

GUIからのVUI生成による入力手順数の減少手法

篠原佑友¹ 岩田一² 白銀純子³ 深澤良彰¹

概要: ユーザーが発話した言葉の意図を解釈して動作する Voice User Interface (VUI)が Graphical User Interface (GUI)に組み込まれたデバイスが市場に登場している。VUIの利点は手の操作や視線への集中を必要とせず、学習コストが低いためにお年寄りや障害のある人でも使えること、声色に感情などの人格を付与できることなどが挙げられる。VUIとGUIが組み込まれたデバイスは視覚と音声の両方で操作できる利点がある一方で、VUIでは詳細で長い情報が入力面では困難であること、また出力面では長い文面の情報はユーザーが受け取り切れないなどの問題があることから、現在のVUIの用途は一言で済む情報の入力と出力にとどまっている。そこで本研究では、ユーザーの入力手順数を減らしユーザビリティを向上させることを目的として、GUIをもとにVUIを生成し、GUIとVUIを組み合わせる使用手法を提案し、評価を行う。

A method to reduce the number of input steps by generating VUI from GUI

YUTO SHINOHARA^{†1} HAJIME IWATA^{†2}
SHIROGANE JUNKO^{†3} FUKAZAWA YOSHIAKI^{†1}

Abstract: Devices with Voice User Interface (VUI) integrated into Graphical User Interface (GUI) have been appeared. VUIs can interpret users' speaking intentions. The advantages of the VUIs are operations without hands and concentrations of line of sight, easy use by elderly people and people with disabilities because of low learning costs, and availability of voice tones with characteristics, such as emotions. Although devices with both VUIs and GUIs can be operated with visions and voices, VUIs are not good at detailed and long inputs, and users difficultly recognize long outputs. Thus, current VUIs handle only short phrases and sentences. In this study, to reduce the number of user input steps and to improve usability, we propose a method to generate VUIs based on GUIs and to use the combination of GUI and VUI, and evaluate it.

1. はじめに

ハードウェアとソフトウェアの進歩により、ユーザーインターフェース（以下 UI）の種類が増えている。UIの種類が増えることで、コンピュータと人の情報の入出力形態が多様化し、UIはマルチモーダルになっていくと考えられている[1]。そんな中で自然言語処理、機械学習といった技術が発展したことで、音声認識の精度が飛躍的に向上し、ユーザーが発話した言葉の意図を解釈して動作する UI である Voice User Interface(以下 VUI)が Graphical User Interface(以下 GUI)に組み込まれたデバイスが市場に登場している[2]。VUIの利点は手の操作や視線への集中を必要とせず、学習コストが低いためにお年寄りや障害のある人でも使えること、声色に感情を付与できることなどが挙げられる[3]。VUIとGUIが組み込まれたデバイスは視覚と音声の両方で操作できる利点がある一方で、VUIでは詳細で長い情報が入力面では困難であること、また出力面では長

い文面の情報はユーザーが受け取り切れないなどの問題があることから、現在のVUIの用途は数語で済む情報の入力と出力にとどまっている。そのため、GUIとVUIのユーザーの状況に合わせた実用的な統合が確立されれば、デバイスの利便性はさらに高まると考えられる。

そこで本研究では、選択入力が連続するGUIアプリケーションにおける入力手順数が不要に増加してしまう問題に対して、GUIからVUI生成する手法を提案する。選択入力が連続する状況のGUIアプリケーションを作成し、GUIソースコードを元に「ユーザーがGUIの複数選択入力を実現したい1つの行為」を元にVUIを生成する。そしてGUIアプリケーションにVUIを付与したUIを作成する。GUIにVUIを付与したUIのアプリでユーザーにタスクを行ってもらい、既存GUIと提案手法UIを比較し、入力手順数が減ることでユーザビリティが向上するかどうかについて評価する。これにより、GUIとVUIを搭載したデバイスで選択入力が連続する場合は入力手順数がVUI入力の方が少なく、ユーザビリティが向上するという仮説検証をし、GUIとVUIの効果的な統合に寄与できると期待される。

¹ 早稲田大学
Waseda University
² 神奈川工科大学
Kanagawa Institute of Technology
³ 東京女子大学
Tokyo Woman's Christian University

2. 関連研究

VUI の活用のために、VUI アプリケーション使用時におけるパフォーマンスへユーザー特性が与える影響に関する研究がされている。Myers らは VUI の使用経験やプログラミング経験などのユーザー特性がユーザーが新しい VUI アプリケーションを扱った際のパフォーマンスにどう影響するかについての研究をしている[4]。デスクトップパソコン上の GUI を伴う VUI カレンダーアプリで被験者にタスクを行ってもらい、事前にユーザーに行ったアンケート結果とタスク実行時のパフォーマンスの関連を分析した。調査の結果、性格に慎重な傾向があるユーザーは視覚的な VUI にまつわる操作の指示をより欲しがること、タスク実行時のパフォーマンスにプログラミング経験は大きな影響はないが以前の VUI 使用経験による同化バイアスは大きな影響があること、技術力に自信があるユーザーはトライアンドエラーによって試行する傾向があることなどがわかった。これらのことから、新しい VUI をデザインする際には VUI の使用経験があるユーザーに関しては、他の VUI との操作上の違いを明確にすることが有用であることなども述べられている。この研究は、ユーザー特性が新しい GUI と VUI によるアプリケーションを扱った際のパフォーマンスにどう影響するかについての調査がされているが、アプリケーションの機能によって GUI と VUI のどちらの UI がユーザーパフォーマンスに寄与するかについては触れられていない。

また、マルチモーダルな UI におけるユーザビリティ評価をモデル化する研究がされている。Schaffer らはマルチモーダルな UI が標準化される前にマルチモーダル UI におけるユーザビリティの評価手法が統一されていない問題に対して、自動ユーザビリティ評価をモデル化するという研究を行った[1]。モバイル端末用のタッチ入力ができる GUI を伴った VUI モバイルアプリケーションを作成し、レストランの予約するタスクを被験者に行わせた。また本実験での VUI は自動音声認識技術が完成されたことを前提にシミュレーションをしているため、人間によるオペレーターが実験室とは別室で応答することで実装された。レストランの予約タスクはトップページに都市、料理、希望時間、人数の 4 項目があり、それぞれの項目を選択すると表示される選択肢一覧の中から 1 つを選択して予約完了ボタンを押すというものである。なおアプリケーション上の全ての 1 つ 1 つの操作は GUI と VUI の双方で可能となっていたため、トップからの項目選択は GUI、選択肢一覧からの選択は VUI といったように UI を混ぜて操作することもできた。そのため、VUI を使用した場合には項目選択をして、一覧を表示させるという過程を飛ばして、何を選択するかが入力が直接行え、ユーザーの手順を 1 つ減らすことができた。そして被験者がタスクを行う際の手順の数、どの UI

を選択したかなどを記録した。調査の結果、ユーザーがタスクを行う際にどちらの UI 選択をするかには入力手順の数がどれだけ少ないかが最も影響を与えることがわかった。また実験中に GUI と VUI にそれぞれ意図的なエラーを加えたところ、UI のエラー率があがると、その UI を使用する割合は減少するが、GUI の方がエラーによる使用率の甚大な減少が見られて VUI のエラー率よりも GUI のエラー率の方がユーザー行動への影響が大きいことがわかった。次に、被験者実験を行った際のデータを元に、自動ユーザビリティ評価を UI の数、エラー率などを定数として置き、考案した式を作成してモデル化した。結果として、被験者の平均データに対して自動ユーザビリティ評価はある程度の予測がきくが、個人単位にはあまり効果がないことがわかった。この研究では、GUI と VUI によるモバイルアプリケーションを扱った際の手順数の変化と使用率についての評価がされているが、選択入力が連続してしまい GUI 上では手順が増えてしまう入力行為を VUI 上で入力手順を減らすということについては触れられていなかった。

GUI に向けて作成された既存のデザインガイドラインを VUI にまで拡張する研究がされている。Murad らは現在のデザインガイドラインが GUI を重視して作成されたために、VUI デザインにおける要素を包括しきれていない問題に対し、VUI デザインについての既存論文と現在のデザインガイドラインの比較する調査を行った[5]。ここで言及されているデザインガイドラインとは Nielsen[6]、Norman[7]、Shneiderman[8]によって作られた GUI のデザインガイドラインのことである。調査の結果、10 個にカテゴライズされた既存のデザインガイドラインのうち、「一貫性」に関して言及されているものはなかったが、それ以外の 9 つのデザインガイドラインには内容が概ね重複していることがわかった。また、GUI を基準に作られた 10 個のデザインガイドラインだけでは包括できない、「透明性/プライバシー」と「社会的なコンテキスト」という 2 つの VUI にまつわるデザインガイドラインの提案もされた。「透明性/プライバシー」は発話時における周囲への情報流出を考慮すべきというもので、「社会的なコンテキスト」とは公共の場での発話はしづらいことなどを考慮すべきというものである。また Murad らは上記の調査に加え、実際に GUI デザインガイドラインを VUI に適応した場合についての調査を具体的なユースケースを添えてマッピングすることで行った[9]。これらの研究では、GUI のガイドラインを元に VUI をデザインする際に必要なデザインガイドラインの確立について触れられているが、GUI と VUI の使い分けについて具体的に触れられていない。

3. Voice User Interface

3.1 VUI が搭載されている製品

市場に出回っている代表的な VUI としては、Apple 社の

Siri[10], Microsoft 社の Cortana[11], Amazon 社の Alexa[12], Google 社の Google Assistant[13]などがある。それぞれ、モバイルや PC、室内に設置するスピーカー、スピーカーとタッチパネルが融合したものといたった形態で存在する。

3.2 VUI の特徴

VUI とは音声でコンピュータを操作できる対話型のインターフェースのことである。Google のマテリアルデザインガイドライン[14]によると、サウンドデザインや音楽では代替できない情報をユーザーからの音声発話とデバイスからの音声読み上げによって伝達する方法とされている。単純な音による操作ではなく、対話的であり発話意図への解釈があるというのが特徴である。スマートスピーカーに搭載される形で近年普及している[15]。

3.3 VUI を構成するもの

VUI を構成するもので、本研究に特に関わる要素について記載する。なお VUI の種類によって要素を指し示す名称が異なる場合があるが、今回は Google Assistant[17]における名称で説明する。

(1) インテントとエンティティ

ユーザーが行いたい行為 1 つにつき、「インテント」と言われる動作の枠組みを設定する。インテントとは、「ユーザーの問いかけとソフトウェア側の処理をマッピングする仕組み」のことである[17]。例えば、「今日の天気は？」と発話された場合にはユーザーが今日の天気を知りたいというとして今日の天気の情報伝える動作をするというように、「天気情報」としてインテントを作成する。インテント内部で、ユーザーが発話する文言をトレーニングフレーズとして登録し、返答する文言をレスポンスとして登録する。またユーザーからシステムが抽出したい日時、場所などのキーワードを「エンティティ」として、あらかじめ開発者が指定することができる。さらに、エンティティ内にキーワードを言い間違えた場合や誤字の入力を受け取ってしまった場合の文言を登録することで正確にキーワードを抽出できる。ユーザーが行いたい行為を「インテント」から認識し、「エンティティ」によってキーワードを抽出してサーバー側に渡すというのが VUI の一連の流れとなる。

(2) 対話モデル設計

ユーザーの VUI 使用時における対話ユースケースをモデル化することが重要となっている。対話モデルの設計を、ユーザーの発話バリエーション、使用時の状況などから対話モデルの設計を行うことでインテントやエンティティを適切に自然言語処理システムと結びつけることができる。

4. 本研究の手法

この章では、選択入力が続く状況の GUI アプリケーションを元に「ユーザーが GUI の複数選択入力を実現したい 1 つの行為」を元に VUI を生成し、GUI を付与する手法を述べる。

4.1 GUI に VUI を付与するまでの流れ

図 1 に、GUI アプリケーションに VUI を付与するまでのプロセスと役割を示す。アプリケーションの開発者は、選択入力が続く状況の GUI アプリケーションに本手法を用いて VUI を生成し、その後 GUI アプリケーションに VUI を付与するという流れとなる。

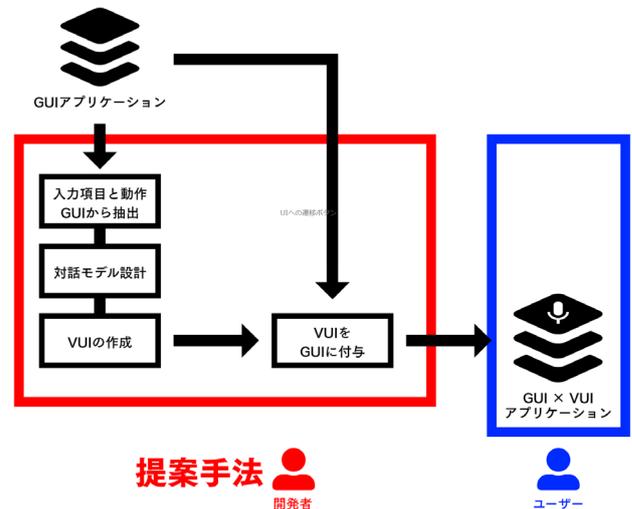


図 1 GUI に VUI を付与するまでの流れ
 Figure1 The flow of adding VUI to GUI

4.2 対象の GUI アプリケーション

本手法の適用対象の GUI アプリケーションは、選択入力が続く状況のものであり、かつ入力項目を一文で表現できるものが望ましい。例えば、乗車駅や降車駅といった 2 つの場所と時間のみを選択する乗換案内アプリケーションなどである。以降は本手法の説明を、乗換案内アプリケーションを用いて行う。また図 2 に手法を適用する前の乗換案内アプリケーション GUI、図 3 に本手法を適用した場合における UI の遷移図を示す。



図 2 乗換案内アプリケーション GUI
 Figure2 GUI for transit guide application



図3 乗換案内アプリケーション UI 遷移図

Figure3 UI transition diagram of transit guide application

4.3 VUI の生成

対象アプリケーションの GUI を元に、VUI を作成する。また本節での VUI についての例示は Dialogflow[16]を利用した場合とする。

(1) GUI 上での入力項目の抽出

GUI から、「入力項目」の要素を開発者が抽出する。選択入力を行う画面の GUI 上でプルダウンボタン、選択肢を提示や、入力をするためのカードダイアログを表示するためのボタンなどの入力に使う UI コンポーネント上に表示されているテキストを「入力項目」とする。例えば図3の様に乗換案内アプリケーションを考えた場合のトップ画面には「乗車駅、降車駅」を選択するプルダウンボタンと時刻入力用 UI ダイアログを表示させるための「乗車時刻、降車時刻」と記述されたボタンがあるため、「入力項目」は「乗車駅」「降車駅」「乗車時間、降車時間」とする。

(2) GUI 上での動作の定義

GUI 上で入力項目を選択入力した後にユーザーが最終的に行う「動作」を開発者が抽出する。選択入力を行う画面の GUI 上で、画面遷移をするボタンに表示されているテキストを「動作」とする。例えば乗換案内アプリケーションを考えた場合、選択した入力項目を元に検索した結果を画面遷移して別アクティビティ上で表示させるための「検索する」と記述されたボタンがあるため、「動作」は「検索する」とする。

(3) 対話モデル設計

「入力項目」と「動作」から、ユーザーがその画面上で何をしたいかがわかるので、開発者が対話モデル設計を行う。ユーザーが GUI 上で行っていた行為である「入力項目」と「動作」を、自然言語で言い換えられる一文を対話モデルとして作成する。例えば乗換案内アプリケーションで、入力項目が「乗車駅、降車駅、乗車時間 or 降車時間」で、動作が「検索する」だった場合は「乗車駅から降車駅に何時に着きたい、乗りたい」というように VUI 上での操作文言の作成を行う。この時、開発者が実際に発話してみて違

和感がないか、また微妙な言い間違えなども考慮し、作成した文に対し数パターンの入力パターンを作成する。Dasgupta によると 1 つのユーザーが行いたい行為、つまり Intent につき 30 パターンほどの想定が推奨されている[3]。なお出力に関しては、GUI 上で出力結果を表示しながら内容を読み上げる形を取るためパターン作成は本研究では行わないが、ユーザーと VUI 間のやり取りが複数回であることが前提の VUI を構築する場合には必要となる。

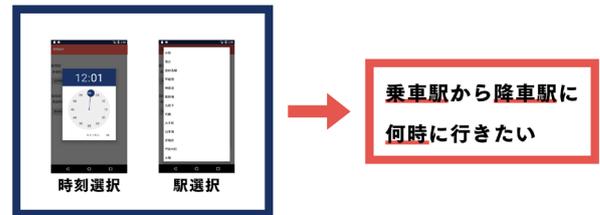


図4 対話モデル設計

Figure4 Conversation model design

(4) Intent とエンティティの作成 (VUI の作成)

自然言語処理を行うために開発者が VUI を構成する Intent とエンティティを作成し、対話モデルと自然言語処理を紐づける[13]。まずユーザーの行いたい行為 1 つに対して Intent を作成する。次に発話時のキーワードとなる入力項目をそれぞれエンティティとして登録する。そして Intent 内部のトレーニングフレーズに対話モデル設計で作成した入力パターン全てを、レスポンスに GUI での出入結果の文言を登録する。その後発話テストによる動作確認が完了すれば VUI の生成が完了する。例えば乗換アプリケーションであれば、「A 駅から B 駅に何時に着きたい」と言えば、「A 駅、B 駅、何時、着きたい」というキーワードを自然言語処理で抽出することで GUI 上の検索と同じレスポンスを返すということになる。なおレスポンスとして、エンティティで受け取った情報を元に動的な処理を行う場合はそのロジックを記述する必要があるが本研究は UI のみを対象としているため、本研究では対象としない。

4.4 GUI への VUI の付与

生成した VUI を開発者が GUI アプリケーションに組み込むことでユーザーが提案手法 UI によるアプリケーションを操作できる状態にする。まず元の GUI アプリケーション上に GUI から VUI へ切り替えるスイッチ、および VUI 入力を行う遷移先の GUI を作成する。GUI から VUI へ切り替えられることが重要であるため、実装手段は何でもよい。最後に遷移先画面に VUI を組み込み、応答に対して GUI 表示を実装する。

5. 評価実験

本手法の効果を確認するために、本手法の UI を搭載したアプリケーションを作成し、被験者による評価実験を行った。本手法の効果を確認するために、本手法の UI を搭載したアプリケーションを作成し、被験者による評価実験を行った。

5.1 実験準備

実験のために、本手法が適用できるシナリオを2つ用意し、簡易的なモバイルアプリケーションをそれぞれ作成した。アプリケーションを起動すると VUI 操作に切り替え可能な FAB ボタンを伴った GUI が表示される仕組みとなっている。被験者にはアプリケーションによるタスク処理とアンケート回答を行ってもらった。

(1) 実験用シナリオ

実験に利用する2つのシナリオを表1に示す。シナリオ A は大学のワーキングスペース予約のアプリケーションである。入力として、入力項目を部屋タイプ、コマ数、人数、何限からかの4項目とした。シナリオ B は乗換案内のアプリケーションである。入力項目を乗車駅、降車駅、乗車 or 降車時間の3項目とした。また、シナリオ A においては座りながら、シナリオ B においてはアプリケーション的に歩いている状態が想定されやすいため、歩きながら行うものとした。

表1 実験用シナリオ
 Table1 Experimental scenario

シナリオの種類	シナリオ A	シナリオ B
アプリの目的	ワーキングスペース予約	電車乗換案内
入力姿勢	座り	歩きながら
入力項目	部屋タイプ、コマ数、人数、何限	乗車駅、降車駅、乗車 or 降車時間

(2) 実験用モバイルアプリケーション

シナリオに沿った GUI を OS が Android7.0、言語は java で作成した。また VUI は Google の Dialogflow を使い、実装を行った。今回は実験用であるため、誤動作を防ぐためにタスクごとにインテントを作成した。VUI を作成した後に、GUI アプリケーションへの組み込みを行った。アプリケーションを開くと、選択入力が連続し、VUI への遷移ボタンを伴った GUI が表示される。なおシナリオ A の場合はトップ画面上の選択入力を行う UI コンポーネントは全てプルダウンボタンに、シナリオ B の場合はプルダウンボタンと時刻入力用のダイアログを表示するボタンとなっている。

5.2 実験の手順

実験は被験者に GUI と VUI とは何かという説明をした後に、「事前アンケート」「アプリケーションを使用した6つのタスク」「事後アンケート」と3段階に分けて行った。

(1) 事前アンケート

被験者に年齢や性別、VUI の使用経験や歩きスマホなどについてのアンケートに回答してもらった。

(2) アプリケーションを使用した6つのタスク

被験者には、以下の手順に従って、各タスクをこなしてもらった。表2にタスク一覧を示す。タスク1~3をシナリオ A で、タスク4~6をシナリオ B で行ってもらった。

- 1 アプリケーションを開く
- 2 図5.1の画面か図の画面で入力を行う
- 3 図5.2か図の画面で出力結果を確認する

表2 タスク一覧

Table2 Task List

タスク番号	タスク内容
1	3限から2コマ、5人でソファ席を画面入力で予約してください
2	5限から1コマ、2人で会議室を音声入力で予約してください
3	2限から1コマ、4人で会議室を予約してください
4	歩きながら、高田馬場駅から中野駅に11:30に着く電車を画面入力で調べてください
5	歩きながら、12時10分に中野駅から早稲田駅へ向かうための電車を音声入力で調べてください
6	歩きながら、早稲田駅から落合駅に12時30に着くための電車を調べてください

(3) 事後アンケート

実験用アプリケーションでタスクをこなしてもらった後に被験者にタスクについての評価をしてもらった。表3にアンケート一覧を示す。質問事項は UI の自由選択するタスクで GUI 入力と VUI 入力のどちらを選択したかとその理由、System Usability Scale[18]を参考とした提案システム自体の使いやすさに関するものとした。また、感想や意見などがあればそれを最後に記入してもらった。

5.3 実験結果

入力 UI を自由選択するタスクに関するアンケート結果を図3と表3.4に示す。被験者数は15人で、20代前半の男女に実験を行ってもらった。

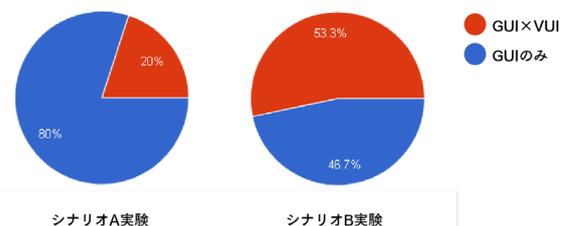


図5 どちらのUIで入力を行ったかのグラフ
 Figure5 Transit guide application UI for experiments

表3 選択した理由一覧（シナリオ A 実験）

Table3 Reasons for selection (experiment A)

入力方法	理由
GUI	音声入力の方がうまく入力できなくて失敗しそうだから
GUI	なれてるから
GUI	いまいち VUI の使い方が分からなかったから
GUI	テキストを見ながら入力する時に GUI の方が慣れているから
GUI	操作に慣れているため
GUI	GUI に慣れているから
GUI	慣れているから
GUI	なんとなく
GUI	GUI の方が普段から慣れているから
GUI	特に理由がないが、GUI 入力に慣れているため
GUI	タップしたほうが早いと感じたから
GUI	使い慣れてるから早い
GUI×VUI	打つのよりもまちがいが少なそう、楽だったから
GUI×VUI	早いし楽であるから
GUI×VUI	入力時間が短そうだったから

表4 選択した理由一覧（シナリオ B 実験）

Table4 Reasons for selection (experiment B)

入力方法	理由
GUI	入力ミスリスクが高いから
GUI	慣れてるから、しゃべりながら歩くのが恥ずかしいから
GUI	歩きとテキスト読み、同時並行の操作が多いので慣れてる方が楽
GUI	操作に慣れているため。
GUI	慣れているから
GUI	慣れているから
GUI	障害物が少ない道だったため GUI でも問題ないと考えた
GUI×VUI	音声入力に慣れてきたから
GUI×VUI	早いし楽
GUI×VUI	歩きながらだと入力しづらいので VUI を選択しました
GUI×VUI	歩きながらだと指での操作が困難、VUI は声で操作できるので楽
GUI×VUI	歩きながらだから
GUI×VUI	歩きスマホは危ないから。
GUI×VUI	歩きながら指示の紙と画面の両方を見るのが面倒だったから
GUI×VUI	歩きながら打つのは大変だから

5.4 考察

選択入力による入力が連続する場合、GUI では選択項目ごとにモードが発生してしまい入力手順が多くなるため、GUI を VUI に付与する方法を提案することでユーザーの入力手順が減り、ユーザビリティが向上するという参考文献を元にした仮説検証を目的に本研究を行った。

UI を自由選択するタスクでは、図 5 からわかる様にシナリオ A による「座り」のタスクでは GUI 入力を選択する被験者が 80%、VUI 入力を選択した被験者は 20% と GUI 入力を選択する被験者が大幅に多かったが、シナリオ B による「歩きながら」のタスクでは GUI を選択する被験者が 46.7%、VUI 入力を選択する被験者が 53.3% と僅かではあるが VUI 入力を選択する被験者が多いという結果となった。また System Usability Scale を元にしたユーザビリティにまつわる質問から VUI 入力に GUI 入力と比べシステム

としての障害は見られなかった。

純粋な入力手順数の上では VUI の方が少ないにも関わらず、シナリオ A では GUI 入力が多く、シナリオ B では VUI 入力が入力回数 A よりも多くなった理由を考察する。GUI 入力を選択した被験者の理由を見ると、シナリオ A では VUI よりも GUI に慣れている、タップしたほうが早いと思ったからという 2 つの理由が主となっていて、シナリオ B では VUI よりも GUI に慣れている、恥ずかしいからという 2 つの主な理由が見られた。シナリオ B では、実際に非被験者がいる室外にて実験を行ったために、恥ずかしいからという動機が増えた。ここで注目したいのが、シナリオ A 実験では GUI に慣れているため GUI 入力の方が早いという動機があったが、シナリオ B 実験では見られないということである。このことから、シナリオ B 実験では被験者が GUI の方を使い慣れていたとしても VUI の方が入力完了までの時間が短いという条件で実験ができたことがわかる。また、シナリオ A 実験で速度を理由に GUI 入力を選択した被験者が全員シナリオ B 実験では VUI 入力を選択している事実からも確認ができる。一方で VUI 入力を選択した被験者の理由を見ると、シナリオ A ではうつのが面倒、入力時間が短いという理由で、シナリオ B では入力時間が短い、音声入力に慣れたから、歩きながらだと指操作より楽といった理由が見られた。いずれも入力手順数が少ないことに起因する理由である。これらのことからシナリオ A では GUI 入力が多く、シナリオ B では VUI 入力が入力回数 A よりも多くなった理由は、歩行という要素が加わり、ユーザーが処理しなければならない認知負荷が増すため、慣れている GUI 入力を行うことよりも手順数の少なさとハンズフリーにおいて優位がある VUI 入力を選択する傾向があるためと考えられる。以上の UI を自由選択するタスクの考察から、GUI の複数選択入力部分を VUI にすることで入力手順数を減らす本手法による試みはユーザビリティに対して有用性があると言える。

6. おわりに

本研究では、ユーザーの目的が入力する前に時間、場所などの入力項目が明確であるにもかかわらず、GUI 上で複数の項目から何かを選択する入力を複数回連続して行わなければならないという問題に対して、「ユーザーが GUI の複数選択入力を実現したい 1 つの行為」を元に VUI を生成し、GUI を付与する手法を提案した。この提案手法の効果を確認するために、モバイルアプリケーションを作成した。被験者にモバイルアプリケーションを用いてタスクを座りながらと歩きながらでそれぞれ 3 つずつ、合計 6 つのタスクを行ってもらった。タスクを行った後、UI を自由選択するタスクで GUI 入力と VUI 入力のどちらを選んだかとその理由をアンケートに記述してもらうことで評価してもらった。その結果、GUI の選択入力部分を VUI にすること

で入力手順数を減らす試みはユーザビリティに対して有用性があることがわかった。

さらにこの手法について研究を進める場合、どれほどの情報量を持つ GUI 選択入力なら VUI 入力に要約すると有用性が生まれるのかといった評価の研究が考えられる。また、被験者のコメントから「VUI を表示するための GUI デザインとしてチャット画面は最適なのか」、「被験者によっては一文の形で音声入力するのではなく個別の単語連呼する自由構文入力の方が VUI として使いやすい」なども今後の研究対象となり得る。また、Google Home を日常的に使用している被験者が VUI の音声入力の認識精度を考慮して UI を選択するタスクで全て GUI を選んでしまうといったことが起きた。これは先行研究で触れられていた、VUI の使用経験がある人は新しく扱う VUI に対して、かつて使用した VUI と同様の扱いをする傾向があること同様な現象と言える[4]。

また、今後の課題として以下のことが挙げられる。

- どれほどの情報量を持つ GUI 選択入力なら VUI 入力に要約すると有用性が生まれるのかの評価
- VUI を表示するための GUI デザインとしてチャット画面は最適なのか
- VUI 入力コマンドとしての自由構文の評価

参考文献

- [1]Stefan Schaffer, Robert Schleicher, Sebastian Möller, Modeling input modality choice in mobile Graphical and speech interfaces, Int. J. Human-Computer Studies, 2015, p.21-34.
- [2]Alex Sciuto, Armita Saini, Jodi Forlizzi, Jason I. Hong, "Hey Alexa, What's Up?": Studies of In-Home Conversational Agent Usage, DIS 2018.
- [3]Ritwik Dasgupta, Voice User Interface Design Moving from GUI to Mixed Modal Interaction, Apress, p.13-37, 2018.
- [4]Chelsea M. Myers, Anushay Furqan, Jichen Zhu, The Impact of User Characteristics and Preferences on Performance with an Unfamiliar Voice User Interface, CHI 2019.
- [5]Christine Murad, Leigh Clark, Cosmin Munteanu, Benjamin R. Cowan, Design Guidelines for Hands-Free Speech Interaction, MobileHCI'18.
- [6]Jakob Nielsen, 1994. Enhancing the explanatory power of usability heuristics. Proc, CHI 1994.
- [7]Donald Norman, The design of everyday things, BASIC BOOKS, 2013, p.71-73.
- [8]Ben Shneiderman, Catherine Plaisant, Designing the User Interface: Strategies for Effective Human-Computer Interaction, p.60-66 Pearson Education, 2005.
- [9]Christine Murad, Leigh Clark, Cosmin Munteanu, Benjamin R. Cowan, Revolution or Evolution? Speech Interaction and HCI Design Guidelines, IEEE Computer Society, 2019.
- [10]Siri - Apple ,<https://www.apple.com/siri/>, 2020/04/12 閲覧.
- [11]Cortana とは,
<https://support.microsoft.com/ja-jp/help/17214/cortana-what-is>, 2020/04/12 閲覧.
- [12]Amazon Alexa Official Site: What is Alexa?,
<https://developer.amazon.com/ja-JP/alexa>, 2020/04/12 閲覧.
- [13]Google アシスタント - あなたの Google,
https://assistant.google.com/intl/ja_jp/, 2020/04/12 閲覧.
- [14]Material Design, <https://material.io/>, 2020/01/31 閲覧.
- [15]アクセシビリティ 2018 年 デジタル消費者調査,
https://www.accenture.com/t00010101T000000Z_w_/jp-ja/_acn-media/PDF-75/Accenture-ACNJPN-FIX.pdf, 2020/01/31 閲覧.
- [16]Dialogflow, <https://dialogflow.com/>, 2020/01/31 閲覧.
- [17]里山南人, 一円真治, Google アシスタントアプリ開発入門, ソシム株式会社, 2018, p.36-74.
- [18]System Usability Scale (SUS) | Usability.gov,
<https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>, 2020/04/12 閲覧.
- [19]Google Home - スマートスピーカー-Google ストア - Google Store,
https://store.google.com/jp/product/google_home, 2020/04/12 閲覧.