

HapTalker : 視覚障害者のための電子書籍向けユーザインタフェースの提案

中井 峻日* 伊藤 精英† 柳 英克† 美馬 義亮†

概要. 本研究の目的は弱視および全盲の障害者が日常的に、不自由なく電子書籍を利用できる環境の構築である。電子書籍はテキスト読み上げ機能を利用することで視覚障害者も扱うことが可能である。その為、電子書籍は視覚障害者にとって情報を得るのに重要な役割を果たしている。しかし、視覚障害者が晴眼者のように書籍を扱う場面では困難を伴う場合がある。その例として、瞬時に読みたいページに移動する、重要な文章にマーカをつけるなどの行為が挙げられる。視覚障害者の読書環境を改善するため、視覚障害者向けタッチスクリーンデバイス用電子書籍リーダー「HapTalker」の設計をおこなった。HapTalkerを設計する過程で視覚障害者にとって扱いやすいユーザインタフェースを探った。ユーザインタビューの結果、HapTalkerにはユーザにとって一定の有用性があることが分かった。本稿では、設計したHapTalkerの機能およびそれに関連したUIについて述べる。加えて、実施したユーザインタビューの結果を報告する。

1 はじめに

本研究の目的は視覚障害者の読書環境を改善することである。その背景には、KindleやiBooksといった電子書籍リーダーの出現による電子書籍の普及がある[1]。電子書籍はテキスト読み上げ機能を利用することで視覚障害者でも扱うことが可能となった。その為、電子書籍は視覚障害者にとって情報を得るのに重要な役割を果たしている。しかし、現在の読書環境において視覚障害者には晴眼者に提供されているものと同等の利便性が提供されているとは言えない。例として、瞬時に読みたいページを探る、重要な文章にマーカをつけるなどの操作が挙げられる。視覚障害者の読書環境を改善するため、視覚障害者向けタッチスクリーンデバイス用電子書籍リーダー「HapTalker」の設計をおこなった。HapTalkerはタッチスクリーンデバイス上で動作するアプリケーションである。HapTalkerを設計する過程で視覚障害者にとって扱いやすいユーザインタフェース（以下UI）を探った。HapTalkerの設計方針は、晴眼者が行っている電子書籍の操作を分析し、音声や振動で表現することである。これにより、視覚障害者は電子書籍を容易に操作することができる。

本稿では設計したHapTalkerに導入された視覚障害者向けの機能およびそれに関連したUIについて述べる。加えて、実施したユーザインタビューの結果を報告する。

2 関連事例

2.1 既存の電子書籍リーダー

市販の電子書籍リーダーとしてはKindleやiBooksなどがある。視覚障害者の多くは、VoiceOverなどのテキスト読み上げ機能を利用することで、電子書籍を読むことができる。これらの電子書籍は、録音された朗読のような音声ではなく、テキストベースのコンテンツであるため制作が容易である。現在、これらのリーダーにおいては、スクリーン上にある仮想のボタンに対してキーやマウス、指やスタイラスを用いて操作を行うUIが一般的である。しかし、視覚障害者はスクリーン上にあるボタンの形状、大きさ、位置を視覚で認識することが困難である。その為、視覚障害者は操作前に操作対象であるボタンの位置や内容を手探りで把握する必要がある。故に、作業目的を達成するまでのステップが晴眼者に比べて多い[5]。そして、既存の電子書籍リーダーにも、こういったUIが用いられている。

タッチスクリーン上において、ボタンを扱わず、触覚や聴覚を用いた出力、入力を主とした視覚非依存のUIを構築することが、視覚障害者の読書環境を改善することに繋がると考えた。本研究で設計するHapTalkerは、スクリーンのタッチ開始位置に依存せず操作できるUIを扱う。

2.2 視覚障害者向け文字入力方式に関する研究

スクリーンのタッチ開始位置に依存しない視覚障害者向け文字入力方式に関する研究は幾つかある。例えば、青木ら(2013)[3]はタッチスクリーンを用いた視覚障害者向け文字入力方式を提案している。この文字入力方式は指がタッチスクリーンに触れた任意の位置から指を離さずにスライドさせることで文字入力を可能とするものである。この文字入力方式は実験により、その有用性が示されている。この方

Copyright is held by the author(s).

* 公立はこだて未来大学大学院 システム情報科学研究科

† 公立はこだて未来大学 システム情報科学部

式は視覚非依存的ではあるが、フリック入力を扱っている。フリック入力は情報量が多いテキストを扱うのに向いていないため、電子書籍を扱う UI に応用することが難しい。また、井川ら (2013)[2], 大橋ら (2015)[4] においても視覚非依存的な文字入力方式が提案されている。しかし、何れも電子書籍を扱う UI に応用できるとは考えにくい。

3 HapTalker の設計

3.1 概要

HapTalker は iOS のタッチスクリーンデバイス上で動作するアプリケーションである。以下では HapTalker の設計コンセプトを述べる。

対象ユーザ

後天的に障害を負った視覚障害者を HapTalker の対象ユーザとした。視覚を全く使うことがなくとも扱うことが可能な UI を目指すこととした。

操作開始位置に依存しないフィードバック

視覚障害者が扱いやすいよう聴覚や触覚を利用した視覚非依存の UI を構築した。一般的な UI はスクリーン上の特定の位置に存在するボタンに対して操作を行う。しかし、このような従来の UI では視覚を用いずに瞬時にボタンの位置を特定するのが困難なため、視覚障害者にとって扱いづらい。この問題に着眼し、HapTalker はスクリーンのどの位置から操作を開始しても、同様のフィードバックが得られる UI となることを目指した。

なにも表示しないスクリーン

HapTalker は全盲の視覚障害者を対象ユーザとしている。対象ユーザはスクリーン内に表示された文字や図を視覚によって認識することが出来ない。そのため、HapTalker はスクリーンに視覚的な情報を表示することを前提としない設計にした。

ページの排除

内容にアクセスする手段として、ページ概念を取り扱わないこととした。スクリーン上に文字を表示する場合、表示できる文字の数には限りがある。そのため、従来は書籍に記された文章をページという単位に切り分けて表示していたと考えた。スクリーン上に文字を表示しないことから、ページは不要である。ページには文章にアクセスしやすくなるという長所がある。しかし、どんな内容が書かれているかをページ数から予測することは困難である。ページよりも、章、節、段落、行などの文章の内容に沿って区切った単位の方が、内容の特定が容易である。故に、HapTalker ではページを扱わず、文章の内容に沿って区切った単位を中心に扱う。

3.2 開発環境

HapTalker の開発には、iOS アプリケーション開発用のプログラミング言語である Swift4 を用いた。

3.3 実装内容

HapTalker を実現するために、考案した GUI におけるコントロールに対応する機能のプロトタイプを制作した。プロトタイプの機能および UI を以下に示す。用意した 3 つの UI をそれぞれ、直線型、ダイヤル型、ドアノブ型と名付けた。

直線型

ドラッグとタップを利用した操作方法を用意した。ドラッグとタップは、タッチスクリーンデバイスにおいて頻繁に扱われるタッチジェスチャであることから操作難易度が低く扱いやすいと判断した。

直線型には小説、専門書など書籍の種類に囚われずに利用されるであろう電子書籍における操作の基本となる機能を設計、実装した。HapTalker に実装した機能は以下の通りである。

- 読み上げる文章を行単位で選択する
- 文章の読み上げ開始時に現在、選択している文章が何行目なのかを音声により通知する
- 文章を現在選択している行から自動で昇順に読み上げる・読み上げを止める
- 1 行目より前などの、文章がないところを選択しようすると警告音を出す
- 読み上げる速度を変更する
- 行単位の文章に付箋をつける・はずす
- 付箋のついた文章のみを選択する

本研究における付箋とは、後に読み返したい文章や気になった文章に行単位でつける目印を意味する。付箋は複数の行につけることが可能であり、付箋の有無は振動で検知可能である。読み上げる文章を選択する際に付箋がついた文章を見つけると iPhone 本体が振動する。振動を採用した理由としては文章の読み上げを阻害することなく瞬時に付箋の有無を確認できるためである。

ドラッグを利用した操作方法と機能の対応づけは以下の通りである。

- ドラッグ開始時の指の位置と、現在ドラッグしている指の位置における Y 座標の差が 17.5mm 増えるごとに読み上げる行を選択する
 - － 下向きにドラッグすると読み上げる文章を昇順に選択する (図 1)
 - － 上向きにドラッグすると読み上げる文章を降順に選択する

- － 1 行目未満など、文章がないところを選択しようとする警告音を出す（図 2）
- － 文章選択時にドラッグを中断すると、次回は操作を中断した行から選択を再開する

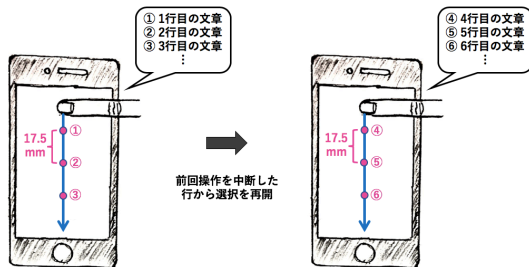


図 1. 下向きのドラッグで読み上げる文章を昇順に選択

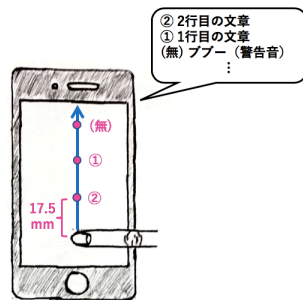


図 2. 文章がない場所を選択した場合に警告音を出す

- ドラッグ開始時の指の位置と、現在ドラッグしている指の位置における X 座標の差が、26.5mm を越えた場合に付箋の付け外しを行う
 - － 右向きにドラッグすると現在読み上げ中の文章に付箋がつく（図 3）
- 既に付箋がついている場合、スクリーンを左向きにドラッグすると現在読み上げ中の文章から付箋が外れる

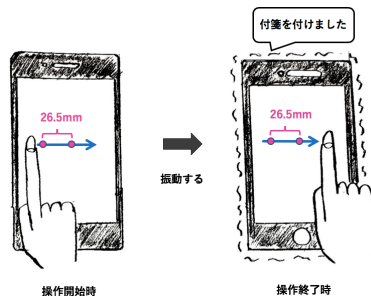


図 3. 右向きにドラッグすると振動の付箋を付ける

- 2 本の指でスクリーンに触れたまま上下にドラッグすると付箋がついた文章のみから読み上げさせるものを選択する（図 4）

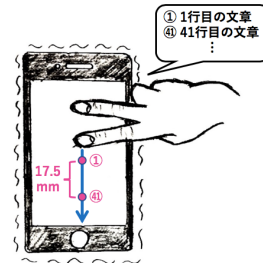


図 4. 2 本の指で下向きにドラッグすると付箋がついた文章のみから昇順に選択

- 2 本指でピンチイン・ピンチアウトをすると文章を読み上げる速度を変更する
 - － 指の間隔が 0.5mm 広がるごとに、読み上げる速度が速くなる（図 5）
 - － 指の間隔が 0.5mm 狭まるごとに、読み上げる速度が遅くなる

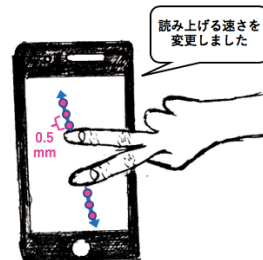


図 5. 指の幅が広がるにつれ読み上げる速度が速くなる

以下では HapTalker に設定したパラメータの選定理由について述べる。読み上げる行の選択間隔を 17.5mm に設定した理由は、スクリーンをドラッグする際のヒューマンエラーを防止するのに適切だと判断したことである。ここでのヒューマンエラーとは対象の行以外の行を誤って選択する行為を指す。ヒューマンエラーを防ぐには最低でもユーザの親指の横幅程度の間隔が必要だと考えた。

また、行を選択する操作に X 座標ではなく、Y 座標を採用した理由としては、iOS デバイスのスクリーンが縦に長いことが挙げられる。行を選択する操作に X 座標を用いた場合、最大の移動量が Y 座標を用いた場合に比べて小さくなる。このため、一度のドラッグで選択できる行数が多くなるように、X 座標ではなく Y 座標を採用した。

昇順に文章を移動するにあたり、下向きのドラッグを採用した目的は、晴眼者が紙媒体の書籍を読む感覚に近づけることにある。晴眼者が書籍を読む時、横書きの文章は上から下へ向かって読んでいく。このため、昇順の移動には下向きのドラッグを採用した。降順に文章を選択するにあたり、上向きのドラッグを採用した理由も同様である。

付箋を付け外す操作において、X座標の差を26.5mm以上とした目的は、誤動作を防ぐことである。例えば、付箋の付け外しの操作において、X座標の差を9mm以上と設定した場合、誤って指を右向きに少しドラッグした場合でも付箋の付け外し機能が動作してしまう。実装の中で、このような誤動作を防ぐために有効だと判断したため、26.5mmに設定した。

ピンチイン・ピンチアウトによって読み上げる速度を変更できるようにした理由は、指を広げるという動作が音量を大きくするという機能と操作感が一致すると考えたためである。また、変更間隔は、実装の中で適切だと判断したため0.5mmに設定した。

タップを利用した操作方法と機能の対応づけは以下の通りである。

- タップすると自動読み上げを開始する
- 自動読み上げを既に実行している場合にスクリーンをタップすると自動読み上げが停止する

自動読み上げは現在選択している行から読み上げが始まる。自動読み上げが終わると終わった行が、次の操作時に読み上げる文章として選択される。

ダイヤル型

ダイヤル型はスクリーン上に円を描くという動作を扱う。円を描くという動作は他の操作との共存が難しい。そのため、合成音声に読み上げさせる文章を選択する機能のみを取り扱うことにした。スクリーン上で円を描くことにより文章を選択する。円を描くという動作はダイヤルを回す動作やカセットテープを指で巻き戻す動作を連想させると考える。ユーザの経験、既有知識を利用するとメンタルモデルと操作感が一致しやすく操作方法が記憶しやすいのではないかと予想した。

機能と操作方法の具体的な対応付けは以下の通りである。

- 時計回りに円を描くと20°ごとに現在選択している文章の次の1行を読み上げる文章として選択する(図6)
- 反時計回りに円を描くと20°ごとに現在選択している文章の前の1行を読み上げる文章として選択する

なお、ダイヤル型の操作は、スクリーンの中央を囲うように円を描く場合に動作する。操作を中断した場合、次回は操作を中断した行から読み上げる文章の選択を再開する。

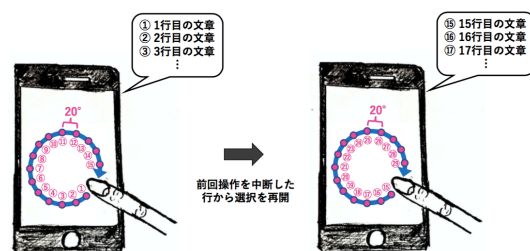


図 6. 時計回りに円を描くと20°ごとに一行進む

ドアノブ型

ドアノブ型ではiPhone本体をドアノブに見立て動作を行う。ドアノブは引く、押す、回転させるなどの使い方があり、使い方は誰もが知っていると考えた。ダイヤル型と同様に、ユーザの既有知識を利用することで、操作方法がユーザにとって記憶しやすく扱いやすいものになると考えた。

機能と操作方法の具体的な対応付けは以下の通りである。

- iPhone本体を時計回りに30°以上回転させると、次の1行を読み上げる文章として選択する(図7)



図 7. iPhone本体を時計回りに30°以上回転させると次の1行を選択する

- iPhone本体を反時計回りに30°以上回転させると、前の1行を選択する(図8)



図 8. iPhone本体を反時計回りに30°以上回転させると前の1行を選択する

- iPhone 本体を手前に引く動作をすると文章に付箋がつく（図 9）
- 付箋がついている時に、iPhone 本体を手前に引く動作をすると、現在選択している文章から付箋が外れる

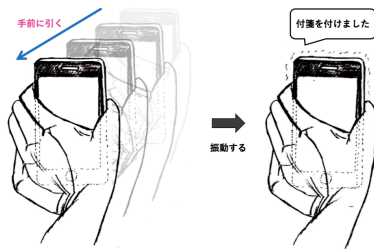


図 9. iPhone 本体を手前に引くと付箋を付け外す

4 視覚障害者によるフィードバック

インタビューによる HapTalker の評価実験をおこなった。ここでは、全盲の視覚障害者をインタビュー対象に選んだ。

4.1 インタビューの概要

本インタビューの目的は HapTalker の機能および操作方法が全盲の視覚障害者にとって有用ものであるかどうかを質的に評価することである。インタビュー対象は HapTalker の対象ユーザである後天的全盲視覚障害者とした。インタビューは 2017 年 12 月 16 日に函館市内に在住する 2 名に対して、グループ形式で行った。HapTalker を試用してもらいフィードバックを得た。

4.2 インタビューによる評価結果

HapTalker のコンセプト、機能、操作方法に関して以下のフィードバックを得た。

コンセプト

「ページの概念がないことについてはどう思うか？」という問いに対して「ページの概念はあった方がよい」というコメントが寄せられた。理由としては以下が挙げた。

- 読んだ内容や読んでいる状況を他人に説明しやすいから
- 書籍のイメージが湧きやすく、HapTalker の構造が理解しやすいから
- 書籍を読んでいる感覚に浸りやすいから

「ボタン指向ではなくスクリーン全体に対してタッチジェスチャで操作を行うことに対してはどう思うか？」という問いに対しては「ボタン指向の操

作と比較すると扱いやすい」というコメントが寄せられた。

操作方法

直線型については「操作がシンプルで扱いやすい」というコメントが寄せられた。加えて、無駄な機能がなく、シンプルであることが評価された。

振動で表現された付箋については「合成音声による読み上げを阻害しないので好印象」というコメントが寄せられた。合成音声による読み上げの速度を変更する機能については「読み上げる速度を変更すると文章の読み上げが中断されないようにしてほしい」というコメントが寄せられた。

ダイヤル型については以下のコメントが寄せられた。

- 直線型と比較すると操作が複雑である
- スクリーンのどの位置に円を描いても反応するようにしてほしい

複雑性が高いという評価を受けた要因として、円を描くというタッチジェスチャがタッチスクリーンデバイスにおいて一般的ではないということが挙げられる。このことから、タッチスクリーンデバイスの操作において、一般に用いられていないジェスチャに関しては、複雑性が高いという認識を持って、機能と対応付ける必要がある。

ドアノブ型については以下のコメントが寄せられた。

- iPhone 本体を動かして操作するのが楽しい
- メインの操作方法ではなくオプションの操作方法という位置づけが相応しい
- 操作中の様子を他の人に見られると怪しまれそう

コメントからドアノブ型は人に見られてない状況での使用に向いており、かつオプションとしての位置づけが相応しいと考えた。また、ユーザインタビューでは「腕が疲れる」という意見が寄せられた。腕が疲れないように小さな動きで移動ができる構造に改善することが必要であると考えた。

5 議論と今後の課題

本研究では視覚障害者の読書環境を改善するため、視覚障害者向けタッチスクリーンデバイス上の電子書籍リーダー「HapTalker」の設計をおこなった。HapTalker を設計する過程で視覚障害者にとって扱いやすい UI を探った。ユーザインタビューの結果、HapTalker がユーザにとって一定の有用性があることが分かった。

スクリーンからの視覚的フィードバックを前提としない UI は、既存の UI の概念を拡張しうるものである。情報端末の操作入力における UI が従来の触

覚を中心としたものから、触覚の乏しいフラットなものへと切り替わっている。入力装置のフラット化により視覚障害者は一度に複数のボタンの位置を認識できず操作が困難な状態になった。この問題を解決するためには、タッチスクリーンデバイスの UI に、視覚障害者向けの合理的な枠組みが必要であると考えられる。本研究はタッチスクリーンデバイスにおける枠組みの構築に寄与する。本研究により、電子書籍リーダーのみならずフラットな入力装置における、視覚障害者向け操作入力およびフィードバックの枠組みに、目が向けられると期待する。

多角的な評価をするため、今後は操作の効率性、エラー発生率、操作方法の学習しやすさ、記憶しやすさ、主観的満足度などのユーザビリティを検証する。また、Kindle などの既存の電子書籍リーダーと読書に要する時間を比較し、HapTalker のアクセシビリティを検証することが必要である。

また、本研究ではドアノブ型を取り上げたが、傾けたり振ったりなど装置本体を動かす UI について、未だ有用性が十分に検証されていない。今後の研究ではこういった形式の UI の有用性について検証する予定である。操作方法が記憶しやすい、肢体が不自由なユーザにも利用できるなどの利点が期待できる。本研究を起点として今後、装置本体を動かす UI が世の中に広く浸透することを期待する。

本研究で提案した HapTalker は、対象とするユーザを広げていくことにより、より多くの人々にとって意義のあるものへ発展させることができる。考えられる将来の対象ユーザとして、後天または先天的に弱視となった視覚障害者、晴眼者などが挙げられる。今後もインタビューを行い、ユーザの特性に合わせて HapTalker を再設計していく。

参考文献

- [1] インプレス総合研究所. 電子書籍ビジネス調査報告書 2017. 調査報告書. インプレス, 2017.
- [2] 井川洋平, 宮下芳明. アイズフリーで速記できる「方向のみ」のフリック入力手法. インタラクシオン 2013 論文集, pp. 631–651, 2013.
- [3] 青木良輔, 橋本遼, 瀬古俊一, 片岡泰之, 井原雅行, 渡辺昌洋, 小林透. Drag&Flick: タッチスクリーンを用いた視覚障がい者向け文字入力方式. インタラクシオン 2013 論文集, pp. 72–79, 2013.
- [4] 大橋隆, 三浦貴大, 坂尻正次, 大西淳児, 小野東. タッチスクリーン端末を利用する視覚障害ユーザのための音声フィードバックを活用した入出力インタフェース. 情報科学技術フォーラム講演論文集, pp. 565–566, 2015.
- [5] 野口武悟, 中和正彦, 成松一郎, 植村八潮. 電子書籍のアクセシビリティに関する実証的研究 (II). 人文科学年報, pp. 187–199, 2013.

未来ビジョン

本研究では HapTalker を設計し、視覚障害者用の電子書籍向け UI を提案した。本研究で提案した UI は電子書籍のみならず、テキストデータを扱う様々なアプリに応用可能だと考える。その一例として、メモ用のアプリが挙げられる。実施したユーザインタビューの中で視覚障害者は PC のメモ用アプリを使って学習をおこなうことが分かった。視覚障害者が行っていた学習の手順は以下の通りである。

1. 録音図書や IC レコーダに録音した音声ベースの教材を聴く
2. 聴いた内容の中で深く理解したい内容や暗記したい内容を電子メモに書き連ねる
3. 書いたメモを合成音声に読み上げさせる
4. 読み上げさせた音声を聞き、他の知識と結びつけながら学習事項の構造を理解する

電子メモの他にも LINE 等の SNS や Twitter を利用していることがユーザインタビューで分かった。本研究により電子メモや SNS などのテキストデータを扱うアプリの利便性が向上することが期待できる。

また、本研究では、UI のみならずテキストを自然言語処理により章、節、段落、行などの単位に自動で構造化する仕組みを追求する。章や節は文章内の改行を検知して判断する。段落は文頭の空白、行は句点を検知して判断する。本研究で提案した UI と組み合わせることにより、章や節ごとに読み上げる電子書籍内の文章を選択することが容易になる。その結果、電子書籍に対するアクセシビリティが向上し視覚障害者の読書環境が改善されると期待できる。

この他にも、電子メモを HapTalker にインポートするための仕組みや、人々が効率よく学習を行うために必要な機能を追求する所存である。