが原因で，誤り訂正作業に多くの時間がかかることも課題であった[13]．

音声認識率の低さを改善するための研究には，ユーザの対話記録から作成したコーパスを利用し，使用頻度が高い指示代名詞「それ」への対応を実装した，音声対話機能付き家電品の開発と評価[12]や，情報が断片的に発話された場合でも，発話された単語の断片の残りをシステム側が補う「音声補完」という音声入力インタフェースの検討[14]や評価[15]などがある．また，システム側が認識誤りを起こした場合でも，ユーザが効率的にその誤りを訂正可能な「音

†1 青山学院大学大学院社会情報学研究科
†2 青山学院大学社会情報学部

a https://www.pictogramming.org/editor/
b https://www.pictogramming.org/editor/sp.html

ⓒ2019 Information Processing Society of Japan

声訂正」という音声入力インタフェースの検討[16]や評価[17]も行われている．いずれも，ユーザの満足度は高く，使いやすいシステムである，文章入力が容易である，今後も使いたい，といった結果が示されており，音声認識率の低さを改善する取り組みは有効であると言える．

しかし，これらの研究では，認識誤りを改善するために，形態素解析ソフト「茶筌[18]」を利用したり[12]，音声の持つ基本周波数を測定したり[14][15]，単語を音響的にクラスリングする「Confusion Network[19]」を利用したり[16][17]しており，簡便に実装が行えるとは言い難い．

本提案では，モバイルピクトグラミングで用意されている命令セットが単純であるという利点を活かし，コンテンツを作成できる環境を実装する．ユーザは，スマートフォンに標準でサポートされている音声入力機能を利用するため，日常的になれた操作で扱える．音声入力された語句の変換は，入力語句の使用履歴に基づいて行われることから，ユーザが繰り返し本実装を使用することで，音声入力された語句の誤変換が少なくなる効果が見込めると考えられる．

3. ピクトグラミング

3.1 概要

PC のブラウザでピクトグラミングの Web アプリケーションにアクセスした場合のスクリーンショットを，図 1 に示す．図 1 において，領域(a)は実行結果を出力する「人型ピクトグラム表示パネル」である．初期状態では，パネル全体を占有するほどの大きさの人型ピクトグラムが表示される．領域(b)はプログラムを入力する「プログラムコード記述領域」である．この領域に命令を入力することで，人型ピクトグラム表示パネルに表示される人型ピクトグラムの動作や状態を定義する．領域(c)は，「実行ボタン」であり，プログラムコード記述領域に記述した命令を実行し，人型ピクトグラム表示パネルに表示されている人型ピクトグラムを動かすことができる．なお，命令の実行は，カーソルがプログラムコード記述領域に位置する際に，エンターキーを押下することでも，行うことができる．領域(d)は「命令入力支援ボタン」である．このボタンを押下するとボタンのラベルに記述されているプログラム文字列が，プログラムコード記述領域上のカーソル位置に挿入される．

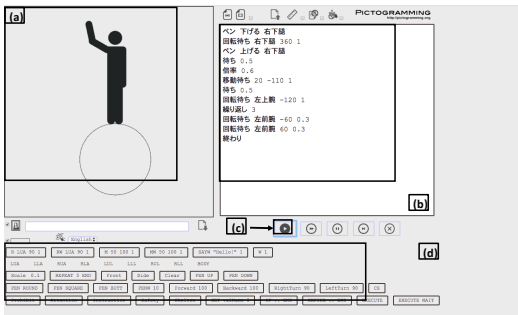

図 1　ピクトグラミングのスクリーンショット

人型ピクトグラム表示パネルに表示される人型ピクトグラムは，ISO 3864 で定義されているサイズ比に準拠している．ピクトグラミングでは，身体の部位名あるいは関節名を指定して，人型ピクトグラムの動作や状態を決定する．図 2 に，部位の名称一覧（上部）と関節の名称一覧（下部）を，それぞれ示す．

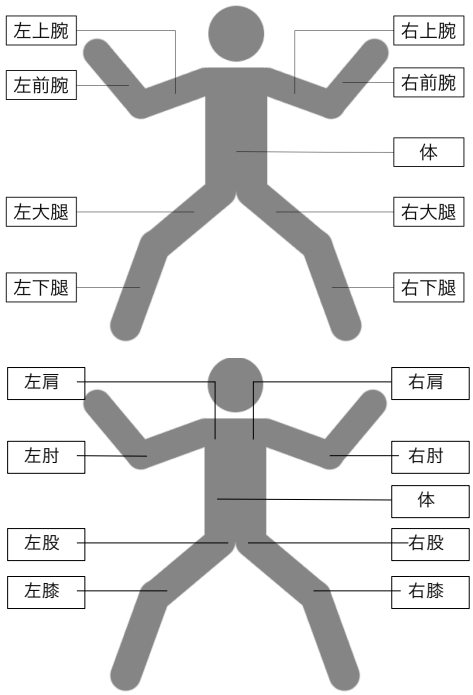

図 2　部位に基づく名称（上側）と
関節に基づく名称（下側）

プログラムコード記述領域へ入力する動作や状態を変化させるための命令の書式は，先頭に命令コードを記述し，空白文字を挟んで，続けて引数列を空白文字で区切って列挙する．図 3 に命令の書式を示す．

命令コード 引数 1 引数 2 …

図 3　命令の書式

続けて，図 4 にプログラム例を示す．

```
01  ペン 持つ 右膝
02  回転待ち 右膝 360 1
03  ペン 離す 右膝
04  待ち 0.5
05  倍率 0.6
06  移動待ち 20 -110 1
07  待ち 0.5
08  回転待ち 左肩 -120 1
09  繰り返し 3
10  回転待ち 左肘 -60 0.5
11  回転待ち 左肘 60 0.5
12  終わり
```

図 4　プログラム例

ⓒ2019 Information Processing Society of Japan

図 4 のプログラムは,「人型ピクトグラムが，自身の右膝で描いた円の上に乗り，左腕を上げ，左前腕を 3 回振る」コンテンツを作成するためのものであり，図 1 の人型ピクトグラム表示パネルの表示は，図 4 のプログラムを実行した際のワンショットである．

3.2 モバイルピクトグラミング

iOS 端末上のブラウザでモバイルピクトグラミング[7]にアクセスした場合のスクリーンショットを，図 5 に示す．ピクトグラミングは, HTML5, CSS, JavaScript を用いた Web アプリケーションなので，スマートフォンでの表示にも容易に対応できる．ただし，表示領域やインタフェースの違いを考慮しなければならないため，3.1 節 図 1 で示した，(a) : 人型ピクトグラム表示パネル，(b) : プログラムコード記述領域，(c) : 実行ボタン，(d) : 命令入力支援ボタン は縦に並ぶレイアウトとなっている（図 5 右部）．そのため，ピクトグラミングとは違い，プログラム記述領域に命令を入力しながら，人型ピクトグラム表示パネルの人型ピクトグラムの動作を確認することはできない．

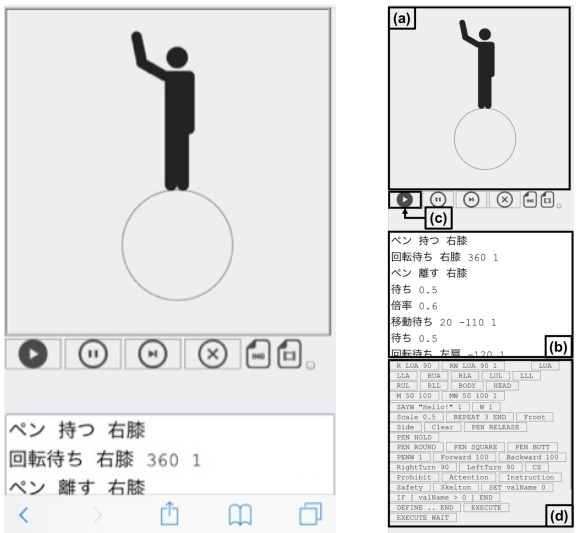

図 5 モバイルピクトグラミングのスクリーンショット

4. モバイルピクトグラミングへの音声入力機能の構築

4.1 実装方式

音声の入力には，スマートフォンに標準でサポートされている音声入力機能を利用する．図 6 に，iOS 端末でのスクリーンショット（左部）と Android OS 端末でのスクリーンショット（右部）を，それぞれ示す．図 6 において，(a)は人型ピクトグラム表示パネル，(b)はプログラムコード記述領域，(b)’は音声入力領域，(c)は実行ボタン，(d)は命令入力支援ボタンである．画面の基本的な設計は，3.2 節 図 5 で示したモバイルピクトグラミングと同様であるが，図 6 の「音声入力領域」は，本実装に特化した画面設計である．

音声入力領域をタップすることで，音声による命令入力が可能となる．本実装にあたり，音声入力領域を用いた理由は，テキスト入力上（プログラムコード記述領域）で，音声入力によって生じた誤認識や誤変換の修正を行ってしまうと，修正後に音声入力ができなくなってしまうからである．そのため，音声入力によるテキストを，そのままの状態で格納するだけの領域（音声入力領域）を作成し，対応した．処理手順については，4.3 節で記す．

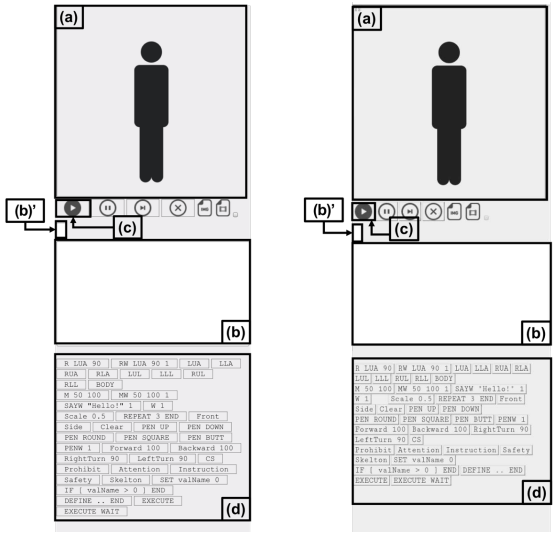

図 6 音声による命令入力機能を追加した
モバイルピクトグラミングのスクリーンショット

4.2 音声入力可能な命令一覧

音声入力可能な命令を表 1，表 2 に示す．

4.3 自然な発話文による入力

本提案では, 自然な発話文による命令入力を可能とする．自然な文を生成するために発話する単位を表 3 に，自然な発話文による命令入力例を表 4 に，それぞれ示す．

モバイルピクトグラミングの命令入力では，「回転 左肩 60 1」のように，単位を入力する必要はないが，そのような発話では, それぞれの数字が指し示すものが不明瞭であり，自然な発話とは言い難い．そこで，自然な発話を実現するために，表 3 に示す単位を発話することとした．しかし，単位を発話することによる最大の利点は別にあり，連続した数値の区切りの判断が容易に可能である点である. 4.2 節表 1 を参照すると，モバイルピクトグラミングでは,「回転待ち 引数 1 引数 2 [引数 3]」や「移動 引数 1 引数 2 [引数 3]」のように，引数に連続して数値が入る命令があるが，例えば，単位の発話がない場合,「180 1」を入力するために,「ひゃくはちじゅう いち」と発話しても, 通常では「181」と単一の数値を発話したと解釈され，変換されてしまう．そこで，数値の後に単位を付与して発話することで，連続した数値の区切りを明確化することができる．

ⓒ2019 Information Processing Society of Japan

表 1　音声で入力可能な命令一覧

命令の様式	処理
回転 引数 1 引数 2 [引数 3]	引数 1 で指定される身体の部位あるいは関節を，反時計回りに引数 2 度だけ引数 3 秒かけて，支点を中心に等速回転する．引数 3 が省略されたときは「0」が入力されているとして取り扱う．
回転待ち 引数 1 引数 2 [引数 3]	引数 1 で指定される身体の部位あるいは関節を，反時計回りに引数 2 度だけ引数 3 秒かけて，支点を中心に等速回転する．回転が終了するまで，次の命令は実行されない．引数 3 が省略されたときは「0」が入力されているとして取り扱う．
移動 引数 1 引数 2 [引数 3]	引数 3 秒かけて x 軸正方向に引数 1 ピクセル，y 軸正方向に引数 2 ピクセルだけ全体を等速直線移動する．引数 3 が省略されたときは「0」が入力されているとして取り扱う．
移動待ち 引数 1 引数 2 [引数 3]	引数 3 秒かけて x 軸正方向に引数 1 ピクセル，y 軸正方向に引数 2 ピクセルだけ全体を等速直線移動する．直線移動が終了するまで，次の命令は実行されない．引数 3 が省略されたときは「0」が入力されているとして取り扱う．
ペン 引数 1 [引数 2]	引数 1 が「持つ」の場合，ペンを持つ（描く）．「離す」の場合，ペンを離す（描かない）．初期状態は，ペンを持っている状態である．ペンの動作を行う身体の部位の名称を，「回転」，「回転待ち」命令と同様の表記で引数 2 に指定できる．引数 2 が省略された場合は，「体」が記述されているとして取り扱う．
透明	人型ピクトグラムを透明モードにする．透明モードで再度命令すると，通常の人型ピクトグラムに変更する．
繰り返し 引数 1	対応する「終わり」までの命令を，引数 1 回繰り返す．
終わり	「繰り返し」の終了を示す．
倍率 引数 1	人型ピクトグラムの拡大率を引数 1 倍にする．初期状態は，1 倍である．
待ち 引数 1	引数 1 秒何もせずに待つ．待ちが終了するまで次の命令は実行されない．

表 2　音声入力機能でサポートするメタ命令の一覧

メタ命令の様式	処理
実行	プログラムコード記述領域に入力したプログラムを実行する．

表 3　音声入力機能で使用する数値の単位一覧

引数が表現するもの	発話する単位
角度	度
時間	秒
x 軸正方向への移動量	横
y 軸正方向への移動量	縦
回数	回
倍率	倍

表 4　自然な発話文による命令入力例

発話文の例		変換後の命令（メタ命令を含む）
[**引数 3 秒で**] **引数 1 を引数 2 度，回転する．**	**引数 1 を[引数 3 秒で]引数 2 度，回転する．**	回転 引数 1 引数 2 [引数 3]
[**引数 3 秒で**] **引数 1 を引数 2 度，回転して待つ．**	**引数 1 を[引数 3 秒で]引数 2 度，回転して待つ．**	回転待ち 引数 1 引数 2 [引数 3]
[**引数 3 秒で**]**横へ引数 1，縦へ引数 2 移動する．**	[**引数 3 秒で**]**縦へ引数 2，横へ引数 1 移動する．**	移動 引数 1 引数 2 [引数 3]
[**引数 3 秒で**]**横へ引数 1，縦へ引数 2，移動して待つ．**	[**引数 3 秒で**] **縦へ引数 2，横へ 引数 1 移動して待つ．**	移動待ち 引数 1 引数 2 [引数 3]
[**引数 2 に**]**ペンを引数 1 (持つ)．**	[**引数 2 の**]**ペンを引数 1 (離す)．**	ペン 引数 1 [引数 2]
透明にする．		透明
引数 1 回，繰り返す．		繰り返し 引数 1
終わりにする．		終わり
倍率を引数 1 倍にする．	**引数 1 倍の倍率にする．**	倍率 引数 1
引数 1 秒待つ．		待ち 引数 1
実行する．		実行

なお，表 3 に示す単位の発話は，あくまで数値の区切りを明確にするためのものであり，プログラムコード記述領域には入力されない．

表 4 の「発話文の例」で太字になっている部分は，必ず発話をしなければならない．命令の区切りを明確にするため，一部の命令では，「する」を末尾に発話しなければならない仕様とした．また，自然な発話にするため，「待ち」の代わりに「待つ」，「繰り返し」の代わりに「繰り返す」と発話する語句として採用した．さらに，「を」や「で」といった助詞の挿入や，引数の順序を入れ替えた発話を許容している．

ⓒ2019 Information Processing Society of Japan

続いて，図 4 に示したプログラム例を，音声で入力する際の自然な発話文の例を，図 7 に示す．

```
右膝にペンを持つ.
右膝を 360 度，1 秒で回転して待つ.
右膝のペンを離す。
0.5 秒待つ.
倍率を 0.6 倍にする。
1 秒で縦に 20，横に-110 移動して待つ.
0.5 待つ.
左肩を 1 秒で-120 度回転して待つ.
3 回繰り返す.
0.5 秒で左肘を-60 度回転して待つ.
0.5 秒で左肘を 60 度回転して待つ.
終わりにする.
実行する.
```

図 7　音声入力の際の自然な発話文の例

4.4　変換テーブルの作成

音声入力には誤認識や誤変換が伴う．この修正には，用意された変換テーブルを用いる．変換テーブルは，筆者の発話によって収集された誤認識や誤変換の語句をもとに，作成した．表 5，表 6，表 7，表 8 に作成した変換テーブルを示す．

表 5　変換テーブル（命令，身体）

種別	語句	誤変換の候補
命令	回転	はい、/閉店/会見/外で
	移動	自動/伊藤/イトウ
	持つ	目/小木/重来/僕/重く
	離す	話す/話し/花/離風
	透明	東名/タオル名/トウメイ
	繰り返し	繰り返す
	終わり	終り/お会い
	倍率	バイりく/大菊/梅肉/バイク
	待ち	待つ/マック/幕/巻く
	ペン	店/辺/片
	実行	亀甲/執行/結構
身体	左	日大/平木/平井/平戯/平着
	右	握/日木/美里
	上腕	城王/冗談/上段/強腕/今日は
	前腕	電王/電話/前王/言腕/鍵盤
	大腿	大体/大大/太太
	下腿	難い/かたい/固い/硬い
	肩	方/こと
	肘	日日/富士/ひじ
	股	また/今田/俣
	膝	比嘉/光/日が/陽日/ギガ
	体	カラーラ

表 5 は，メタ命令を含む命令や，身体の部位，関節に関する変換テーブルである．なお，「種類」は，4.5 節での説明に用いるための列であり，実装した変換テーブルには記されていない．

表 6 は，数字に関する変換テーブルである．数値を音声で入力した際，漢数字に変換されてしまうことがある．そのため，「誤変換の候補」に漢数字も追加することで，漢数字を算用数字に変換し，対応する．

表 7 は，表 3 に示した単位に関する語句の変換テーブルである．4.3 節で述べたように，単位を発話する目的は，連続した数値の区切りを判断するためであるので，引数に連続して数値が入らない，「繰り返し 引数 1」の引数の単位として発話される「回」や，「倍率 引数 1」の引数の単位として発話される「倍」といった語句の変換テーブルは作成しなかった．

表 8 は，一部の命令において終端を示す「する」の変換テーブルである．

表 6 変換テーブル（数字）

語句	誤変換の候補	語句	誤変換の候補
1	一	8	八
2	二	9	九
3	三/間	10	十
4	四	100	百
5	五	1000	千
6	六	−	マイナス
7	七		

表 7　変換テーブル（単位）

語句	誤変換の候補	語句	誤変換の候補
度	事/°/摂氏温度	横	ヨコ
秒	票	縦	館

表 8　変換テーブル（命令の終端）

語句	誤変換の候補
する	送る/来る/降る/付で/振る/つける

4.5 音声認識により出力される文字列の変換手順

音声認識により出力される文字列に対する誤認識・誤変換の修正を行う処理手順の概要を，［手順 1］から［手順 5］に示す．図 8 (b)’ : 音声入力領域 の文字列の修正は，「実行する」が発話されるまで，命令ごとに繰り返し行う．以後の説明のため，音声入力領域の文字列における，修正対象の命令の文字列は，*txt* と定義する．図 8 に，「移動 -10 20 1」と「実行」が音声入力された際の処理の流れを示す．図 8 中の *startPoint* と *finishPoint* は，*txt* に格納する文字列を識別するための識別子として用いている．

ⓒ2019 Information Processing Society of Japan

[手順 1] 4.2 節で述べたように，「する」，「待つ」，「持つ」，「離す」，「繰り返す」のいずれかが，1 つの命令の終端文字列となる．すなわち，音声入力領域の文字列が，「する」，「待つ」，「持つ」，「離す」，「繰り返す」もしくは，表 5，表 8 で示したそれらの語句の「誤変換候補」の語句を新たに含んだ場合，1 つの命令が終了したとみなすことができる．そのため，その時点で音声入力領域に格納されている文字列に対して，誤変換や誤認識の修正を行う．

[手順 2] 表 6 を用いて，*txt* に格納されている漢数字を算用数字に置換する．

[手順 3] 表 5 を用いて，*txt* に格納されている，命令や身体に関する語句の置換を行う．その上で，表 5 中で「種類」が「命令」である語句を用いて，*txt* が表 4 のいずれの命令様式に該当するかを判別する．

[手順 4] 配列 *num* を作成し，*txt* に格納されている算用数字を先頭から順に格納する．その上で，*txt* のどの場所に，表 7 の「語句」または「誤変換の候補」が格納されているかを特定し，*num* に格納された算用数字が，表 4 の命令様式のいずれの引数に該当するかを判別する．

[手順 5] *txt* を表 4 に示すピクトグラミングの命令形式に変換する．*txt* に「実行」が格納されている場合は，プログラムコード記述領域のプログラムを実行し，それ以外の場合は，プログラムコード記述領域に命令文字列を追加する．

図 8　処理手順の概要

5. まとめと今後の展望

本稿では，人型ピクトグラミングを用いたコンテンツ作成環境「ピクトグラミング」のスマートフォン版である「モバイルピクトグラミング」の入力インタフェースを改良し，命令の音声入力機能を実装した．音声による命令入力では，自然な発話文による入力を可能とし，通常の会話のような自然対話の中で，人型ピクトグラムを用いたコンテンツが作成可能な環境を構築した．

今後は，中学生や高校生を対象に実験を行い，モバイルピクトグラミングを用いた人型ピクトグラムのコンテンツ作成における，本提案の有効性を検証する予定である．

参考文献

1) 太田幸夫：国際安全標識ピクトグラムデザインの研究 http://www.tamabi.ac.jp/soumu/gai/hojo/seika/2003/kyoudou-ota1.pdf (2019/02/11 閲覧).

2) 大野森太郎，上西綺香，原田利宣. (2015). 色彩表現を用いたピクトグラムにおける視覚言語の抽出とその検証．日本感性工学会論文誌, 14(3), 391-400.

3) Mori, Y., Takasaki, T., Ishida, T. (2009, February). Patterns in pictogram communication. In Proceedings of the 2009 international workshop on Intercultural collaboration (pp. 277-280). ACM.

4) 上西くるみ，青木輝勝. (2017). ピクトグラムマッチングのための輪郭情報を取り入れた局所形状記述子．第 205 回コンピュータビジョンとイメージメディア研究発表会 (CVIM).

5) Hassan, E. M. M. (2017). The Semiotics of Pictogram in the Signage Systems. International Design Journal, 5.

6) 伊藤一成. (2018). ピクトグラミング—人型ピクトグラムを用いたプログラミング学習環境．情報処理学会論文誌教育とコンピュータ (TCE), 4(2), 47-61.

7) Ito, K. (2019). Mobile Pictogramming. IMCOM(International Conference on Ubiquitous Information Management and Communication) 2019

8) 五十嵐健夫, Hughes, J. F. (2004). 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．情報処理学会研究報告ヒューマンコンピュータインタラクション (HCI), 2004(74 (2004-HI-109)), 47-51.

9) 萩野実咲，高橋祐樹，安藤禎晃，岡田謙一. (2013). 音声入力を用いた電子トリアージ用情報伝達システム．研究報告グループウェアとネットワークサービス (GN), 2013(8), 1-8.

10) 後藤淳. テレビ操作のための音声対話インタフェースの試作．言処学年次大, 455-458.

11) Manaris, B., McCauley, R., MacGyvers, V. (2001). An intelligent interface for keyboard and mouse control. In Proc. 14th Int'l Florida AI Research Symposium (FLAIRS-01) (pp. 182-188).

12) 渡辺裕太，関口芳廣，鈴木良弥. (2003). ビデオ装置を例とした家電品の音声対話機能について．情報処理学会論文誌, 44(11), 2690-2698.

13) Karat, C. M., Halverson, C., Horn, D., Karat, J. (1999, May). Patterns of entry and correction in large vocabulary continuous speech recognition systems. In Proceedings of the SIGCHI conference on Human Factors in Computing Systems (pp. 568-575). ACM.

14) 後藤真孝，伊藤克亘，速水悟. (2000). 音声補完:“TAB” on Speech. 情報処理学会研究報告音声言語情報処理 (SLP), 2000(64 (2000-SLP-032)), 81-86.

15) 後藤真孝，伊藤克亘，秋葉友良，速水悟. (2002). 音声補完: 音

ⓒ2019 Information Processing Society of Japan

声入力インタフェースへの新しいモダリティの導入．コンピュータ ソフトウェア, 19(4), 254-265.

16) 緒方淳, 後藤真孝. (2004). 音声訂正: 認識誤りを選択操作だけで訂正ができる新たな音声入力インタフェース．日本ソフトウェア科学会第 12 回インタラクティブシステムとソフトウェアに関するワークショップ (WISS 2004) 論文集, 47-52.

17) 緒方淳, 後藤真孝. (2007). 音声訂正: 選択操作による効率的な誤り訂正が可能な音声入力インタフェース．情報処理学会論文誌, 48(1), 375-385.

18) 松本裕治, 北内啓, 山下達雄, 平野善隆, 松田寛, 高岡一馬, 浅原正幸. (2000). 形態素解析システム 「茶筌」. 情報処理, 41(11), 1208-1214.

19) Mangu, L., Brill, E., Stolcke, A. (2000). Finding consensus in speech recognition: word error minimization and other applications of confusion networks. Computer Speech & Language, 14(4), 373-400.

ⓒ2019 Information Processing Society of Japan