

コマンドリズムを用いたタップ入力による携帯端末操作手法

石山 英貴^{1,a)} 高橋 伸² 田中 二郎²

概要: 携帯端末を操作するコマンドを一定規則に基づいてリズムに変換し、タップで入力することで操作を行う手法を提案する。リズムはコマンドの発音情報を基に生成され、起動したいアプリケーション名やその操作などのコマンドを思い浮かべることで入力リズムを連想することができる。コマンドをリズムに変換する際にリズムの重複が発生し入力が一意に定まらないことがあるため、日本語の言語特徴を入力に利用することで重複の減少を検討した。本稿ではコマンドのリズム変換方式と重複に対する考察、およびリズムタップ入力の認識について詳しく述べる。

An Interaction Technique with Tap Input using Rhythm Converted from Command

ISHIYAMA EIKI^{1,a)} TAKAHASHI SHIN² TANAKA JIRO²

Abstract: We propose an input method to operate commands on mobile device using rhythms performed by tapping. In our system, rhythm is directly generated from command's pronunciation by using a fixed set of rules, so users could associate the rhythm with the command by recalling the name of the operation or application that they want to invoke. Because there could be duplication when converting from the command pronunciation to the rhythm, there would be some cases in that the input could not be uniquely determined. Dealing with this problem, we have investigated reducing duplication by using the characteristics of the Japanese language. In this paper, we will describe in detail the tap rhythm input recognition, and also, the command rhythm converting rule with considerations about duplication.

1. はじめに

近年では携帯端末は高機能化しており、音楽プレイヤーや Web アプリケーション利用など様々な操作が可能になっている。操作が増加するにつれ、複数の機能を並行して利用したい場面が多くなる。例えばメモ帳やブラウザなどを操作しているときに後ろで流している音楽を変更したり、動画を閲覧しているときに画面の明るさを変更したりなどの操作を行いたい場合などが挙げられる。しかし、例えば、音楽プレイヤーで音楽の変更操作を行うためには、メモ帳を終了して音楽プレイヤーを起動しなければならないと

いったように、複数機能の利用にはいくつもの手順が必要となる。これは切り替えが何度も発生する場合や、ちょっとした操作を行いたい場合などに手間となる。この手間を減らすためには、操作しているアプリケーションを終了せずに、他のアプリケーションの操作が行えるようにすれば良い。上で挙げた例でいうと、メモ帳を起動したまま任意の歌手名をコマンドとして直接入力することで音楽を変更できたら、切り替えの手間は大きく削減される。

このような操作を可能にするためには、既存の携帯端末の入力手法を拡張する必要がある。現在、スマートフォンなどではタッチパネルによる入力が主であるが、他のアプリケーションの操作 UI も設置すると画面を占有し、操作領域を制限する。他の入力拡張手法としては端末を動かすジェスチャにより操作を行う手法 [1] や、iPhone の Siri *1のように音声を入力とする手法などがある。しかし、

¹ 筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻

Department of Computer Science, Graduate School of Systems and Information Engineering, University of Tsukuba

² 筑波大学システム情報系

Faculty of Engineering, Information and Systems, University of Tsukuba

a) ishiyama@iplab.cs.tsukuba.ac.jp

*1 <http://www.apple.com/jp/ios/siri/>

Sami らは公共の場におけるジェスチャや音声入力には心理的な抵抗があり、端末の利用が制限されることを示している [2]。また、Karlson らによると、ユーザの多くは携帯端末を片手で操作することを望んでおり [4]、カメラの前で手を動かさずようなジェスチャよりも小さな動作で手軽に行える入力が必要とされる。

Emilien らはコンピュータに対する入力手法として、タップのリズムパターンによる入力を提案している [12]。この手法では 2-6 拍のタップと、タップを行わない空白の組み合わせでインタラクションを行う。タップ入力は端末の物理ボタンや加速度センサー、マイクなどで用意に取得できるため、携帯端末の操作にタップ入力を利用しているものは多い [3]。Emilien らはタップのリズムパターンを操作に割り当て、1 次元のタップ入力でショートカットキーのような操作が行えることを示した。しかし、この研究ではリズムと操作がランダムに割り当てられており、多くの操作を利用したい場合には操作に対応したリズムパターンを暗記するのが困難になってくる。特に携帯端末は表示画面が小さく、リズムのヒントなどを表示するのも難しい。様々な機能の操作を行うためには、操作を行うためのコマンドから入力するリズムが連想できることが必要である。

そこで、本研究では、コマンドを発声した際の音のタイミングなどの情報に基づいてリズムに変換し、タップで入力を行うことにより操作を行う手法を提案する。例えば、「あいう」のように発声が連続する場合は、連続した一定のタイミングのリズムとする。また、「すとっぷ」のような発音が詰まる場合や、「ぼーず」のように伸びる場合はそれに準じてタイミングを変化させる。これにより行いたい操作のコマンドを思い浮かべることで入力を想起できる。また、端末側面などをタップして入力することで、アプリケーションの起動状況などに関わらず利用でき、歌手名の直接入力や明るさなどの設定変更操作が端末の状態によらず可能となる。

本手法を適用する上での予想される問題として、複数のコマンドから同一のリズムが重複して生成される可能性が挙げられる。重複が多い場合、リズムタップ入力から一意にコマンドを確定することが困難になり、リズムタップ入力後に複数のコマンドから選択するなどの手間が生じてしまう。そこで、我々は発音のアクセント情報を利用してリズムの種類を増加し、重複を減少させる手法の検討を行った。入力の種類の増加はユーザに負担を与えるが、発音情報を利用したアクセントリズムの利用はコマンドからのリズムの推定を妨げず、比較的容易に入力が行える。

本稿の構成を以下に示す。2 章ではまず携帯端末においてタップ入力を行っている研究について述べる。3 章ではリズムタップ入力の説明として、発声タイミング情報のみを利用してリズム変換を行う場合の変換規則の説明と、この場合のリズム種類数を示すことで重複の問題について述

べる。その後アクセント情報を用いたリズム変換規則の説明と重複の減少について述べる。4 章ではユーザがリズムタップ入力を用いてどのように操作を行えるかを示す。5 章では作成した試作機の実装について述べる。6 章でリズムタップ入力の入力精度などを調査した実験について述べ、7 章でまとめる。

2. 関連研究

Scott らの WhackGesture [7] は、腰につけた携帯端末をタップする動作と揺する動作を組み合わせたジェスチャーにより、端末を手にとらずに電話応答などの操作を行うことを目的とした研究である。よく使う機能の操作を素早く行うために、入力の種類は少なくジェスチャーも短いもので操作を行うことができる。

また、リズムをセキュリティのために使用する研究として、Westeyn らの Recognizing Song-Based Blink Patterns [8] と、Jacob の TapSongs [9] が挙げられる。Song-Based Blink Patterns はリズムカルにまばたきをすることで個人の認証を行う研究であり、TapSongs はイヤホンの音量ボタンを用いてリズムを入力することで音楽プレイヤーのロックを外すという研究である。これらはある程度の長さのあるリズムを入力する際にユーザによって微妙な違いが発生することを利用して安全性を高めている。

これらの研究は Emilien らの研究と同じく、あらかじめ設定したリズムを入力することで操作を行う手法である。我々の手法では言語の発音特徴などを利用してコマンドからリズムを生成し、行いたい操作から入力を連想することができる。

Roger らの Query by Tapping [13] は楽曲からリズムを抽出しそのリズムをタップ入力することで楽曲の検索を行う研究である。楽曲の音楽情報を検索の入力とすることで、入力を暗記することなく曲を検索することができる。本研究では文字情報をリズム化して入力とすることで、歌手名やジャンルの直接選択や、アプリケーション操作など多種の操作を実現することを目指す。

3. リズムタップ入力

本章ではまず、コマンドからのリズム変換規則として単純に発音タイミングのみを用いた手法についての説明を行う。次にその規則により生成されたリズムの種類を求めることによりリズム重複の危険性を調査し、対策を検討する。その後、リズム変換規則と入力手法の拡張としてのアクセント情報を利用したリズムタップ入力について述べ、同様にリズム種類を求めることで重複の減少について議論する。

3.1 発話タイミングを用いたコマンドのリズム変換規則

コマンドをリズムに変換する手法として、基本的には 1 拍ごとにタップを行うような入力に変換する。日本語にお

ける拍とは、主に文字で書いた際の仮名1文字が1拍となる[5]。俳句における五・七・五とはこの拍を数えたものであり、拍を用いたリズムは一般的に理解しやすいと考える。「あ」や「ゆ」のような拗音は仮名1文字で1拍とはならず、「ちゃ」や「ひょ」のような仮名+拗音の組み合わせで1拍となる。拍の変換の例として、ブラウザを起動するコマンドが[ブラウザ]だとすると、このコマンドは4拍であり、“タップ・タップ・タップ・タップ”といった入力に変換される(図1右上段)。

コマンドに促音符「っ」や撥音符「ん」、長音符「ー」が含まれる場合は発音リズムが変化するため、これに対応してタップ間隔を1拍空ける入力や、タップを長押しするといった入力に変換する。これらの仮名もそれぞれ1拍として数えられるが、これらは発音時に前の母音とつながって発音される。音の区切りである音節の数え方では「お・ん・が・く(4拍)」という単語では「おん・が・く(3音節)」と数えられ[6]、促音、撥音、長音がコマンドに含まれる際は発音時のリズムが変化する。促音と撥音の箇所では音の空白が発生し、長音の箇所では前の音が伸びて発音される。コマンドをリズムに変換する際にはこの発音特徴をリズムに対応させ、前者の場合は前のタップから1拍空け、後者の場合は前のタップを長押しするといった入力に変換する。以降では1拍空ける入力を“ブレイク”，タップの長押しを“ロングタップ”と表記する。例えば、[すとっぶ]というコマンドは“タップ・タップ・ブレイク・タップ”という入力になり(図1右中段)，また[ぼりゅーむ]というコマンドをリズム変換した場合は“タップ・ロングタップ・タップ”といった入力となる。

長音に関して、長音符「ー」が表記されていないが伸ばして発音する箇所にもタップ長押しの入力を割り当てる。長音符は日本語ではあまり用いられず、長音符の箇所に母音が割り当てられて表記される。例として、「けいたい」という表記でも発音時には「けーたい」のような音で発音する。このような発音は、「しい」のように前の母音と同じ母音が続く場合と、「こうしん」のような「お」の母音のあとに「う」が続く場合、また「けいたい」のように「え」の母音のあとに「い」が続く場合に発生する。コマンドにこれらの仮名の繋がりがあった場合は2つをまとめて“ロングタップ”に変換する。先ほどの例の「こうしん」の場合、“ロングタップ，タップ，タップ”といった入力となる(図1右下段)。

3.2 変換リズムの重複

本手法ではコマンド名に対して規則的にリズムを割り当てるため、複数のコマンドのリズムが重なる場合がある。リズムが重複してコマンドを一意に確定できない場合はリズムタップ入力のみで任意の操作を行うことが難しくなる。そこで、まず3.1で述べた方式でリズム変換を行った

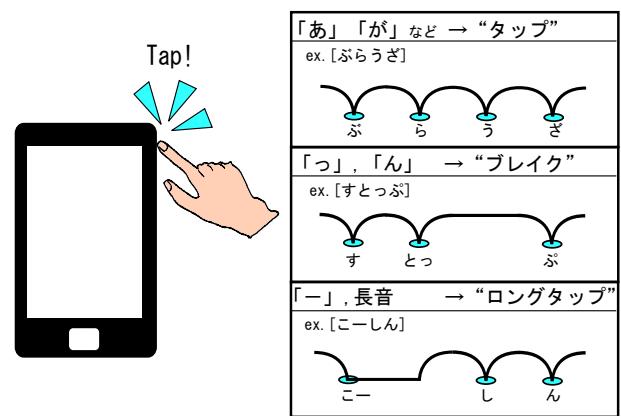


図1 コマンドからのリズム変換規則一覧

場合にリズムがどの程度存在するかを検証し、その結果を考察する。

リズムのパターンの検証のために、リズムタップの特徴を以下に述べる。

- リズムの最初は“タップ”又は“ロングタップ”である。
- リズムの最後も同様“タップ”又は“ロングタップ”であり、“ブレイク”で終わることは無い。
- “ロングタップ”の直後に“ブレイク”が続くことは無い。

この特徴を満たすものとして、文字数Nのコマンドに対して 2^{N-1} 種類のリズムが存在する。

本手法でリズム変換を行うコマンドとしてはアプリケーション名や連絡先選択などのための人名が全体の多くを占めると考えられ、文字数が6~8程度のコマンドが多くなると予想される。文字数が6~8のリズムは合計で224種類となるが、連絡先が200人を超える場合も多く、これは非常に少ないといえる。また、文字数が3文字以下のコマンドに対してはリズムの重複が高い確率で発生してしまう。これを改善するためには、リズム変換手法や入力手法を拡張して追加要素を加えるなどで、リズムの種類を増加させる必要がある。

リズム変換手法の拡張を考える際に、コマンドからのリズム連想を妨げるような手法を用いるとコマンドに割り当てられたリズムの連想が困難になる。そのため、発音の特徴を利用した拡張を行う必要がある。また入力手法の拡張について、現在の入力日本語の拍を利用した1次元時系列情報の入力であり、タップ位置を区別せずに入力を行う。そこでタップ位置を区別することで入力手法の拡張を行うことができると考えられる。しかし、例えば母音ごとに5つのボタンを用意し、仮名の母音の種類によって5ボタンを使い分けてリズムタップ入力を行うような、あまり多くの拡張を行うと入力の際に考える必要が生じ入力速度や精度が低下することが予想される。そのため、入力の拡張は2~3ボタン程度での最低限の拡張に抑える必要がある。

3文字	
“HLL” かめら	“LHL” すつつ
“LHH” じゅしん	
4文字	
“HLLL” ねくすと	“LHLL” いばらき
“LHHL” くらしっく	“LHHH” いしやま

図 2 3,4 文字数単語のアクセント高低推移

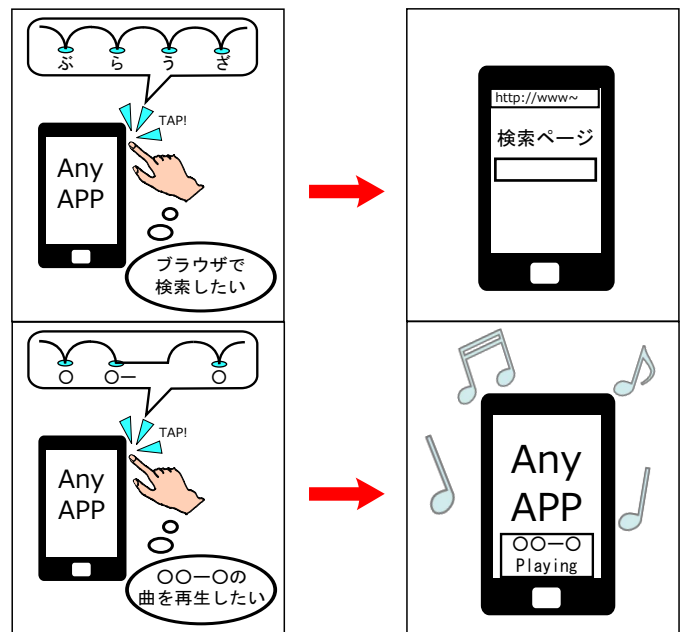


図 3 リズムタップ入力による操作

3.3 アクセント情報を利用したリズムタップ入力

3.2 で述べた拡張の要件を受け、本研究では発音時のアクセント情報を利用してリズム種類を増やすことを検討する。アクセント情報は発音情報の1つであり、コマンドからのリズム推定は行いやすいと考えられる。また、日本語におけるアクセントは高低アクセントといわれており、発音の強さではなく音の高低（ピッチ）が変化する。そのため、入力の拡張はタップ箇所を2箇所拡張するだけで済む。日本語アクセントのなかで最も一般的に使われている東京式アクセントの特徴を以下にまとめる。

- ピッチの下降箇所が零箇所または一箇所現れる
- ピッチの下降は拍と拍の間で発生する
- 文頭や単語において、ピッチの下降が1拍目と2拍目の間でない場合は1拍目のピッチは低く、1拍目と2拍目の間でピッチの上昇が発生する

例として、3音節の単語の場合にピッチの高い箇所をH、低い箇所をLと表記すると、“HLL”、“LHL”、“LHH”の3つのアクセントが存在し（図2上段）、4単語の場合は“HLLL”、“LHLL”、“LHHL”、“LHHH”の4つのアクセントが存在する（図2下段）。アクセントの種類はN音節の単語に対してN種類存在する。

アクセント情報を利用することで3.2で挙げた各文字数でのリズムパターンを増加させることができる。促音や長音の箇所ではアクセントの変化が起こらないため、リズム種類の増加分は各リズムのタップ入力数分であることができる。2~6拍のリズム種類数は、2拍の場合は3種類、3拍の場合は9種類、4拍の場合は23種類、5拍では57種類、6拍では135種類となる。

単純に拍数でみると3.1で述べた変換手法と比べてリズム種類数の増加分は多くはないが、歌手名などを選択する

ための人名コマンドや複数単語からなるコマンドをリズム変換する場合、苗字と名前、または単語ごとにアクセントが存在する。6拍の単語のリズムは135種類だが、苗字の拍数が4、名前の拍数が2の人名をリズム変換した場合、苗字には4拍のリズム23種類が存在し、名前には3拍のリズム9種類が存在するため、 $23 \times 3 = 69$ 種類のリズムが追加で存在する。よって、6拍のリズムは合計で204種類存在することとなる。他にも6拍の組み合わせはあり、合計のリズム数は大きく増加する。しかし、拍数が3以下の単語のリズム種類はそれでも少ないため、アクセント情報の利用のみでは実用の際にはリズム入力後に複数のコマンドからの選択を行うなどの必要がある場面が存在する。

4. リズムタップによるコマンド入力システム

4.1 システム概要

本システムの利用イメージを図3に示す。ユーザはブラウザやメールなどのアプリケーションを操作中に端末の側面などでリズムタップを行うことでコマンドを入力し、他のアプリケーションを含めた様々な機能の操作を行うことができる。これにより、画面を切り替えることなく後ろで動いている音楽プレイヤーの操作を行ったり、UIを表示せずに端末やアプリケーションの設定を変更したりなどが可能となる。また、アプリケーションや連絡先を探すことなく、任意の画面からそれらを直接リズムタップ入力を選択して呼び出すことも可能である。

4.2 コマンド入力

リズムタップ入力による操作の詳細について説明する。ユーザは任意の画面で〔(アプリケーション名などのコマンド種別)〕コマンド+〔(それに対する操作内容)〕コマンド

のように、コマンドプロンプトにおけるコマンドとコマンドオプションの入力のようにリズムタップを行うことで操作を実行することができる。これは、例えば人名がコマンドとして入力された際に、それが電話の連絡先なのか、音楽プレイヤーでの歌手名なのかの判別をすることが困難であり、[電話] コマンド+ [(人名)] コマンドのように入力を行うことで行いたい操作を特定するためである。またこの入力方式により、最初に受け付けるコマンドはコマンド種別のみ、以降に受け付けるコマンドはその機能に存在する操作のみに制限できるため、リズムの重複も削減できる。

コマンドとしては、現在は以下の操作が可能である。

- [電話] コマンド：ダイヤル画面を起動する。
 - － [(人名)] オプション：指定した人の電話番号がセットされた状態でダイヤル画面を起動。
- [ブラウザ] コマンド：デフォルトのブラウザをホーム画面で起動する。
 - － [(お気に入りページ名)] オプション：指定したページを開いてブラウザを起動。
- [音楽プレイヤー] コマンド：デフォルトの音楽プレイヤーを起動する。
 - － [(曲名)] オプション：音楽プレイヤーを起動せず、指定の曲を再生。
- [メール] コマンド：デフォルトのメールアプリを起動する。
 - － [(人名)] オプション：指定した人のメールアドレスがセットされた状態でメール作成画面を起動。
 - － [受信] オプション：受信ボックスを表示。
 - － [送信] オプション：送信ボックスを表示。
- [(任意のアプリ名)] コマンド：上記以外のアプリケーションの名前を入力することでそのアプリケーションを起動する。

最初にアプリケーション名などのコマンドが入力されると短いバイブレーションによるフィードバックが発生し、入力が行われたことが通知される。コマンドオプションの入力が可能なアプリケーションの場合、一定時間入力待ちの状態となり、その状態でリズムタップ入力を開始すると追加の入力を受け付ける。入力待ち状態で何もしない、もしくは1度だけタップ入力を行うことでそのコマンドのデフォルト操作が実行される。リズムタップを間違えた場合はロングタップし続けることで現在の入力をキャンセルし、入力をやり直すことができる。入力のキャンセルは長いバイブレーションにより通知される。

4.3 リズム重複時のコマンド選択操作

3.2 で述べたように、現在は特に短いコマンドでリズムの重複が発生する可能性がある。そのため、入力したリズムに対して複数のコマンドが存在する場合、そのコマンドの一覧をユーザに提示し、選択してもらうことで操作を実

行することとなる。コマンドの提示は、直近で使用したものの、入力頻度が高いものの順で優先的に表示される。

5. 実装

本研究では試作機として、SAMSUNG Galaxy SII *2を用いてリズムタップ入力を認識するアプリケーションを開発した。現在はこのアプリケーションが前面に表示している時にリズムタップ入力が可能となる。

タップの入力には音量調整ボタンを利用しており(図4)、上に配置されている音量増加ボタンを押すことで高いアクセントの入力とし、下に配置されている音量減少ボタンを押すことで低いアクセントの入力としている。例えば“LHL”の入力は“下ボタン, 上ボタン, 下ボタン”といった入力となる。

携帯端末のタップ入力の認識は加速度センサーやマイクを用いているもの [7] [10] やタッチパネル上でタップを行うもの [11] が多くあるが、本研究で入力にボタンを用いている理由には2つの理由がある。1つ目は、リズムタップ入力において“ロングタップ”を利用するので、タップのタイミング情報のみでなく指の接触情報も取得する必要があるためである。2つ目は、タップの入力は端末をポケットに入れているときや、端末がスリープ状態であるときにも行いたいと考えており、そのような状態でも入力を認識するには物理的なボタンが適しているためである。しかし将来的には端末の側面や背面に設置した接触センサーなどでタップタイミングと位置を認識するなど、本体に設置されている既存の物理ボタンなどの入力手法を占領せずに入力を行えるようにする必要があると考える。

リズムの認識はボタンが押されている時間と、リズムタップ入力中でボタンの押下が行われていない時間を計測して認識を行う。ボタンが押されている時間が700msec以下で“タップ”、700msec~1500msecで“ロングタップ”として認識される。1500msec以上の長押しで入力がキャンセルされる。また、入力中でボタン押下が行われていない時間が600msec~1600msecで“ブレイク”として認識され、1600msec以上で入力の終了とする。この時間は事前に5名にタップ入力の試用を行ってもらった際に適切とされた時間の平均を設定している。

また、コマンド種別の入力とされた場合は入力終了後に2500msecの追加入力待ち時間が発生する。この時間内に詳細な操作のコマンドリズムを入力することでその操作が実行される。この詳細操作のコマンドのリストはコマンド種別ごとに設定されており、その操作として存在するコマンドのみ認識され、存在しないコマンドが入力された場合はコマンド種別のデフォルトの状態の操作が行われる。入力待ち時間中に何も入力されない場合も同様にデフォル

*2 <http://www.samsung.com/jp/galaxys2/>

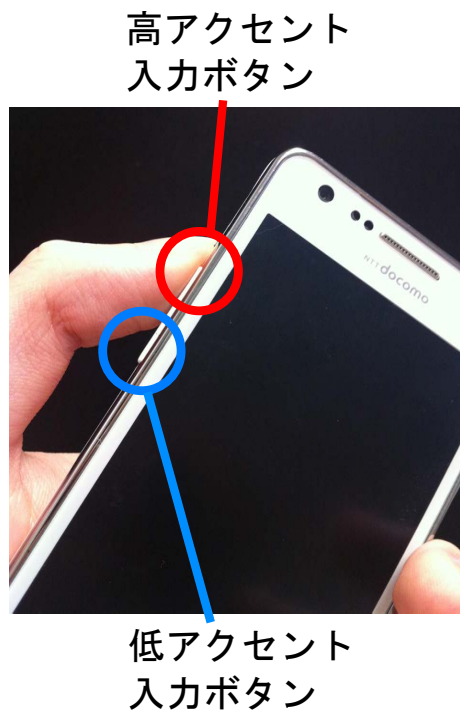


図 4 アクセント入力ボタン

ト状態の操作が行われる。

アプリケーションの起動や操作は_intentを用いて行う。_intentは Android^{*3}の機能であり、アプリケーション同士やアプリケーションとシステム橋渡しし、アプリケーションから外部アプリケーションやシステムの起動や操作などを行うことができる。電話、ブラウザ、メールアプリケーションの操作には暗黙的_intentを用い、入力されたコマンドに対応したメールアドレスや電話番号、URLを_intentのパラメータに指定することでそれぞれのデフォルトのアプリケーションにパラメータがセットされた状態で起動する。その他のアプリケーションの起動には明示的_intentを用いる。_intentのパラメータにアプリケーションのアクティビティを直接指定し、起動を行う。

6. 実験

アクセント情報を利用したリズムタップ入力の認識精度とリズムやコマンドの連想性を調査する実験を行った。まず実験 1 として指定したコマンドに対するリズムの連想と入力の精度を調査した。次に実験 2 として大まかな操作指示によりコマンドを特定し入力を行えるかを調査する実験を行った。以下で各実験の詳細を示す。

6.1 実験 1

6.1.1 実験内容

アクセント情報を利用したリズムタップ入力において、被験者にコマンドを提示し、そこからリズムを推測して正

^{*3} <http://www.android.com/>

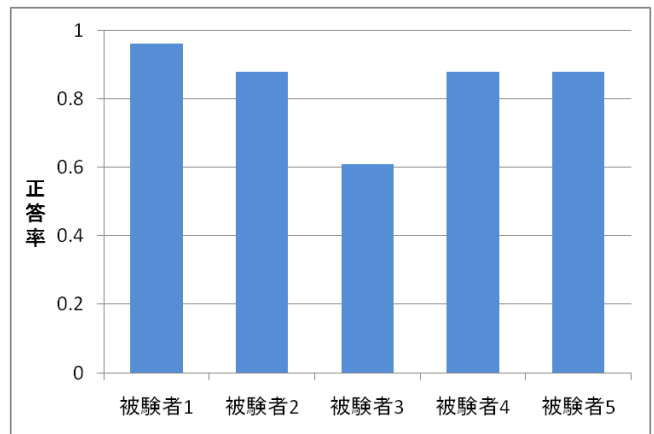


図 5 被験者別正解率

しくリズムの入力を行えるかを調査した。コマンドは[ブラウザ]や[ツイッター]のようなアプリケーション名や操作コマンドなどが 25 個、[人名]コマンドが 25 個の計 50 個を提示し、入力を行ってもらった。コマンドの文字数は 2~7 文字である。被験者にはあらかじめリズム変換規則を伝え、十分な練習を行ってもらった。被験者は 22~24 歳の日本語を母語とした大学生、大学院生 5 名である。4 名はスマートフォンの操作に慣れており、全員が日常会話では主に東京式アクセントを使用していた。

6.1.2 実験結果

被験者が正しくリズムを入力できた正解率の平均は 83.4%であった。被験者の中で最高の正解率は 96.2%で、最低の正解率は 61.0%であった。被験者ごとの正解率を図 5 に示す。不正解のうち、単純な入力ミスによるものは 18.2%であり、2 ボタンを用いた入力の精度は高いと言える。また入力時間について、コマンドを提示してから被験者が入力を開始するまでは平均 2.4 秒であり、最長で 6.2 秒であった。

6.1.3 考察

ほとんどの被験者は高い確率でアクセントリズムを正しく入力できていた。また入力時の様子を見ると、ほとんどの場合被験者はコマンドに目を通した後すぐに入力を開始していた。被験者のコメントとして「あまりリズムを悩まなくても思ったとおりに押したらできた」というものもあり、アクセントリズムはコマンドから連想しやすいと考えられる。

しかし、図 5 に示されるように、被験者 3 の正解率は低かった。被験者 3 のコマンド発声を調査すると、発声アクセントは東京式アクセントに非常に近いもので、正解率の高い他の被験者と同程度にアクセントリズムを理解していたと考えられる。実験後にこの被験者からは、「普段アクセントは気にしていないので高いのか低いのが感覚的にわかりづらかった」という意見が得られた。しかし実験後にこの被験者が間違ったコマンドのリズムを再度尋ねたと

ころほとんど正解したため、アクセントの高低感覚になれることでこのようなユーザもアクセント入力を習得できると考える。

全ての被験者に共通する傾向として、撥音「ん」の前後でピッチの高低が変わる箇所での入力を悩むことが多かった。撥音は単独で音節を構成せず、直前の語と結びつくため、本手法のリズム変換規則では“ブレイク”の入力を割り当てている。しかし、例えば[おんがく]というコマンドにおいて、アクセントは「お」と「ん」の間で変化する。このため、被験者からは「ん」の箇所もタップした方が直感的であるという意見をもらった。しかしその他の箇所では撥音の入力を“ブレイク”とした方がわかりやすいという意見もあり、同じ撥音でも場合により感じ方が異なると考えられる。今後はより日本語の特徴を調査し、ユーザが直感的に操作できるよう調整する必要がある。

6.2 実験2

6.2.1 実験内容

被験者が操作内容からコマンドとリズムを推定し、実際に操作を行えるかの調査を行った。この実験ではオプションコマンドを含む複数コマンドの入力を含めたいくつかの操作を指示し、リズムタップ入力により操作を行ってもらった。操作の指示は具体的なコマンド名では行わず、被験者には大まかな操作の説明のみを行った。

被験者は22~24歳の日本語を母語とした大学生、大学院生の5名であり、全員がスマートフォンの操作に慣れており、日常会話では主に東京式アクセントを使用していた。被験者には事前にリズム変換規則と複数コマンドの説明を行い練習をしてもらった。練習で指示したコマンドはその後の本番タスクで使用するものとは異なったものとした。本番タスクで行ってもらった操作は以下である。

- (1) 設定画面を起動する ([設定] コマンド)。
- (2) カメラを起動する ([カメラ] コマンド)
- (3) ブラウザのお気に入りにあるヤフー^{*4}のトップページを表示する ([ブラウザ] コマンド + [ヤフー] コマンド)
- (4) 指定の人物に電話をかける ([電話] コマンド + [指定の人名] コマンド)。
- (5) 指定の曲を再生する ([音楽] コマンド + [指定の曲名] コマンド)

本番タスクでは指示に対して正しくコマンドを選択できるか、また推定したコマンドを正しくリズムタップ入力できるかを記録し、本番タスク終了時に本手法を用いた操作についてのコメントをもらった。

6.2.2 実験結果

全ての被験者が(1)~(3)の入力を1度で成功させることができた。(4)の電話を起動する操作では、1人の被験

者が[電話]コマンドの入力を間違えていた。(5)では2名が1度で成功させたが、他の被験者は[音楽]コマンドを入力するところで[ミュージック]コマンドを入力していた。

6.2.3 考察

(1)と(2)は全ての被験者が正解したが、「設定」はコンピュータ操作時によくみる項目であり、「カメラ」はアプリケーション名が統一して広く認識されているためであると考えられる。(3)も全ての被験者が正解した。ブラウザについては[ブラウザ]コマンドを入力することで端末デフォルトのブラウザが起動するが、被験者の中には自分の端末に複数ブラウザをインストールしている者もあり、「自分で起動するブラウザの設定をするか入力時に選びたい」といった意見も得られた。音楽プレイヤーやメールソフトなども端末デフォルトのもの以外を使っていることが考えられるので、幾つかのコマンドと起動アプリケーションの対応は手動で調整することも検討する。

(4)の入力ミスは6.1.3で述べたように撥音箇所でのアクセントが変化したために入力を悩み、間違えていた。このコマンドは3文字と短いものであったため、ほとんどの被験者は少し考えた後に正しく入力できていたが、やはり調整が必要であると考ええる。

(5)について、音楽を再生するアプリケーションの名称が被験者により異なっていたために間違いが発生した。実験で使用した端末(5で述べたもの)にインストールされていたアプリケーション名が「音楽」であったためコマンドをそのように設定していたが、端末によっては「ミュージック」である場合があり、被験者が混乱していた。他にもユーザによって操作名の認識が異なるものや操作名があいまいなものは多くあると考えられる。1つの操作に対して複数のコマンドを設定することでこのような操作の入力を行うことは可能であるが、リズム重複の問題も考慮して今後検討していく必要がある。

7. まとめ

コマンドを一定規則でリズムに変換して入力とすることで操作と入力の意味的な対応を保ち、端末の状態によらず任意の操作を実行できる手法を提案した。コマンドをリズムに変換する手法について、コマンドの発音タイミングを利用しての変換とその際のリズムの重複危険性について調査し、重複を減少するための拡張手法としてアクセント情報を用いたリズム変換手法を提案した。また、リズムタップ入力の入力精度とコマンドからの連想性を調査する実験を行った。実験において、ほとんどの被験者がコマンド名とリズムを高い精度で正しく連想し、入力を行うことができた。

今後の課題としては、アクセントリズム入力では直感的に入力が行えない場合があったため、よりユーザが連想したとおりのリズムで入力が可能となるようなリズム変換手

^{*4} <http://www.yahoo.co.jp/>

法と入力手法を検討する。また、リズムが重複した場合やコマンドが曖昧な場合でも行いたい操作を実行できることを目指す。

参考文献

- [1] Jani Mntyjrv. Juha Kela, Panu Korpip, and Sanna Kallio. Enabling fast and effortless customisation in accelerometer based gesture interaction. In Proc. MUM' 04, pp. 25-31, (2004).
- [2] Sami Ronkainen, Jonna Hkkil, Saana Kaleva, Ashley Colley and Jukka Linjama. Tap input as an embedded interaction method for mobile devices. In Proc. TEI' 07, pp. 263-270, (2007).
- [3] Simon Robinson, Nitendra Rajput, Matt Jones, Anupam Jain, Shrey Sahay and Amit Nanavati. TapBack: towards richer mobile interfaces in impoverished contexts. In Proc. CHI' 11, pp.2733-2736, (2011).
- [4] Karlson, A. K., Bederson, B. B. and Contreras-Vidal, J. L. Understanding Single-Handed Mobile Device Interaction. Technical report, Department of Computer Science, University of Maryland(2006).
- [5] 榑保格. 『日本の言語学』第2巻. 大修館書店. 1980年.
- [6] 佐伯哲夫, 山内洋一郎. 『国語概説』. 和泉書院. 1983年.
- [7] Scott E. Hudson, Chris Harrison, Beverly L. Harrison and Anthony LaMarca. Whack Gestures: Inexact and Inattentive Interaction with Mobile Devices. In Proc. TEI'10. pp.109-112. (2010).
- [8] Westeyn, T. and Starner, T. Recognizing song-based blink patterns: Applications for restricted and universal access. Proc. FGR '04. Washington, D.C.: IEEE Computer Society, pp.717-722.(2004).
- [9] Jacob Otto Wobbrock. TapSongs: tapping rhythm-based passwords on a single binary sensor In Proc. UIST'09. pp.93-96. (2009).
- [10] Hiroyuki Manabe and Masaaki Fukumoto. Tap control for headphones without sensors. In Proc. UIST'11. pp.309-314. (2011).
- [11] Felix Xiaozhu Lin, Daniel Ashbrook, and Sean White. RhythmLink: Securely Pairing I/O-Constrained Devices by Tapping. In Proc. UIST'11, pp.263-272, (2011).
- [12] Emilien Ghomi, Guillaume Faure, Stphane Huot, Olivier Chapuis and Michel Beaudouin-Lafon. Using Rhythmic Patterns as an Input Method. In Proc. CHI' 12, pp.1253-1262, (2012).
- [13] Jyh-Shing Roger Jang, Hong-Ru Lee and Chia-Hui Yeh. Query by Tapping: A New Paradigm for Content-Based Music Retrieval from Acoustic Input. IEEE Pacific Rim Conference on Multimedia, pp.590-597, (2001).