

Media Overlays 機能を有する EPUB 3 電子書籍の制作と教育実践

Self-made EPUB 3 eBooks with Media Overlays and School activities

生田 茂

大妻女子大学人間生活文化研究所

Shigeru Ikuta

Institute of Human Culture Studies, Otsuma Women's University

12 Sanban-cho, Chiyoda-ku, Tokyo, 102-8357 Japan

キーワード：電子書籍，EPUB 3，Media Overlays，教育実践

Key words：eBooks, EPUB 3, Media Overlays, School activity

抄録

単語や文節をハイライトしながら同期をとって読み上げを行う Media Overlays 機能を有する EPUB 3 準拠の電子書籍の制作の方法について詳述するとともに、これらの電子書籍を用いた通常学校や特別支援学校、特別支援学級における教育実践について報告する。JavaScript を用いた音声の再生、HTML の button のクリックイベント機能を活用した音声の再生についても報告する。こうした Media Overlays 機能や音声の再生の機能を有する電子書籍は、手指や上肢が不自由で、自分では紙の教科書や絵本のページめくりができない児童生徒が EPUB 3 対応のリーダーの自動ページ送り機能を用いて購読できた事例、音読活動において読みの誤りや読み飛ばしの多い児童が電子書籍を使ってこうした欠点を改善できた事例、通常の教科書や副読本を読むことのできない弱視の児童が音声で副読本の学習を進めることができた事例などを生み出すことができた。本論文では、EPUB 3 準拠の「単語や文節の音声の再生を可能とする」電子書籍を制作する上でのポイントを解説するとともに、こうした電子書籍の教育における活用について論じる。

1. はじめに

2021年3月に「学校教育法第三十四条第二項に規定する教材の使用について定める件(平成30年文部科学省告示第237号)」の一部が改正され、「学習者用デジタル教科書を各教科等の授業時数の2分の1に満たないこと」としていた基準が見直されるとともに、「児童又は生徒の学習及び健康の状況の把握に特に意を用い、検定済教科用図書等を使用する授業と学習者用デジタル教科書を使用する授業を適切に組み合わせた教育課程を編成すること」とされた^[1]。

また、中央教育審議会は2022年8月25日の作業部会で、2024年度から小学5年生から中学3年生の英語で紙の教科書と併用する形で電子書籍の先行導入を行うことを発表、さらに英語における活用状況を踏まえて、2025年度以降に算数・数学に導入する考えを示した^[2]。

本論文では、こうした教育現場における電子書

籍をめぐる新しい動きを踏まえ、学校の教員が、無料のソフトウェアを用いて「文や文節をハイライトしながら同期をとって読み上げを行う」Media Overlays 機能^[3]、また、動画などを再生できる電子書籍を手作りする手法について詳述する。

電子書籍の国際規格は、2011年10月に発表された EPUB 3.0^[4] で日本語などを扱うのに必要なマルチバイト言語への対応や縦組み、縦中横、圏点、ルビ表記、禁則処理などの仕様が盛り込まれた。(最新の国際規格は EPUB 3.3^[5] である。)

電子書籍は、WEB ページの記述言語である HTML (最新のバージョンは HTML Living Standard^[6]) を用いて記述し、さらに、各ページのレイアウトやデザインを整える CSS (Cascading Style Sheets)^[7]、文や文節をハイライトしながら同期をとって読み上げを行う Media Overlays 機能を実現する SMIL (Synchronized Multimedia Integration Language)^[8] ファイルなどと共に EPUB 3

の仕様でコンテナに収め zip 形式^[9] で圧縮したファイルとなっている。

電子書籍を手作りするのは難しそうに思われるが、既存の電子書籍を「雛形」として用いることで、本論文で示すように、比較的簡単にオリジナルの電子書籍を作ることができる。

本論文では、Media Overlays 機能を有する電子書籍の制作について手順を追って詳述するとともに、これらの電子書籍を用いて、通常学校、特別支援学級、特別支援学校、日本語学校などで取り組まれた教育実践の成果や課題について論じる。

2. 方法

2.1. EPUB 3 電子書籍 の制作と購読

EPUB 3 に準拠した電子書籍を制作するには、EPUB 3 eBooks Editor と呼ばれるソフトウェアが必須となる。また、EPUB 3 準拠の電子書籍を読む（購読する）には、EPUB 3 eBooks Reader と呼ばれるソフトウェアが必須となる。

(1) EPUB 3 eBooks Editor

EPUB 3 に準拠する電子書籍は、W3C^[10] (World Wide Web Consortium) の国際規格に従って作成するように推奨されている。

2010 年代の初めには、日本のフューズネットワーク社が開発した FUSEe ベータ版^[11] が使い勝手に優れた無料の Editor として有名だったが、今は配布を中止したこともあり、Sigil^[12] によって代わられている。Sigil はメニューの日本語化もサポートしており、Windows OS, macOS の PC に対応した使い勝手の優れた EPUB 3 eBooks Editor として評価が高い。本論文でも、この Sigil を用いて EPUB 3 対応の電子書籍を手作りしている。

無料の EPUB 3 eBooks Editor として、他には The Daisy Consortium^[13] とマイクロソフトが開発している WordToEPUB^[14] がある。このソフトウェアを用いると、マイクロソフトのワードの文書を電子書籍に簡単に変換することができる。ただし、現在は、Windows OS の PC のみへの対応となっている。

ワード文書の章や節を「ホーム」タブの「見出し」スタイルで指定することで、章や節に対応した目次が自動的に生成された電子書籍に変換される。この WordToEPUB は、Daisy Consortium のサイトからダウンロードできる。（著者が WordToEPUB のメニューの日本語化を行なっている。）

現バージョンでは、「文や文節をハイライトし、同期をとって音読を行う」Media Overlays の機能を実装することはできないが、WordToEPUB で作成した電子書籍に、次のような操作を行うことで Daisy 図書ライクな Media Overlays 機能を追加することができる。

- ePub Zip/Unzip 3.0^[15] などのソフトウェアを用いて電子書籍を解凍する。
- Daisy Consortium が開発している Tobi^[16] を用いて、解凍されたフォルダーの content.opf^[5] ファイルを開く。
- Tobi で開かれた編集画面で、音声同期するテキストを選択し、PC の内蔵マイクロフォンを使って音読して音声を重ねる。
- 上記の作業を繰り返して必要なテキストに音声を重ね、最後に File メニューから Export して Media Overlays 機能の付加された電子書籍を制作する。

こうして Tobi を用いて制作された電子書籍は、後述する Thorium Reader^[17] を用いることで「文や文節をハイライトしながら同期をとって音声を読み上げる」ことができる。（ただし、アップルの Books リーダー^[18]では読み上げることができない。）

これらの EPUB 3 eBooks Editor の他に、有料ではあるが、日本のシナノケンシ株式会社が作成している PLEXTALK Producer^[19] というソフトウェアがある。読み込んだテキストを、自動で分節に区切り Media Overlays 機能を有する Daisy 図書に変換することができる。テキスト中の漢字に一斉にルビを振ることもでき、HTML や CSS の詳細を知らなくても Media Overlays 機能を有する電子書籍を作ることができる。この PLEXTALK Producer で制作した電子書籍は、同じくシナノケンシ株式会社が開発している Daisy 再生アプリの「イーリーダー^[20]」で購読することができる。PLEXTALK Producer は Windows OS 用のソフトウェアとなっている。

(2) EPUB 3 eBooks Reader

無料の EPUB 3 eBooks Reader として、アップルが iPad, iPhone, macOS 用に開発している「ブック (Books)^[18]」や、EDRLab が開発している Thorium Reader^[17] がある。Thorium Reader は

Windows OS と macOS の PC に対応している。また、Media Overlays 機能において、読み上げる音読のスピードをユーザーが変更することができる機能を持つ。(Thorium Reader で、縦書きの電子書籍が正しく表示されない時は、電子書籍を開いた後で、画面の拡大や縮小を行なって購読画面の調整を行なうと正しく読めるようになる。)

本論文で取り上げた教育実践においては、iPad タブレットのブックを用いて購読を行っている。

2.2. 教育実践

「文や文節をハイライトしながら同期をとって読み上げを行う」Media Overlays 機能を有する電子書籍を用いて、府中市や八王子市、町田市、多摩市の通常学校の図書館の時間や総合的な学習の時間に教育実践を行なった。

つがる市や我孫子市、府中市、筑波大学の附属特別支援学校、宇都宮大学共同教育学部附属特別支援学校で、また、町田市の特別支援学級で、上肢が不自由で紙の教科書や絵本などのページめくりができない児童生徒、知的障害や弱視のために読みに困難を抱える児童生徒に対して、教育実践を行なった。

新宿区にある日本語学校では、留学生がネイティブな日本語の発音を聞いて授業の振り返りができるようにと Media Overlays 機能を有する電子書籍を用いた教育実践が取り組まれた。

3. Media Overlays 機能を有する EPUB 3 電子書籍の制作

3.1. 電子書籍の制作

Sigil を用いることで、以下に詳述するように、Media Overlays 機能を有する EPUB 3 の国際規格に準拠する電子書籍を制作することができる。

(1) SMIL 言語を用いる方法

ここでは、新宿日本語学校が制作した紙の教科書を、ゼミ生の永野が Media Overlays 機能を有する EPUB 3 準拠の電子書籍に組み替えた「新実用日本語 I」を例に解説する。

図 1 に、制作した電子書籍を iPad のブックで購読しているスクリーンショットを示す。文や文節をハイライトしながら同期をとって読み上げを行なっている様子が見て取れる。(ここでは、読み上げを行なっているテキストが、黄色の背景色に赤色で、そして、大きな文字サイズで表示されている。)



図 1. 「新実用日本語 I」の購読

iPad を横長にしたブックの購読では、同時に 2 ページの内容が表示される。ここでは、左に page_1.xhtml が、右に page_2.xhtml が表示されている。

図 2 には、右のページに相当する page_2.xhtml を Sigil で編集している画面を示す。

Sigil の初期設定では、次のような 3 つの画面に分割されて表示される。左に電子書籍を構成するファイルがフォルダーごとに整理される「ブックブラウザ」、中央に編集中の.xhtml ファイル、そして、右にそのプレビューである。プレビュー画面をチェックしながら、.xhtml ファイルや CSS ファイルなどを編集し、電子書籍を作り込んでいく。

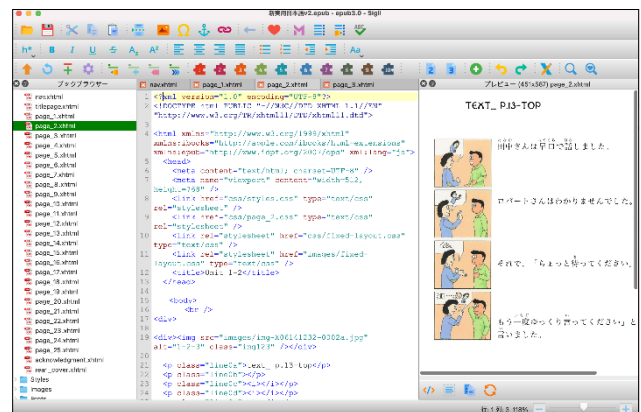


図 2. Sigil による編集画面

図 1 の画面の右のページである page_2.xhtml のコードは、次のように記述されている。(実際にはコードの先頭には番号が付いていないが、ここでは説明がしやすいように番号を付してある。)

[コード例 1 (page_2.html)]

```
1. <?xml version="1.0" encoding="utf-8"?>
2. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
   "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
3. <html
   xmlns="http://www.w3.org/1999/xhtml"
   xmlns:ibooks="http://apple.com/ibooks/html-extensions"
   xmlns:epub="http://www.idpf.org/2007/ops" xml:lang="ja">
4. <head>
5. <meta content="text/html; charset=UTF-8" />
6. <meta name="viewport"
   content="width=512, height=768" />
7. <link href="css/styles.css"
   type="text/css" rel="stylesheet"/>
8. <link href="css/page_2.css"
   type="text/css" rel="stylesheet"/>
9. <title>TEXT_P.13-Top</title>
10. </head>
11. <body>
12. <div>
13. 
14. </div>
15. <div>
16. <p class="line0a">text_ p.13-top</p>
17. <p class="line1"><span id="W1"><ruby>田中<rt>たなか
   </rt></ruby>さんは<ruby>早口<rt>はや
   <rt>くち</rt></ruby>で<ruby>話<rt>はな
   </rt></ruby>しました。</span></p>
18. <p class="line2"><span id="W2">ロボ
   ートさんはわかりませんでした。
   </span></p>
19. <p class="line3"><span id="W3">それ
   で、「ちょっと<ruby>待<rt>ま
   </rt></ruby>ってください</span></p>
20. <p class="line4"><span id="W4">もう
   <ruby>一度<rt>いちど</rt></ruby>ゆっ
   くり<ruby>言<rt>い</rt></ruby>ってく
   ださい」と<ruby>言<rt>い</rt></ruby>
   ました。</span></p>
21. </div>
22. <p ibooksMedia Overlays="start"
   id="raplay">&#x27A4;</p>
23. <p ibooksMedia Overlays="stop"
   id="rastop">&#x2716;</p>
24. </body>
25. </html>
```

コード番号 1 の XML 宣言では、使用する XML のバージョンや文字コードについて記述する。コード番号 2 のドキュメントタイプ宣言では、HTML ファイルで使用するバージョンを宣言する、これら 2 つの宣言は、全ての xhtml ファイルの先頭に記述する^[5]。

これらに続く、コード番号 3 の HTML 要素では、文書が HTML 文書であることを示し、XML の名前空間や使用する言語を記述する^[5]。

<html> 要素には、子要素として、<head> 要素と <body> 要素が記述される。

<head> 要素には、<meta> 要素 (コード番号 5 と 6)、<link> 要素 (コード番号 7 と 8)、<title> 要素 (コード番号 9) などが記述される。<meta> 要素には文字コードや画面の表示領域などの HTML 文書の基本情報を、<link> 要素には HTML 文書にリンクする CSS ファイルや JavaScript ファイルなどを記述する。コード番号 5 から 7 までは、全ての xhtml ファイルで共通に記述している。コード番号 8 は、編集集中のページ page_2.xhtml ファイルだけにリンクする CSS ファイルである。<title> 要素には、そのページのタイトルを記述する。これら <head> 要素に記述した内容は、購読時の画面には表示されない。

一方、<body> 要素 (コード番号 11~24) に記述した内容は、購読者から見えるコンテンツとなる。<body> 要素の中で、<!-- コメント --> のようなコードを記述することで、文書に影響を与えないコメントを挿入することができる。(xhtml ファイルで使用する <style> 要素や <script> 要素、及び、独立した CSS ファイルや JavaScript ファイルでは、/* コメント */ のように記述する。)

図 1 のスクリーンショットの赤字でハイライトされているテキストはコード例 1 のコード番号 17 の赤字の部分に対応している。このコードでは、テキストの配置位置などを設定する class 属性 (line1)、文や文節をハイライトしながら同期をとって読み上げを行う SMIL 言語との対応を取るための id 付きの span 要素 (W1)、そして、テキスト中の漢字にルビを振っている <ruby> 要素などからなる。

ところで、コード例 1 で示した page_2.xhtml ファイルは、2 つの CSS ファイル (styles.css と page_2.css) をリンク (コード番号 7 と 8) している。この中で、styles.css ファイルは、全ての xhtml

ファイルでリンクされ、それぞれのページで表示される画像の大きさや配置位置、また、音声の再生時にハイライトされるテキストの文字色、その背景色、フォントサイズなども、次のコード例 2 のように記述されている。

```
[コード例 2 (styles.css 一部)]
.-epub-media-overlay-active {
  color: red;
  background-color: yellow;
  font-size: 28px;
}
```

本電子書籍のような「横書き、左から右へのページ送り」の設定は、次のコード例 3 の赤字のコードで示すように記述する。購読の際の画面の表示に用いるテキストのフォントの種類やその大きさなどとともに、同じく styles.css ファイルの中で記述してある。

```
[コード例 3 (styles.css 一部)]
html {
  font-family: "HiraMinProN-W3", "@MS
明朝", serif, sans-serif;
  writing-mode: horizontal-lr;
  -webkit-writing-mode: horizontal-lr;
  -epub-writing-mode: horizontal-lr;
  line-height: 1.75;
  text-align: justify;
}
p {font-size: 16px; letter-spacing: 1px;
position:absolute;
}
```

コード例 1 で示した page_2.xhtml ファイルでリンクしているもう一つの CSS ファイル (page_2.css) は、次のコード例 4 のように記述されている。(それぞれのコードには実際には番号が付いていないが、説明がしやすいように番号を付してある。)

```
[コード例 4 (page_2.css 一部)]
1. @charset "utf-8";
2. @font-face {
  font-family: "Hadley";
  font-style: normal;
  font-weight: normal;
  src:url("../fonts/Hadley.ttf");
}
3. p.line0a {top: 20px; left: 180px;
font-family: "Hadley"; font-size:
20px; }
```

```
4. p.line1 {top: 100px; left: 160px; }
5. p.line2 {top: 220px; left: 160px; }
6. p.line3 {top: 340px; left: 160px; }
7. p.line4 {top: 460px; left: 160px; }
```

ページのタイトルは、コード例 4 のコード番号 3 で示すように、Hadley フォント^[21] (ブックブラウザの Fonts フォルダに収められている) を用いて記述してある。また、ハイライトされている「田中さんは早口で話しました。(コード例 1 のコード番号 17 に対応)」というテキストは、コード番号 4 で、「上から 100 ピクセル、左から 160 ピクセルの位置から表示」するように <p> 要素の class 属性 (class="line1") を用いて設定している。

ところで、文や文節をハイライトし同期をとって読み上げを行う Media Overlays の機能は、次のような SMIL を用いて実現している。

例えば、コード例 1 のコード番号 17 の span 要素の id 属性 (id="W1") で指定したテキスト「田中さんは早口で話しました。」は、次のコード例 5 の SMIL ファイル (page_2.smil) の赤色のコードで示した <par> 要素 (<par id="par1">) に対応している。

```
[コード例 5 (page_2.smil 一部)]
<?xml version="1.0" encoding="UTF-8"?>
<smil xmlns="http://www.w3.org/ns/SMIL"
  version="3.0"
  profile="http://www.idpf.org/epub/30/
  profile/content/">
<body>
<par id="par1">
<text src="page_2.xhtml#W1" />
<audio src="audio/all+.mp3"
clipBegin="15.576s" clipEnd="18.548s" />
</par>
<par id="par2">
<text src="page_2.xhtml#W2" />
<audio src="audio/all+.mp3"
clipBegin="20.226s" clipEnd="22.895s" />
</par>
...
</body>
</smil>
```

音声の再生は <audio> 要素を用いて、読み上げる音声ファイル名、読み上げの開始点 (clipBegin) と終了点 (clipEnd) を指定することで実現している。コード例 1 のコード番号 17 のテキストの再生は、all+.mp3 という音声ファイルの 15.576 秒

から 18.548 秒までの再生となっている。(音声ファイル中の該当するテキストの再生の開始点と終了点は, Audacity^[22] などの音声編集ソフトウェアを用いて読み取ることができる。)

最後に行う大切な作業は, 制作中のファイルを Sigil のファイルメニューの「名前を付けて保存」を選んで一度保存し, 「ブックブラウザ」の一番下にある Package Document^[5] と呼ばれる content.opf ファイルの <manifest> 要素の中の <item> 要素に, 次のコード例 6 の赤字のコードが追加されているかを確認することである。追加されていない時は, 追加する必要がある。

[コード例 6 (content.opf 一部)]

```
<item id="page_2.xhtml"
href="page_2.xhtml" media-
type="application/xhtml+xml" media-
overlay="page_2.smil"/>
```

(2) prefix を用いた機能の追加

ここでは, SMIL 言語を用いて Media Overlays 機能を有する英語学習用の Emi & Alex の電子書籍を制作した際に追加した機能について解説する。

図 3 に示すように, この電子書籍では, 文や文節をハイライトしながら同期をとって読み上げを行えるだけでなく, 英文の単語の品詞ごとに, 「形と色」を変えた図形を被せてあり, 英文法も学べるように工夫してある。

購読中の iPad のスクリーンの左のページ (page_1.xhtml) の赤色にハイライトされている Emi に続く Alex の会話 (Hello. I'm Alex. Nice to meet you.) の箇所のコードは, 次に示すコード例 7 の赤色の部分のように記述してある。

[コード例 7 (page_1.xhtml 一部)]

```
<p class="line0a">Unit 1-1</p>
<p class="line0b">Nice to meet you.</p>
<p class="line0c"><i>Emi:</i></p>
<p class="line0d"><i>Alex:</i></p>
<p class="line0e"><i>Emi:</i></p>
...
<p class="line2"><span id="W2">Hello.
<span class="prefix_hh8"> I</span>'<span
class="prefix_hh9">m</span> <span
class="prefix_hh1">Alex</span>. <span
class="prefix_hh11">Nice</span> to <span
class="prefix_hh9">meet</span> <span
class="prefix_hh3">you</span>.</span>
</p>
```

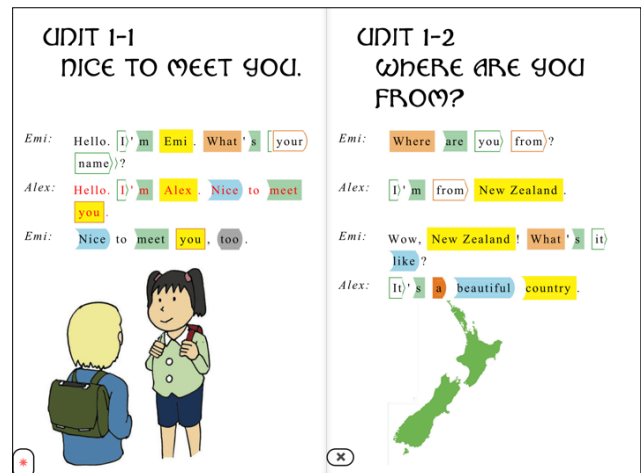


図 3. 「Emi & Alex」の購読
(品詞ごとに, 形と色を変えた図形を被せてある)

単語の品詞ごとに被せている図形 (例えば, Alex という単語 (名詞) に被せてある黄色の長方形) は, 要素の class 属性を用いて class="prefix_hh1" のように指定することで実現している。ここで用いている prefix_hh1 は, 全ての xhtml で共通に参照する styles.css ファイルで, 次のコード例 8 のように記述している。

[コード例 8 (styles.css 一部)]

```
.prefix_hh1 {
-moz-border-image:
url("../images/h1.jpg") 20 round
stretch;
-webkit-border-image:
url("../images/h8.jpg") 20 round
stretch;
-o-border-image:
url("../images/h1.jpg") 20 round
stretch;
-ms-border-image:
url("../images/h1.jpg") 20 round
stretch;
border-style: solid; border-width:
3px 8px 3px 3px;
}
```

コード例 8 のように記述することで, 同じ品詞に被せる図形は, 単語の長さに応じて自動的に拡大・縮小される。(prefix_hh1 で参照している図形 (h1.jpg) は, 「ブックブラウザ」の Images フォルダに保存してある。)

3.2. 縦書き，右から左へ列を並べる右縦書きの電子書籍の制作

国語の教科書のような「縦書き，右から左へ列を並べる右縦書き」の電子書籍は，すべてのページで参照する `styles.css` の中で，次のコード例9の赤色のコードのように，縦書きのテキストフォント（ここでは，@MS 明朝）と `writing-mode: vertical-rl;` を指定して実現する。

[コード例9 (styles.css 一部)]

```
html {
  font-family:"ヒラギノ明朝 ProN W3",
  "HiraMinProN-W3", "@MS 明朝", serif,
  sans-serif;
  writing-mode: vertical-rl;
  -webkit-writing-mode: vertical-rl;
  -epub-writing-mode: vertical-rl;
  line-height: 1.75;
  text-align: justify;
}
```

図4には，五十音のひらがなとカタカナなどを学ぶ右縦書きの電子書籍を iPad のブックで購読しているスクリーンショットを示す。

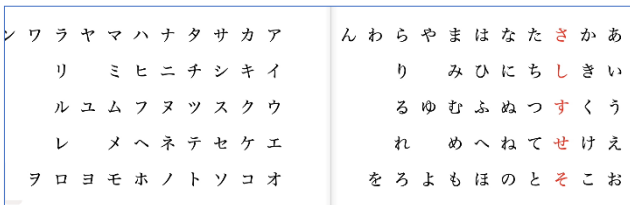


図4. 縦書き，右から左へ列が進む
右縦書きの電子書籍の購読

この図4の右のページの赤字でハイライトされている「さしすせそ」は，`p1.xhtml` ファイルの `<body>` 要素の中で，次のコード例10の赤字のコードのように記述されている。

[コード例10 (page1.xhtml 一部)]

```
<!-- 「本文テキスト」の挿入 (css と smil
ファイルとの連携あり) -->
<p class="line1"><span id="W1">あ い
う え お</span></p>
<p class="line2"><span id="W2">か き
く け こ</span></p>
<p class="line3"><span id="W3">さ し
す せ そ</span></p>
<p class="line4"><span id="W4">た ち
つ て と</span></p>
```

右縦書きの電子書籍の購読を行った際に，テキストの縦書きと右から左への列の展開は実現できているが，国語の教科書のようなページめくり(右開き)が実現していない時は，「ブックブラウザー」の最下部にある `content.opf` ファイルの `<spine>` 要素の `page-progression-direction` を確認し，`default` になっていたら，次の赤字のように `rtl` に修正する必要がある。

```
<spine
page-progression-direction="rtl">
```

3.3. button のクリックイベント機能を用いた音声の再生

図5は，新宿日本語学校の教科書「新実用日本語中級 I」を Media Overlays 機能を有する電子書籍に作り替えたものである。この電子書籍では，Media Overlays 機能に加えて，テキストの左横に配置されているスピーカーアイコンのクリックで，音声によるテキストの解説が再生される。この機能は，以下で詳述するように，`button` のクリックイベント機能と JavaScript の関数を用いて実現している。（さらに，この電子書籍では，画面トップに配置されている「江副文法^[23]」を学ぶ画像をクリックすると動画で文法の解説が流れる。）

音声によるテキストの解説を実現するために，コード例11に示すように，`<head>` 要素の中で `<style>` 要素を用いて，スピーカーのトリガー画像 (`img11`) の配置位置を，そして，スピーカー画像に被せる（目には見えない）透明なボタン (`button11`) を被せている。さらに，ボタンがクリックされたら音声の再生を開始し，再度押されたら停止するように JavaScript を用いて関数 (`myFunction()`) の定義を行ない，`<button>` の要素として設定している。スピーカーのトリガー画像の大きさやリンクする音声ファイルは，`<body>` 要素の中で `` 要素を用いて，また，音声の再生は `<audio>` 要素の中で `myFunction()` を参照して実現している。

コード例11に挿入してあるコメント行を頼りにコードを追いかけることを勧める。（実際の電子書籍では，スピーカーの横の「例1」をクリックすると様々な言語でテキストがポップアップ表示される機能も追加してある。）

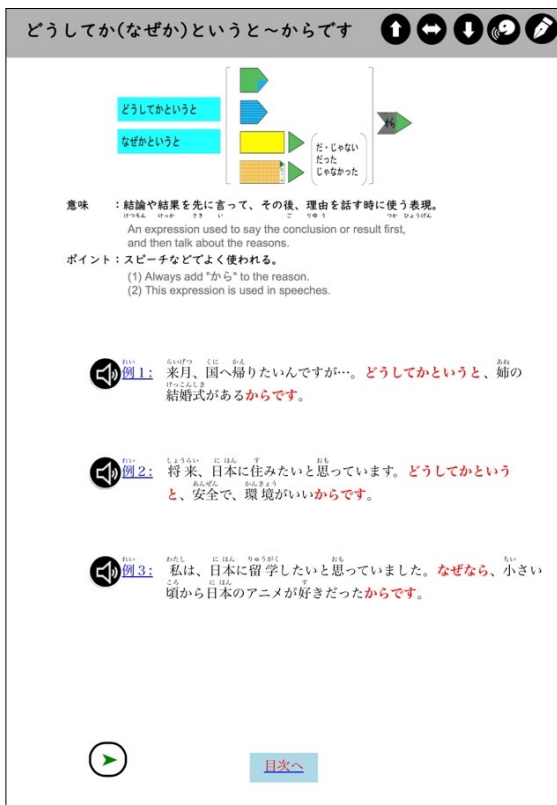


図 5. クリックイベントを活用した電子書籍の購読

[コード例 11 (page2.xhtml 一部)]

```

<head>
<style>
/* クリックイベントアイコン画像の配置位置 */
#img11{
position: absolute;
display: block;
top: 325px;
left: 80px;
}
</style>
<style>
/* クリックイベントアイコン画像に被せる透明
な button の大きさや配置位置などの定義 */
.button11 {
background: transparent;
color: white;
width: 30px;
height: 30px;
border-radius: 50%;
position: absolute;
top: 325px;
left: 80px;
}
</style>
<script>
/* 透明な button がクリックされたときに起こ

```

```

るイベント (音声の再生・停止) 関数の定義 */
function myFunction(id1) {
var audio =
document.getElementById(id1);
if (audio.paused) {
audio.play();
$('#play').removeClass('glyphicon-play-
circle')
$('#play').addClass('glyphicon-pause')
}
else{
audio.pause();
audio.currentTime = 0
$('#play').addClass('glyphicon-play-
circle')
$('#play').removeClass('glyphicon-
pause')
}
}
</script>
</head>
<body>
<div>
/* クリックイベントアイコン画像のファイル名
とその大きさ*/

</div>
<div>
/*ボタンがクリックされた時の音声再生イベン
トの定義 */
<audio id="L11"
src="audio/p2_1.mp3"></audio>
<button class="button button11"
onclick="myFunction('L11')"></button>
<p id="L11"></p>
</div>

```

3.4. JavaScript (ibooks.js) を用いた音声の再生

これまで述べた SMIL 言語や <button> 要素と関数を組み合わせて音声を再生する代わりに JavaScript パッケージを用いることで、音声を再生する電子書籍を作ることができる。

テキストが多すぎて「同期をとってハイライトするのが困難」な電子書籍や、逆に全くテキストがなく SMIL 言語を用いたテキストとの対応ができない電子書籍で威力を発揮する。

ここでは、アップルの `ibooks.js`^[24] という JavaScript パッケージを用いて音声の読み上げを実現した電子書籍について解説する。

図 6 に示す電子書籍は、町田市の教育委員会の許可を得て、ゼミ生の櫻井と高柳が、弱視の児童

のために、町田市立本町田東小学校の教員とともに作ったものである。小学校3,4年生用の副読本「わたしたちの町田」の電子書籍である。

図のスピーカーのアイコンをクリック（タッチ）することで、リンクしてある音声再生される。



図 6. JavaScript (ibooks.js) を用いて制作した「わたしたちの町田」の購読

図の左のページの一番上のスピーカーのアイコン (img1) は、次のコード例 12 のように、<head> 要素の中で <style> 要素を用いて、上から 30 ピクセル、左から 10 ピクセルの位置に配置している。また、音声の再生を実現するために、<script> 要素を用いて JavaScript (ibooks.js) パッケージの宣言を行なっている。この ibooks.js ファイルは、「ブックブラウザ」の Misc フォルダの中に入れてある。

[コード例 12 (page_92.xhtml の <head> 要素の一部)]

```
<style>
  #img1{
    position: absolute;
    display: block;
    top: 30px;
    left: 10px;
  }
</style>
<script src="ibooks.js"
type="text/javascript"></script>
```

画像 (img1) のファイル名 (audio-icon.jpg) と画像の大きさ (width="50" height="50"), クリックさ

れたときに再生される音声ファイル名 (p92-1.mp3) は、次のコード例 13 のように <body> 要素の中で 要素の属性として記述してある。

[コード例 13 (page_92.xhtml の <body> 要素の一部)]

```

```

スピーカーのアイコンをクリックすることで音声再生され、もう一度クリックすると停止、再度クリックすると音声再生される。この ibooks.js パッケージを用いた電子書籍作りは簡単であり、テキストが多すぎたり、逆に少なすぎて SMIL による Media Overlays 機能を有する電子書籍を制作することが困難なときに重宝する。ただし、アップルの JavaScript パッケージを用いており、iPhone, iPad, MacBook などのブックでの購読に限られる。

3.5. <button> 要素のクリックイベントと JavaScript 関数を用いた動画の再生と制御

図 7 に示す電子書籍では、表示されているボタン (再生/停止, 大きく, 小さく, 元) をクリックすることで、ポスター画像にリンクしてある動画の再生やその大きさなどを可能にしている。

ポスター画像にリンクされている動画の再生などの制御は、次の手順で実現している。

- <head> 要素の中で、動画のポスター画像 (img1) の配置位置を設定。
- 4 つの button (button1~button4) のバックグラウンドの色, ボタン内のテキストの色, ボタンの枠の太さや色, ボタンの大きさ, ボタンの配置位置などを設定。
- それぞれの button がクリックされたときに起こるイベント (動画の「再生/停止, 大きく, 小さく, 元」) 処理の内容を定義。
- <video> 要素で、ポスター画像 (site21.jpg) とボタンのクリックによって再生される動画ファイル (site21.mp4) やその大きさ (横幅 125px) などを設定。
- 4 つの button と連動して動画の再生を実現する関数 (myVideo()) を定義。



図 7. <button> 要素, <video> 要素, 関数を用いて動画を制御する電子書籍の購読

次のコード例 14 に, 具体的なコードを示す. コメントを頼りに追いかけることを勧める.

[コード例 14 (Section1.xhtml 一部)]

```
<style>
/* ポスター画像の配置位置の設定 */
.img1{
display: block; position: absolute; top:
250px; left: 100px;
}
/* 「再生/停止」のボタンの大きさと配置位置
*/
.button1 {
background-color: green; color: white;
border: 2px solid #4CAF50; width: 80px;
height: 30px; position: absolute; top:
10px; left: 100px;
}
/* 「大きく」のボタンの大きさと配置位置 */
.button2 {
background-color: blue; color: white;
border: 2px solid #008CBA; width: 80px;
height: 30px; position: absolute; top:
10px; left: 183px;
}
/* 「小さく」のボタンの大きさと配置位置 */
.button3 {
background-color: red; color: white;
border: 2px solid #008CBA; width: 80px;
height: 30px; position: absolute; top:
10px; left: 266px;
}
/* 「元に」のボタンの大きさと配置位置 */
.button4 {
background-color: black; color: white;
border: 2px solid #008CBA; width: 80px;
height: 30px; position: absolute; top:
10px; left: 348px;
}
</style>
</head>
<body>
<div style="text-align:center">
<!--それぞれのボタンのクリックイベントの処
```

理内容 -->

```
<button class="button button1"
onclick="playPause()">再生/停止</button>
<button class="button button2"
onclick="makeBig()">大きく</button>
<button class="button button3"
onclick="makeSmall()">小さく</button>
<button class="button button4"
onclick="makeNormal()">元に</button>
<!--表示されているポスター画像とボタンのク
リックによって再生される動画ファイルのファ
イル名, 動画の再生を行う関数への引数渡し-->
<video id="video1" width="125"
poster="images/site21.jpg"
class="ibooks-media-video; img1"
src="video/site21.mp4"
controls="controls">
</video>
<script>
/* 動画の再生を実現する関数, それぞれの
<button> 要素と連動 */
var myVideo =
document.getElementById("video1");
function playPause() {
if (myVideo.paused)
myVideo.play();
else
myVideo.pause();
}
function makeBig() {
myVideo.width = 450;
}
function makeSmall() {
myVideo.width = 100;
}
function makeNormal() {
myVideo.width = 125;
}
</script>
</div>
</body>
```

4. Media Overlays 機能を有する電子書籍を用いた教育実践

4.1. 通常学校における教育実践

府中市, 町田市, 多摩市, 八王子市の通常学校で Media Overlays 機能を有する電子書籍や音声の再生機能を有する電子書籍を用いて, 次のような教育実践を行なった.

- ・クラス毎に図書室で行われた読書の時間に, ゼミ生がブースを出し, 児童が順番に回りながら教科書の単元や絵本の電子書籍を音読すること

ができた。iPad に付けたイヤフォンを二人で共有しながら聞きながら電子書籍に聞き入っていた。

- ・「戦争の悲惨さを後世まで語り継いで欲しい」と作者に託された「ランドセルをしょったじぞうさん^[25]」の電子書籍は、国語の時間や総合的な学習の時間で平和学習に活用された。この電子書籍の日本語版の音声は小学校の国語の先生が、英語版の音声は ALT (Assistant Language Teacher) が担当した。
- ・英単語や簡単な英文を学ぶために Media Overlays 機能を有する電子書籍を作成し、外国語活動の時間や学校のイベントなどで音読活動を行った。
- ・Media Overlays 機能を実現するために、教員が音声を録音して電子書籍の制作に参加する学校もあり、児童たちが「あ！先生の声だ！と大喜びする姿が数多く見られた。
- ・退職する教員の声を用いて Media Overlays 機能を有する電子書籍を制作することで、学校のイベントなどで、児童がその電子書籍を聴きながら退職した先生を懐かしむできごとが起こった。

4.2. 特別支援学校、特別支援学級における教育実践

青森県立森田養護学校、筑波大学附属桐が丘特別支援学校、筑波大学附属大塚特別支援学校、東京都立府中けやきの森学園、町田市立本町田東小学校の特別支援学級などで Media Overlays 機能を有する電子書籍を用いて、次のような教育実践を行なった。

- ・上肢が不自由で紙の教科書や絵本のページめくりができない児童生徒が、EPUB 3 eBooks Reader の自動ページ送り機能を用いて、一人で電子書籍の音読活動を行うことができ、達成感、自立感を味わうことができた。
- ・読みのスピードが遅い、読み間違いや読み飛ばしの多いダウン症の児童が、電子書籍の Media Overlays 機能を有する電子書籍を用いて、「音読を行う、音読に続けて真似て読む、再度音読を聞く」活動を繰り返すことで、読みのスピードが改善しただけでなく、読み間違いや読み飛ばしが少なくなった。

- ・知的障害を有する児童が、食事後に、上の歯、下の歯など、歯の様々な部分を「丁寧に、満遍なく磨く」ことができるように、担任の教員が「歯の各部分の画像と 10 まで数える音声」が流れる電子書籍」制作した。iPad を見ながら、児童が一人で歯磨きに挑戦できた。
- ・知的障害を有する生徒たちが、「餃子を作る、ラスクを作る、シフォンケーキを作る」ための手順を記したテキストと生徒自身の声で手順を解説する電子書籍を制作し、調理実習で活用した。それぞれの生徒の調理実習の取り組みの様子を定量的に評価することで、Media Overlays 機能を有する電子書籍の効果を明らかにすることができた。
- ・知的障害を有する児童生徒の学校で、昼休みや中休み（20 分休み）、余暇の時間に Media Overlays 機能を有する絵本の電子書籍を購読して過ごす活動が取り組まれた。こうした学校の教員の中には、自ら絵本の電子書籍を制作した教員もいた。
- ・発語が曖昧な児童生徒、2・3 語文程度の読みの児童生徒、単語を絵や塊として記憶している児童生徒のために、五十音や簡単な単語の読みや発音を学ぶ取り組み、語彙を増やす取り組みが行われた。図 4 に示したような、「ひらかな・カタカナ、濁音、撥音、拗音、簡単な単語や文の学習」のために、電子書籍の改訂が続けられている。
- ・教科書の通常の文字サイズでは音読ができない弱視の児童のために、JavaScript パッケージを用いた電子書籍（図 6）を制作した。社会科の紙の副読本の代わりに、電子書籍の音声で副読本を学ぶことができた。

4.3. 日本語学校における教育実践

新宿日本語学校と協働で、次のような活動に取り組んだ。

- ・留学生の日本語学習用に制作された紙の教科書を Media Overlays 機能を有する電子書籍に変換することで、学校での学習を各自が振り返り、深めることができることを狙った（図 1, 5）。ネイティブな日本語を聞き直すことで、授業で学んだことをブラッシュアップできると好評だった。

- ・図5に示した日本語学習用の教材においては、Media Overlays 機能に加えて、江副文法を学ぶ動画を付加している。また、留学生が自分の母国語と対比しながら学べるように、各ページの日本語のテキストを様々な国の言語でポップアップ表示するボタンを付ける作業を続けている。
- ・英会話学習用の教材(図3)では、新宿日本語学校の校長が開発した江副文法の学習できるように、単語の品詞ごとに違った形と色を被せることで、英文法が学べる電子書籍を開発した。この電子書籍は、日本語学校だけでなく通常学校や特別支援学校などでも活用されている。

5. まとめと今後の課題

「文や文節をハイライトしながら同期をとって音声再生する」Media Overlays 機能を有する電子書籍の制作^[26-33]は、当初、日本のフューズネットワーク株式会社の FUSEe を用いて行った。しかしながら、FUSEe がまもなく販売中止となりバージョンアップもされないことから、著者は、ここ数年、最新の国際規格にも準拠し、評価の高い Sigil を用いて電子書籍の制作を行っている。

教育現場で電子教科書を用いた授業が始まろうとする今こそ、一人ひとりの教員が電子書籍に関する基礎的な知識を習得するとともに、Sigil のような無料のソフトウェアを用いて電子書籍作りに挑戦してみることが大切と考える。本論文で解説した Media Overlays 機能を有する電子書籍を一から制作することは簡単ではないが、既存の電子書籍を雛形として用いることで、自分の目指す電子書籍に作り替えることはそんなに難しいことではない。

EPUB 3 eBooks Reader の中には、Thorium Reader のように、読みのスピードをユーザーが購読時に自由に変えることができものがある。こうしたリーダーは、様々な障害のために、棒読みをしてしまう、読みのスピードが遅い、読みの誤りが多い、読み飛ばしが多い、書いていないことを読んでしまう、などの音読活動の苦手な児童生徒に効果を発揮するものと期待される。

Media Overlays 機能を有する電子書籍は、知的障害を有するために、2・3 語程度の簡単な単語を絵のような塊として覚えてしまっている児童生徒に、「ひらがな一文字一文字に読みがあり、単語を構成している」ことを伝える、言葉の発音が曖昧

で周りの児童生徒に理解されにくい児童生徒の発語の改善に、また、語彙の少ない児童生徒の語彙の獲得のための教材として威力を発揮するものと期待される。

謝辞

電子書籍の制作、その実践にあたっては、科学研究費補助金(18330198, 29653068, 22530992, 16K04844)や大妻女子大学戦略的個人研究費(S2605, S2713, S2810, S2912)のお世話になりました。ここに記して感謝します。

引用文献

- [1]文部科学省, 学校教育法第三十四条第二項に規定する教材の使用について定める件の一部を改正する件の公布及び施行等について(通知), 平成3年3月26日, https://www.mext.go.jp/a_menu/shotou/kyoukasho/seido/1412813_00001.htm, (参照 2023-10-25)
- [2]文部科学省, デジタル教科書に関する制度・現状について, https://www.mext.go.jp/content/20200710-mxt_kyokasyo-000008653_03.pdf, (参照 2023-10-25)
- [3]Media Overlays, W3C, <https://www.w3.org/TR/epub-33/#sec-media-overlays>
- [4]EPUB 3.0, International Digital Publishing Forum (idpf), <https://idpf.org/epub/30/>, (accessed 2023-10-25).
- [5]EPUB 3.3, W3C, <https://www.w3.org/TR/epub-33/>, (accessed 2023-10-25).
- [6]HTML Living Standard, <https://html.spec.whatwg.org/multipage/>, (accessed 2023-10-25).
- [7]CSS Syntax Module Level 3, W3C, <https://www.w3.org/TR/css-syntax-3/>, (accessed 2023-10-25).
- [8]SMIL, W3C, <https://www.w3.org/TR/epub-33/#sec-smil-smil-elem>, (accessed 2023-10-25).
- [9]ZIP, WIKIPEDIA, [https://en.wikipedia.org/wiki/ZIP_\(file_format\)](https://en.wikipedia.org/wiki/ZIP_(file_format)), (accessed 2023-10-25).
- [10]W3C (World Wide Web Consortium), <https://www.w3.org>, (accessed 2023-10-25).
- [11]HON.jp News Blog, フューズネットワーク, Windows 用の ePub 電子書籍作成ソフト「FUSEe (フュージー)」をベータ公開,

- <https://hon.jp/news/1.0/0/1499>, (参照 2023-10-25)
- [12]Sigil, <https://sigil-ebook.com>, (accessed 2023-10-25).
- [13]The Daisy Consortium, <https://daisy.org>, (accessed 2023-10-25).
- [14]WordToEpub, The Daisy Consortium, <https://daisy.org/activities/software/wordtoepub/>, (accessed 2023-10-25).
- [15]ePub Zip/Unzip 3.0, mac.informer, <https://macdownload.informer.com/epub-zip-unzip/><https://macdownload.informer.com/epub-zip-unzip/>, (accessed 2023-10-25).
- [16]Tobi, The Daisy Consortium, <https://daisy.org/activities/software/tobi/>, (accessed 2023-10-25).
- [17]Thorium Reader, EDRLab, <https://www.edrlab.org/software/thorium-reader/>, (accessed 2023-10-25).
- [18]Apple Books, Apple Store, <https://apps.apple.com/jp/app/apple-books/id364709193>, (accessed 2023-10-25).
- [19]PLEXTALK Producer, Shinano Kenshi Co., Ltd., <http://www.plextalk.com/jp/products/producer/>, (参照 2023-10-25)
- [20]リーダー, Shinano Kenshi Co., Ltd., <http://www.plextalk.com/jp/products/producer/>, (参照 2023-10-25)
- [21]Hadley, Font Memo, <https://fontmeme.com/jfont/hadley-font/>, (accessed 2023-10-25).
- [22]Audacity, <https://www.audacityteam.org>, (accessed 2023-10-25).
- [23]江副隆秀. 新版 見える日本語, みせる日本語. 創拓社出版. 2014.
- [24]iBooks.js, Apple Books アセットガイド 5.3, <https://help.apple.com/itc/booksassetguide/>, (参照 2023-10-25)
- [25]古世古和子 さく, 北島新平 え. ランドセルをしょったじぞうさん. 新日本出版. 1980.
- [26]生田茂ほか. Media Overlays 機能を取り込んだ電子書籍の制作と教育実践. 社会情報学研究 (大妻女子大学紀要 - 社会情報系 -). 2014, 23, p.121-135.
- [27]生田茂ほか. 文や文節をハイライトしながら同期をとって読み上げを行う電子書籍を用いた音読活動の取り組み. 人間生活文化研究. 2014, 24, p.160-167.
- [28]生田茂ほか. 最新の情報処理技術を活用した手作り教材の制作と教育実践 - 国内外の共同研究者との協働の取り組み. 人間生活文化研究. 2015, 25, p. 37-64.
- [29]生田茂ほか. 最新の情報処理技術を活用した手作り教材の制作と教育実践 - 国内外の共同研究者との協働の取り組み. 人間生活文化研究. 2016, 26, p. 239-262.
- [30]生田茂ほか. 学校の先生と取り組む合理的配慮指針に基づく手作り教材の制作と教育実践. 人間生活文化研究. 2017, 27, p. 156-204.
- [31]生田茂ほか. 学校の先生と取り組む手作り教材の制作と教育実践. 人間生活文化研究. 2018, 28, p. 137-178.
- [32]Ikuta, Shigeru et al. "Original Teaching Materials and School Activities with E-Books Containing Media Overlays". Handmade Teaching Materials for Students With Disabilities. Shigeru Ikuta, 2019, p. 76-110.
- [33]Ikuta Shigeru et al. "University Students: Schoolteachers Partnership With Newly Developed Technologies". Enhancing Learning Opportunities Through Student, Scientist, and Teacher Partnerships. D. Farland-Smith, 2021, p. 133-157.

Abstract

In collaboration with schoolteachers, the present author has been performing various school activities with self-made eBooks with Media Overlays, where a text highlighting synchronizes with the corresponding audio playback. Self-made eBooks with audio playback, realized with JavaScript and Button OnClick Event, were also used there. Using the automatic page turning in EPUB 3 eBooks readers, eBooks with Media Overlays and audio playback were nicely used by the students, with disabilities in their fingers or upper limbs, who could not turn the pages of regular textbooks and picture books by themselves before. The reading activities for the student who frequently made reading errors or skipping several characters/words before, were improved by using the eBooks with Media Overlays. The student with low vision, who was unable to read regular textbooks and side readers before, came to progress while playing eBook with audio playback. EPUB 3 eBooks with Media Overlays can be hoped well to maximize accessibilities for students with various reading disabilities. In the present paper, the author will carefully explain several important points in creating EPUB 3 eBooks with Media Overlays and EPUB 3 eBooks with audio playback realized with JavaScript and Button OnClick Event, and refer to the potential of self-made eBooks in special needs education settings.

(受付日：2023年10月30日，受理日：2023年11月24日)

生田 茂 (いくた しげる)

現職：大妻女子大学人間生活文化研究所 特別研究員

東北大学大学院理学研究科博士課程修了（理学博士）。

専門は特別支援教育，教育工学。現在も二次元コードを活用した音声ペンやドットコードリーダーを用いて，特別支援学校の教員たちとともに，様々な困難を抱える児童生徒の「これまではできなかった」ことを「可能にする」取り組みに挑戦している。

主な著書：(1) Building Inclusive Education in K-12 Classrooms and Higher Education: Theories and Principles (共著, IGI-Global, 2023) (2) Developing Inclusive Environments in Education: Global Practices and Curricula (共著, IGI-Global, 2023) (3) Handbook of Research on Software for Gifted and Talented School Activities in K-12 Classrooms (IGI-Global, 2020) (4) Handmade Teaching Materials for Students With Disabilities (IGI-Global, 2019)