Multimedia Vision for the Visually Impaired through 2D Multiarray Braille Display

Seondae Kim [ID], **Eun-Soo Park** [ID] **and Eun-Seok Ryu *** [ID]

Department of Computer Engineering, School of Information Technology, Gachon University, 1342 Seongnam-daero, Sujeong-gu, Seongnam-si, Gyeonggi-do 13120, Korea; ele7004@gc.gachon.ac.kr (S.K.); dmsehf804@gc.gachon.ac.kr (E.-S.P.)

* Correspondence: esryu@gachon.ac.kr; Tel.: +82-31-750-8905

check for updates

Abstract: Visual impairments cause very limited and low vision, leading to difficulties in processing information such as obstacles, objects, multimedia contents (e.g., video, photographs, and paintings), and reading in outdoor and indoor environments. Therefore, there are assistive devices and aids for visually impaired (VI) people. In general, such devices provide guidance or some supportive information that can be used along with guide dogs, walking canes, and braille devices. However, these devices have functional limitations; for example, they cannot help in the processing of multimedia contents such as images and videos. Additionally, most of the available braille displays for the VI represent the text as a single line with several braille cells. Although these devices are sufficient to read and understand text, they have difficulty in converting multimedia contents or massive text contents to braille. This paper describes a methodology to effectively convert multimedia contents to braille using 2D braille display. Furthermore, this research also proposes the transformation of Digital Accessible Information SYstem (DAISY) and electronic publication (EPUB) formats into 2D braille display. In addition, it introduces interesting research considering efficient communication for the VI. Thus, this study proposes an eBook reader application for DAISY and EPUB formats, which can correctly render and display text, images, audios, and videos on a 2D multiarray braille display. This approach is expected to provide better braille service for the VI when implemented and verified in real-time.

Keywords: 2D braille display; visual impairment; braille translation; tactile graphic; braille eBook

1. Introduction

In 2015, among the global population of 7.33 billion people, 36 million were classified as legally blind, about 217 million had moderate to severe visual impairment, and 188 million had mild vision impairment. Statistically, the number of people who lose sight increases proportionally with increase in either population or the age of the people [1]. According to the US Census Bureau, the population of the United States is increasing every year [2] and, as it grows older, the number of diseases affecting vision increases [3]. Figure 1 shows that age is correlated with vision disorders [1]. It shows that, as aged populations grow, there is a growing number of vision disorders. To address the difficulties faced by the visually impaired (VI) people, there have been many studies on providing suitable alternatives. Usually, there are several devices, such as walking canes, which help the blind to navigate and there are aids that magnify the size of letters for people with low vision who do not require braille for reading. Generally, the younger among the VI people who have a communication-focused lifestyle, e.g., students and the employed, learn and use braille. For such individuals, information is usually sent and received using braille printing or braille displays (i.e., braille devices).
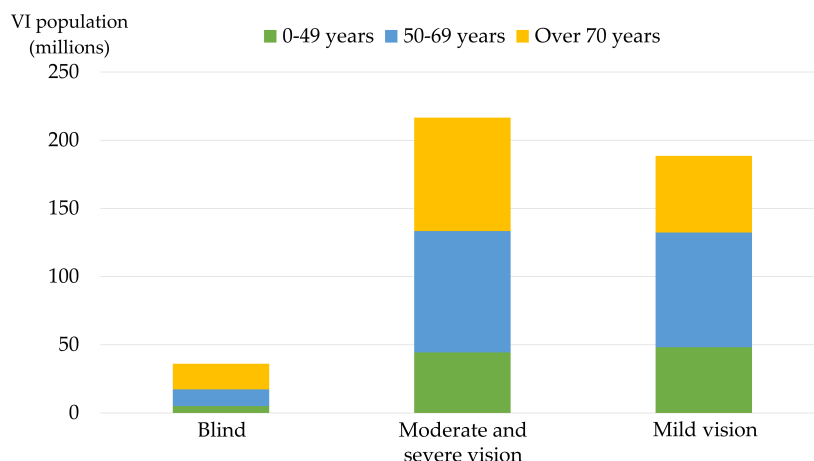
Figure 1. Relationship between the age of people who are blind, degree of visual impairments, and the number of visually impaired.

Typically, smart canes are operative devices that are attached to canes for the VI to assist them in detecting obstacles, to provide navigational guidance, etc. There are also methodologies using cameras that contribute to vision sensing, text reading, guidance, navigation, and obstacle detection. To telepresent information such as 3D shapes and 2D edges, haptic devices are used, which can depict 3D/2D objects. Furthermore, braille displays or reading materials have been investigated for a long time for interpreting written content (e.g., news, eBooks, and documents) efficiently; and most studies have employed actuators that use electrotactile braille pins, vibration, and auditory feedback to alert the VI with information such as obstacles and direction. These studies are typically based on wireless sensors such as Bluetooth, global positioning system (GPS), radio-frequency identification (RFID), and Wi-Fi [4].

This paper proposes a braille display methodology that transforms visual contents (e.g., text, photographs, and paintings) with audio feedback for multimedia information delivery to the VI. Although there have been several studies on braille display, most of them simply provide long text contents within single-lined braille cells. Such devices, however, cannot display paintings, pictures, and videos that are used in multimedia as well as in eBooks. This paper addresses such limitations and proposes improved multimedia and eBook accessibilities for the VI. Therefore, the paper introduces several considerations for displaying contents using a standalone braille display with multiarray braille cells. It also introduces an eBook reader application for the presented braille device.

This paper discusses the following considerations: (i) interaction between the braille display and the smartphone via Bluetooth [5]; (ii) extraction and translation of eBooks from DAISY and EPUB formats, and displaying large amounts of text from eBooks on multiarray braille cells; and (iii) displaying visual elements using multiarray braille display. A conceptual diagram representing the application proposed in this paper is shown in Figure 2. The proposed application comprises a tablet for displaying braille and a smartphone that processes the information. Through Bluetooth communication, braille and the manipulation data can be exchanged remotely. Additionally, in real time, the VI and the non-disabled can share their media information. In this paper, we introduce some related works that have been using tactile methods, along with the details of the proposed method. Section 2 explores research that can help the VI through diverse approaches. Section 3 describes the proposed eBook reader application in detail: it presents the implemented system on a smartphone as well as a tablet. Section 4 discusses the current limitations. Finally, Section 5 summarizes the research on the devices for the VI.
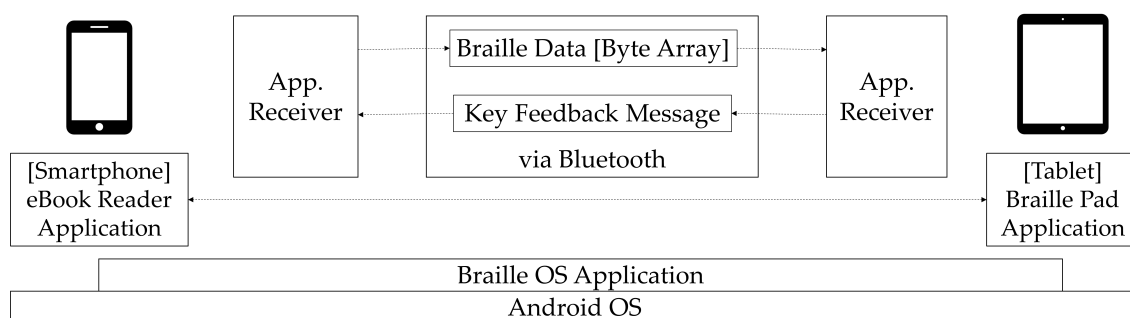
Figure 2. Conceptual diagram of the proposed eBook reader application.

2. Related Work

Most studies related to the VI have commonly used touch, vibration, and recorded voices to help them detect their surroundings. In addition, when information such as text content is delivered, braille or recorded voice is used to understand the context. This section presents a few previous approaches that have used tactile methods and the related research conducted on the devices for the VI. Although there have been several approaches for the VI using braille devices, there have been limited studies that include both braille display and their applications such as eBooks. Especially, there is no real-time braille device that can display 2D images in an eBook. Therefore, this field requires to be studied in greater detail, as it can be useful as an effective method for the VI to obtain useful information from books. The goal of the presented research was to develop a standalone braille display that can help the VI interpret media content. Thus, the primary purpose of this study was to develop an eBook reader application and display, and to simulate and implement the system on both smartphone and tablet. The braille display device mentioned in this study is currently under development as a business model. In this section, we introduce the existing braille applications, available braille devices in the markets, and innovative research for the VI.

Bornschein et al. conducted a study that uses tactile graphics to create figures using braille to introduce information effectively for the VI [6]. Several previous studies have focused only on the quantitative aspect of supporting the sighted graphics producers to speed up the direct transformation. However, this study focuses on the qualitative aspect, i.e., on improving the resulting tactile graphics based on braille display. The issues regarding the transformation of figures in devices for the VI are discussed in Section 3. The study by Byrd presented a conceptual braille display similar to the device proposed in this study, as shown in Figure 3 [7]. It comprises a 4×28 (*row $\times$ column*) light emitting diode (LED) braille display (without actuators), a Perkins-style braille keyboard, a scroll wheel, and navigation buttons. Moreover, it supports Bluetooth with PC so that information can be exchanged remotely. However, this study only considers braille hardware and does not consider software such as the operating system (OS) or applications that can run in the braille display. In contrast, the method presented in this paper focuses on the implementation of eBook reader applications based on multiarray braille display for the VI. Velazquez et al. studied a computer-based system that translates the eBook automatically into braille [8]. This method is called TactoBook, which is for displaying contents of eBook converted into braille and transferred on prototype devices via USB. While there are some inconveniences with transferring data to braille display by a separate storage device, it has simple interfaces for users and easy-to-carry advantages.

Additionally, several studies have been conducted to help the VI read books. For instance, Bae conducted a study to better serve DAISY, an eBook format for people with reading [9]. The study evaluates the ease of reading with other experts based on the service interface being provided to the disabled at LG Sangnam Library in Korea. These evaluation data complement the existing DAISY player (e.g., DAISY viewer) by re-designing and developing a new DAISY player. This study is useful in solving many of the inconveniences that previous DAISY viewers experienced. However, this device is not compatible or portable because it is only accessible to the PC used in the library.
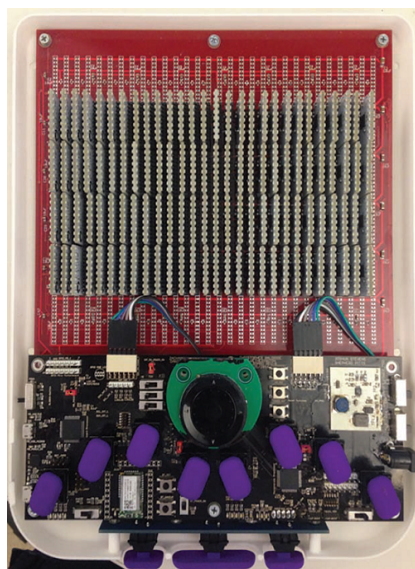
Figure 3. Final prototype internal components. The LED display is on the top, with the microcontrollers and keyboard below it. This Braille display was developed by senior students in North Carolina State University [7].

Kim et al. at Sookmyung Women's University implemented a DAISY viewer into the smartphone [10,11]. The DAISY v2.02 and v3.0 viewers are supported using smartphones so that people with reading disabilities can easily read DAISY eBooks anytime and anywhere. However, since blind people cannot use the touchscreens of smartphones, they cannot use this application. Goncu and Marriott produced an eBook image creator model based on iBooks (an iPad Application) [12] that utilizes graphic contents for people with low vision. Therefore, using graphical tools, viewer, and audio feedback using colors and shapes, the authors created images for eBooks that are easy for the VI people to understand. The authors argued that their proposed model can be combined with tactile displays in the future.

Harty et al. conducted research to allow access to DAISY-format eBooks using Android OS [13]. It supports DAISY v2.02 completely, while DAISY v3.0 is a beta version. The present study examined the DAISY v2.02 reader application based on this research. However, the beta version of the DAISY v3.0 player does not work properly; moreover, it is difficult to find detailed information about Android-based players that fully support DAISY v3.0 eBooks. Mahule studied the DAISY v3.0 application based on Android smartphones, which, however, has not been completely developed [14]: its parsing structure must be redesigned to adapt to the proposed eBook reader application for compatibility between the smartphone and tablet. This issue is described in detail in Section 3. Thus, several studies have been conducted on braille display and DAISY player, although few further studies have been conducted until date. Moreover, most studies have been carried out by companies that do not release useful information. Since there is lack of research about braille display and a need to set better research goals, this study introduces current well-known braille displays as shown in Figure 4 and compares them with ours in Section 5. We clarify that it is based on open information in their manuals and websites.

There are several considerations for VI people when choosing a satisfactory braille display. For example, a device having a screen for receiving instant feedback, the mobility of the braille device, Perkins-style keyboards, different braille cells (32 is the commonly popular choice), a support system to maintain them, etc. In this context, however, panning is the most important issue. Because, when a given text is too long to fit in the display screen, the user has to "pan" either to the left or right to continue reading along the line. For example, while using a 20-cell braille display to read a 72-character line, the reader will have to "pan" at least twice to read the entire text [15]. To overcome this problem,

we could either extend braille cells with a long braille display (>40 cells) or use a 2D multiarray braille display. Hence, while making the tactile graphic, this study chose the 2D multiarray braille interface.

Similar to this approach, Blitab is a tablet-type 2D braille display that was established by Blitab Technology [16]. It is the first ever braille tablet using a disruptive actuating technology to create tactile text and graphics in real time. Some people have called it "The iPad (tablet) for the blind", as it is a commonly used touch-type display similar to tablets, based on Google's Android, and a braille display, which consists of 14×23 cells that are connected as an integral part of Google's Android platform. Blind people can read information using 2D braille displays, and their assistants or teachers can directly check the tablet screen. However, the expected release date has passed and it has not yet been officially released. Therefore, it is difficult to get more detailed information because the user manual is not available yet.

| (a) | (b) |

Figure 4. Well-known braille display devices: (**a**) BrailleSense Polaris; and (**b**) BrailleSense U2 [17].

As mentioned above, there is very little information regarding braille displays, hence this study focused on a few released by well-known companies. The HIMS BrailleSense U2, which was considered as the best braille display at the time that can provide education, office work, and digital information for the blind. Today, the latest BrailleSense Polaris is gaining considerable popularity owing to its overwhelming performance. Based on compatibility with other devices, information can be shared through wired or wireless connection to any device such as PC, tablet, or smartphone [17]. Polaris can utilize gestures using braille cells and can also translate printed materials made of general text into braille. It supports more braille content than the other available braille displays. However, these devices have single-lined braille cells, which do not offer a comfortable reading experience for long texts and cannot print braille figures.

HumanWare Braille Touch is an assistive device that is equipped with touchscreen display and single-lined 1D braille cells [18]. The innovative design of the device allows the use of virtual keyboards on the touchscreen for blind people. Although blind people cannot see the touchscreens, they seem to be comfortable using a braille tablet. However, this device also has few cells and, hence, the text contents could be cut off, and, moreover, braille graphics are hard to display.

Table 1 presents a comparison of the above-mentioned products. Although these products have many features, this table simply describes their different specifications such as special features, cost, and operating system (OS). As shown in the table, the braille display is primarily being released on Google's Android platform. Furthermore, we see that, although Blitab displays braille images, it uses many cells. Similarly, although BrailleSense Polaris and U2 are specialized for class and office work, they are expensive. While BrailleNote Touch is also expensive, it is impressive as it includes a touchscreen keyboard that is not supported by previous devices. The prices of Blitab and the proposed braille display have not been set yet.

Table 1. Comparison of popular braille displays.

Product	Special Feature	Media Support	Braille Cells	Cost (USD)	OS	Release (Year)
Blitab [16]	Displaying Braille Image	Image and Audio	14×23	Unknown	Android	Unknown
BrailleSense Polaris [17]	Office and School-friendly	Audio Only	32	5795.00	Android	2017
BrailleSense U2 [17]	Office and School-friendly	Audio Only	32	5595.00	Windows CE 6.0	2012
BrailleNote Touch 32 Braille Notetaker [18]	Smart Touchscreen Keyboard	Audio Only	40	5495.00	Android	2016

Through coworking, this research conducted at Gachon University and Dankook University attempted to develop efficient, reasonable, and useful braille display for the VI. Park et al. developed a method for automatically translating scanned images and text contents from print books into electronic braille books by reducing the amount of time and cost required for producing braille books [19]. This method uses grayscaling, binarization, labeling, and filtering to import regional data for elements in scanned images. Furthermore, it reduces the time required for translating print books into braille books by maintaining the information recognition rate of the VI. Additionally, the authors of [20–23], contrary to the traditional methods of obtaining information, i.e., by using braille alone, attempted to transform accurate 3D information, 2D graphics, and photographic data to braille figures.

Finally, there have been some interesting studies that could be useful for the VI. For instance, Leithinger et al. studied a tangible media, inFORM system, for people who want to feel tactility remotely [24,25]; inFORM is a Dynamic Shape Display that can render 3D content physically so that users can interact with digital information in a tangible way [26]. This technology, named the Physical Telepresence technology, as shown in Figure 5, is useful in terms of "Tangible Information Processing" as it attempts to visualize objects remotely via a large number of actuators. However, the purpose of the development itself is not research for the VI and, hence, there is no consideration for their tactile recognition capabilities. In addition, methods for communicating color or images have not been included in the study. Dunai et al. developed a portable system that allows blind people to detect and recognize Euro banknotes [27]. As the VI cannot identify and calculate the value of their banknotes, this study used learning algorithms based on Raspberry Pi using no-infrared (NoIR) camera to classify the notes. Most such learning algorithms are based on neural networks to extract features of the banknotes and classify them efficiently. According to the authors, the classifying accuracy requires enhancement and must be extended to detect fake banknotes to avoid fraud.

Figure 5. inFORM shape display hardware. The inFORM system actuates and detects shape change using 900 mechanical actuators, while user interaction and objects are tracked with the help of an overhead depth camera. A projector provides additional visual feedback [24,25].

3. Proposed eBook Reader Application for 2D Braille Display

In recent years, various braille displays have been developed to improve the information accessibility for the VI. However, most of them are composed of braille cells in a single line and have the following typical disadvantages: (i) panning problem while reading an eBook; (ii) images or videos can be expressed only as text; and (iii) an assistant of the VI must know braille to sufficiently understand the multimedia contents. To address these problems, this study implemented an eBook reader application through a braille display that has multiarray braille cells. Therefore, to develop a more efficient and convenient braille device than the conventional devices, this study was jointly carried out by the Department of Computer Engineering of Gachon University and the assistive device development companies in Korea, namely Dot, and PCT. The braille display developed by the joint research team includes Braille OS, braille display simulator, and an eBook reader application, as shown in Figure 2. In this study, as shown in Figure 6, visual contents such as texts and images were expressed through a tablet having multiarray braille cells, while most computations related to the expression of visual/audio contents were operated in a smartphone. Thus, the issues discussed in Points (i) and (ii) above were addressed, and most of the text content, images, and videos were tactually expressed. Moreover, through the interoperability between the braille display device and regular smartphones, the issue discussed in Point (iii) was also solved.

Braille OS and braille pad simulator were the applications used by the respective devices, i.e., the tablet and smartphone. Braille OS was the OS used in the braille device, while braille pad was a simulator application for expressing the contents on the display. Braille OS has various built-in applications such as image viewer, calculator, games, and web viewer, in addition to an eBook reader application. Furthermore, this OS can be used in future Android-based braille devices. This study was performed by dividing the braille display into a simulator pad and a smartphone so that information could be exchanged through remote communication. In the future, the display and computation functions could possibly be combined in a standalone braille display, which could be separately linked with a smartphone. However, in this study, because the braille display has not yet been fully developed, the simulations were performed through a combination of a tablet and a smartphone.
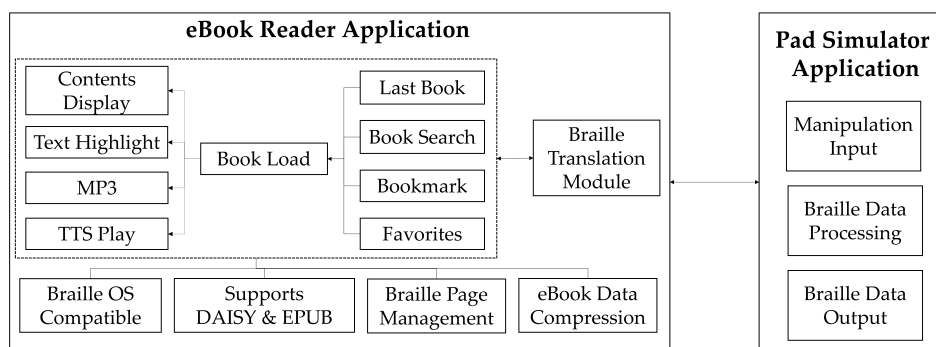

Figure 6. Detailed structure of the display in the proposed eBook braille reader application.

3.1. Features of eBook Reader Application

Figure 6 illustrates the structure of the eBook reader application proposed and developed in this study. This application was based on DAISY v2.02 and v3.0 of the DAISY Consortium [28] specializing in disabled persons, and international electronic publication formats, i.e., EPUB v2.0 and v3.0 [29]. Thus, this reader application supports both DAISY and EPUB formats. Furthermore, every eBook is implemented to be compatible with Braille OS while facilitating book compression and page management. In addition, it supports features such as Contents Display, Text Highlight, MP3, and Text-to-Speech (TTS), and the speaking speed and pitch of the TTS can be controlled. Moreover, the following functions are also provided: a function for searching a book stored in the smartphone, saving/opening a book that has been read most recently, the bookmark function, favorites function,

book/bookmark deletion function, and book compression/decompression function for efficient use of storage space.

All contents were translated to braille and sent to the braille pad, and the braille translation was performed by using the braille translation engine module developed by Dot. This braille translation module can translate both English and Korean. In addition, Figure 6 and Table 2 present the sequence of reading a book after opening the eBook reader application proposed and developed in this study. Here, the contents displayed by the braille device refer to expressing text, image, or audio file. When the contents are displayed, the bookmark list page facilitates saving and fetching a bookmark; however, the favorites option facilitates only the save function, and the favorite list page can be browsed by going back to the first/main page. Moreover, using the backspace, the user can go back to a previous screen in the reverse sequence of entry.

Table 2. The sequence of braille eBook reader application.

Sequence	Activity and Results
1	When an eBook reader application is run through the Braille OS, a book is opened by searching the corresponding book in the smartphone storage, or through using the open last book, bookmark, or favorites placeholders
2	When an MP3 file exists for a book, a selection is made to open the MP3 and text content. If not, the list of chapters in the book is opened immediately.
3	With the help of the chapter list, the multimedia contents can be read chapter-wise.
4-1	When the text player is used, the text contents are translated to braille, which can be used together with the text-to-speech (TTS) feature.
4-2	When the MP3 player is used, the play/pause mark is displayed on the pad screen, and the content is automatically played (in this case, TTS is not supported).
5-1	For both the text player and MP3 player, bookmark favorites setting is facilitated and saving, and fetching are possible.
5-2	When there are images in the braille page, the corresponding image file can be read from the image list and viewer.

3.2. Communication between Braille Display and Smartphone

Braille OS was installed on both the smartphone and tablet, and the messages produced on the eBook reader application and the braille pad simulator were sent and received between the two devices. For this, the Android OS Message object was used, which is provided by default in the Android OS, and is based on Bluetooth [30]. This object facilitated the sending and receiving of messages between devices via a handler, and the key code used for a message was defined in its own library in this study. The key code was transmitted from the braille pad simulator and can be controlled by the VI user. The simulator has a total of 21 buttons and can control the smartphone remotely. Each key consists of several bits. When a key is received in the smartphone, certain operations are performed. Each key can be used independently as a single key, and as other braille displays, a combination of keys can also be used and transmitted. Moreover, by using a long click, a corresponding defined operation can be performed.

Braille OS is a background app, and, when the on/off button is pressed on the pad, the Braille OS application activity [31] pops up on the smartphone, after which the user controls the smartphone remotely through the braille pad. Next, the user opens a desired application in the braille application list. The mentioned application, i.e., the eBook reader developed in this study, can be selected among other applications. Next, the role of the eBook application is performed, as shown in Figure 6. For every task, an operation is performed in the smartphone, which is controlled by the braille pad, and the list and text contents expressed on the smartphone, as shown in Figure 7a,c. Thus, the contents of the eBook are expressed as the contents of the braille display shown in Figure 7a. All this information is computed on the page of the corresponding stage in the smartphone and sent in byte arrays; the reason

for sending information in byte arrays is because a single braille cell has eight dots, which have binary values, and is expressed in a byte. Since the display comprises 12×12 multiarray braille cells, the number of corresponding braille cells is 144; and to efficiently express this on each braille cell, a transmission unit of a byte array is used. All contents are delivered to the pad through the braille translation module, as shown in Figure 6.
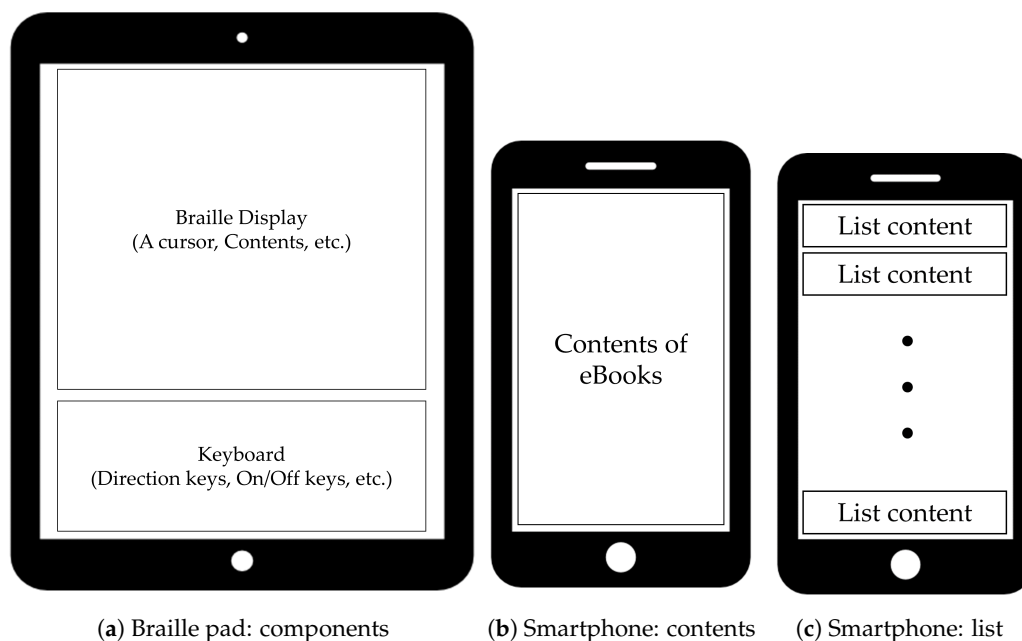

(**a**) Braille pad: components　　　(**b**) Smartphone: contents　　(**c**) Smartphone: list

Figure 7. Screen components of tablet and smartphone.

3.3. Parsing Method of Multimedia Contents for the VI

The formats of the eBooks, i.e., multimedia contents, used in this study are DAISY Consortium's DAISY v2.02 and v3.0, which are book formats for disabled persons, and EPUB v2.0 and v3.0, which are the most common eBook formats used worldwide [28,29]. As shown in Figure 8, this paper presents the process of extraction and translation of an eBook's text into braille data. It also introduces the sequence using open sources and libraries to simplify the implementation of the braille display. In this section, we introduce more details regarding the formats included in an eBook, which require explanation.
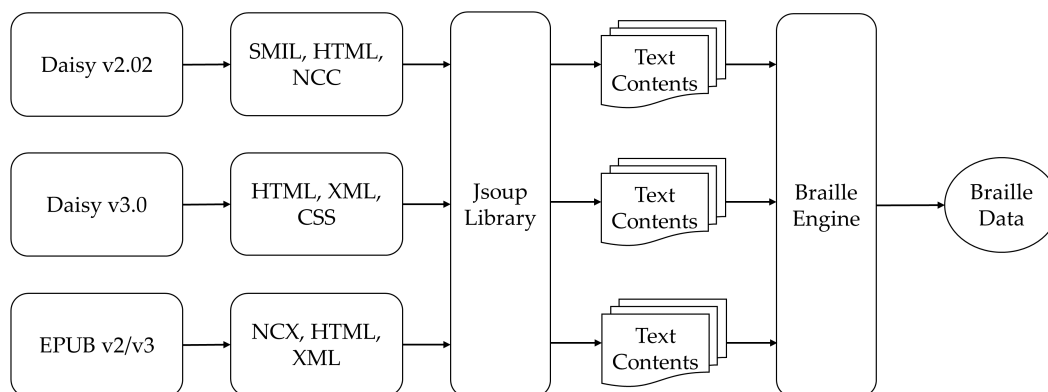

Figure 8. Text extraction and translation sequence of proposed eBook reader application.

The open source of DAISY v2.02 was developed by referencing the Android Daisy Reader [13], which is relatively more commonly implemented, and the open source of DAISY v3.0 was developed by referencing the Daisy3-R eader [14], which is relatively less implemented. These open sources were partially used to extract the contents of an eBook. However, because these open sources could

not be perfectly implemented for parsing all the contents in the DAISY books, additional tasks were performed in this study for the following reasons: (i) compatibility with the application of this study; (ii) extraction of all contents of the eBook; and (iii) suitability for mobile applications. With respect to DAISY and EPUB, the text, MP3, picture, photograph, and book information can be extracted for each book based on the markup language using tagging [32]. To extract such a large amount of information, parsing for the respective formats is required. Therefore, this section mainly explains the parsing of the DAISY and EPUB formats using open sources and libraries.

3.3.1. DAISY Formats

Figure 9 shows an example of text contents extracted from DAISY v2.02. In the relatively simple DAISY v2.02 format, there exist the Navigation Control Center (NCC) document and Synchronized Multimedia Integration Language (SMIL) file corresponding to each chapter or section and the text, images, and audio contents are tagged in the SMIL file [33]. On the other hand, books in the v3.0 format have less complex structure and can support more image and audio files, which was achieved by enhancing the navigation function one level higher through the Navigation Control for Extensible markup language (XML) file (called the NCX file), and strengthening the multimedia functions [9,34]. Table 3 shows the file types existing in the DAISY v3.0 format. This structure of DAISY v3.0 is similar to the book structure of EPUB explained in [35].
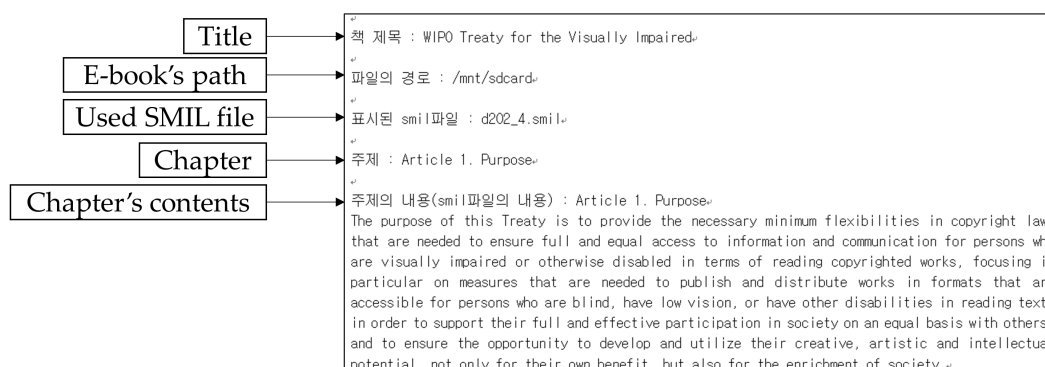

Figure 9. An example of extracting sample content using SMIL and NCC file within smartphone (Android OS). "WIPO Treaty for the Visually Impaired" is the DAISY v2.02 sample book of DAISY Consortium [36] and extracted by modifying [13].

Table 3. The contents of DAISY v3.0 format.

Format	Contents
OPF	Open eBook Publication Structure (OEBPS). It includes bibliography for each book
XML	Text content file containing some or all text in the book with the appropriate markup
NCX	A file containing all the positions of a book that users can browse
SMIL	A file containing information to link audio and text content files
RES	A file containing text segments, audio clips, images, etc., representing the movement information
Others	Audio, image, CSS

The approach of DAISY v2.02 in this study is a typical method that extracts tagged contents by accessing the NCC file. However, because v3.0 had a compatibility problem with conventional applications during the implementation of the application, various contents were extracted using a direct tag-parsing method via XML files, instead of going through the NCX file. In a previous study [37], the open source Android Daisy Reader was used to extract text contents and additionally extract images and MP3 files that can be used as information contained in a book. Through this open source, DAISY v2.02 can browse the SMIL file and its lower level files based on the NCC document. However, when the application is implemented via the open source used in this study, certain chapters

in the book were omitted because the content types were differently entered in some DAISY v2.02 files. This problem occurred because, for every author, as well as for every DAISY v2.02 book, the structure is different, and the chapter list did not match correctly with the text contents. Besides, even when there is corresponding content for a certain chapter, either a blank page, or the text of a different chapter was displayed. To prevent this problem, we applied Algorithm 1 to properly extract the entire text in the original eBook by inserting conditional statements that use three types of categories (i.e., Level, Pagenumber, Unknown), and by creating a chapter-wise index of the text contents.

Algorithm 1: Chapter-wise extraction of text contents from DAISY v2.02 books using an open source [13].

Input : The DAISY v2.02 book
Output: Text contents and appropriate chapter index of the DAISY v2.02 book chapter.

DAISYItemType LEVEL, PAGENUMBER, UNKNOWN

for *0 to length* **do**
    if *navi.getType() == LEVEL* **then**
        add to text contents
    else if *navi.getType() == PAGENUMBER* **then**
        length++
    else if *navi.getType() == UNKNOWN* **then**
        length++
    else
        Add to text contents
end
Insert size of chapters to the index
return text contents and chapter index

These three types were declared as enumerated types in the Android Daisy Reader. level category indicates where the real text content is located, and pagenumber category is separately specified by the author regardless of the text content in the book. Finally, unknown category is the assigned blank type, and there is unassigned space besides these three types; all the text is properly filtered with these branch statements excluding the useless parts.

On the other hand, DAISY v3.0 does not go through the NCX file during the implementation of the application, and extracts contents properly via the tag-parsing method that searches directly below the NCX file. Because there is not much information available for the method of extracting using DAISY v3.0, as compared to v2.02 and EPUB, it was implemented by referencing [34]. This study extracted text contents partially by using the Daisy3-Reader [14], and, going one step further, the information was directly extracted through tags to obtain the information compatible with the conventional studies. Thus, each chapter of a book, and the text contents, audio files, and image files corresponding to that chapter were extracted [35]. Furthermore, as for the text output method, the Document Object Model (DOM), which is an API standard, is used to extract XML and Hyper Text Markup Language (HTML) documents in the application.

The following tag-parsing method was used in this study: (1) The XML file validity test was performed for a DAISY v3.0 book. (2) The Level of the XML document was extracted by searching with the depth attribute. (3) When an author creates a DAISY v3.0 book, the eBook file is composed by classifying parts, chapters, and sections by using the level of the header tags. Therefore, every tag was collected and then, the chapter list and all the contents of the book were extracted. Since this method outputs an XML document without going through the navigator, the computation speed is faster, while maintaining the compatibility in this study. For example, because the method of DAISY

v3.0 open source [14] using previous NCX does not perform the optimization properly, it takes several seconds to perform the braille translation and move to the next chapter in the eBook reader application used in this study. On the other hand, the braille translation of the text content using the tag-based parsing method outputs the text contents in less than 1 s, which is relatively much faster.

3.3.2. EPUB Formats

Using Epublib, an open library that supports EPUB v2.0 and v3.0, a book in the EPUB format was extracted on the Android OS. As shown in Figure 10, Epublib is designed to facilitate extraction of the title, subtitles, and main contents of the EPUB book using tags [38]. Through the Epublib library that is converted into the Android format, parsing is performed on the Braille OS by following the usual EPUB parsing method using NCX [39]. In contrast to the open source of DAISY v3.0, the method using Epublib can extract all contents of the EPUB based on the HTML document without any compatibility problem with the application. Furthermore, because it has been developed for a long time, this open source has been well optimized and debugged. The extraction method is also very simple, and the contents can be extracted in the HTML format. Furthermore, EPUB and DAISY are based on the style of a HTML document. Therefore, to extract text contents from books in these formats, a process of extracting text from HTML is required: using the Jsoup open library, tags of corresponding contents are parsed, and then, all contents are extracted [40]. After extracting all the contents, they are first classified into text, audio (MP3), and images, and then allocated to the respective player or viewer.
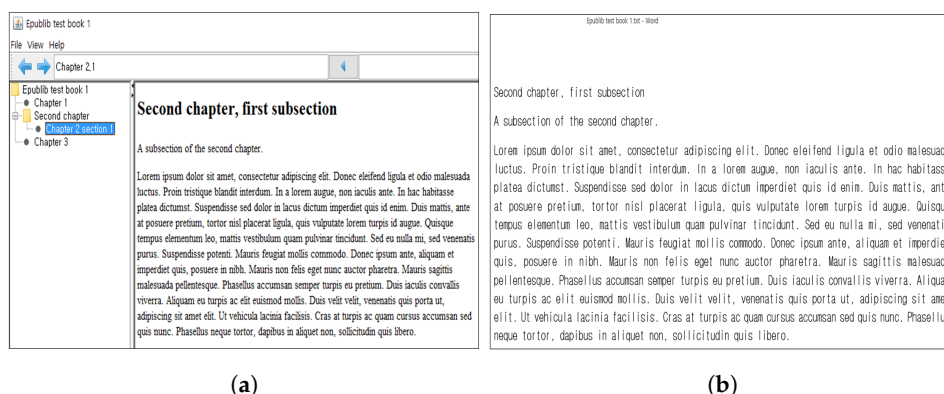

(**a**)   (**b**)

Figure 10. Sample Epublib contents within the java language platform: (**a**) EPUB viewer by Epublib; and (**b**) extracted text contents using Epublib.

3.4. Components and the Design of 2D Braille Display for the VI

As explained in the previous sections, this study describes the methods of managing and outputting pages through multiarray braille cells, playing or outputting TTS or audio files, and extracting and outputting an image file. We tried to express braille on the multiarray braille display, which is not much different from that of the other braille devices, and fit as many braille words as possible into a page without splitting a word. For example, if a word is split between pages instead of showing the complete word on a same page, there will be recognition problem, and VI user may become confused while turning pages; the user may have to go back to the previous page to read the split word again.

3.4.1. The Proposed Method of Page Division for 2D Multi-Array Braille

The method of expressing braille on the multi-array braille display developed in this study is shown in Algorithm 2. This algorithm uses the text contents of the DAISY v2.02 book saved by running Algorithm 1 sequentially. The text contents, which are organized chapter-wise, are converted into braille. First, the carriage return characters and spaces in the text are deleted, and the words are inputted into the array. Second, the divided words are converted into braille, stored in the braille buffer page. When the converted data exceed the size of the braille page, the buffer page is flushed to

a braille page. Finally, if the book is translated to braille until the end, the finished braille translation data are saved and outputted on the braille display. In this algorithm, the rows and cols (columns) refer to the horizontal and vertical cell sizes of the braille content display, respectively, and a word is distinguished and added to each row of the braille array. Through these rows and cols, the maximum amount of text that can be seen on the braille display is expressed. Next, the content of the buffer is saved, and all the variables used for allocating a page are initialized to prepare for the next braille page.

Algorithm 2: Conversion of text contents to braille contents from DAISY v2.02 books.

Input　:The text contents of the DAISY v2.02 book
Output:Braille pages and its corresponding text pages of the DAISY v2.02 book

for *0 to length of the book contents* **do**
    Initialize braille rows and a braille buffer page
    if *translated braille contents + additional one of translated word > max cells* **then**
        Add the additional word to text contents
        Translate additional words
        for *0 to rows* **do**
            for *0 to cols* **do**
                if *rows * cols + cols < temporal data of translated words* **then**
                    Allocate translated words to braille buffer page
                if *temporal byte data is flushed* **then**
                    Adjusting index of the byte data to the next
                else
                    break;
            end
        Save translated braille data to braille buffer page
        Flush the braille buffer page to the braille contents page
        Initialize braille page buffer and braille page data
        continue
    end
    else
        Add a word to temporal text contents
    if *translated braille contents + additional one of translated word > max cells* **then**
        Initialize braille rows and a braille buffer page
        Add the additional word to textContents
        Translate additional words
        for *0 to rows* **do**
            for *0 to cols* **do**
                if *rows * cols + cols < temporal data of translated words* **then**
                    Allocate translated words to braille buffer page
                if *temporal byte data is flushed* **then**
                    Adjusting index of the byte data to the next
                else
                    break;
            end
        Initialize the byte count
        Add the translated braille contents to the braille page
    end
end
return Both braille contents and corresponding text contents per braille page

---

In this study, Algorithm 2 played a crucial role in managing the multiarray braille display to express the text contents. Furthermore, even if the braille display size is changed or the source language of the braille translation is changed, the braille array is assigned to fit the cells in accordance with the byte array by assigning the variables through this algorithm. The outcome, i.e., the braille content, refers to the contents that were translated into braille to create the pages in the braille display, and using the same content, the saved braille text contents are outputted in accordance with the braille page. Initially, the required variables are declared for the respective screen and lines that will be expressed in the display cells. Through this, the contents of the book are divided into words and converted into braille. Subsequently, they are pasted again as the main text of a book. Thus, all the contents of the book are expressed in braille appropriately by fitting in the screen.

According to the conditional statement at the last part of algorithm, if the maximum cell value is exceeded, only the last word is added to the buffer and a braille page is produced, and then a next page is created; this process continues until the end of the book. Furthermore, if the contents are longer than a braille line, there might be more text contents. In such cases, the smartphone shows a long text by using the design setting of activity, but in the case of braille, the user might find it difficult to read if the page is automatically turned. Therefore, by expressing the cell line in a fixed format, a method of changing the text content with a cursor is chosen. With respect to the splitting phenomenon occurring because of insufficient horizontal cells on the screen, the next/previous text content is shown by pressing the left/right button using a cursor on the corresponding braille page, and the data can be revised to fit the braille page by using the dynamic allocation of an ArrayList in Java. Furthermore, because the part of the content being read by the user is automatically highlighted in the smartphone, a teacher, guardian, or assistant who is not disabled can easily track the activity of a VI person and provide appropriate guidance.

The present study used a braille display that has a 2D multiarray display similar to that shown in Figure 11 to effectively express multimedia contents for VI users. The braille display consists of 12×12 cells in total, as shown in Figure 7a. The column (11×1) at the far left indicates the user's cursor position, and the contents cells (11×11) at the right side show the multimedia information received from the smartphone. Excluding these, the last line of the cells (1×11) at the bottom expresses additional information such as page and book information. As shown in Figure 7c, when the cursor points to a content in the list, a total of 11 list contents are displayed, and if there are more than 11 list contents, the list pages are managed automatically to move to the next or previous page; and the screen of the corresponding list activity is appropriately displayed.
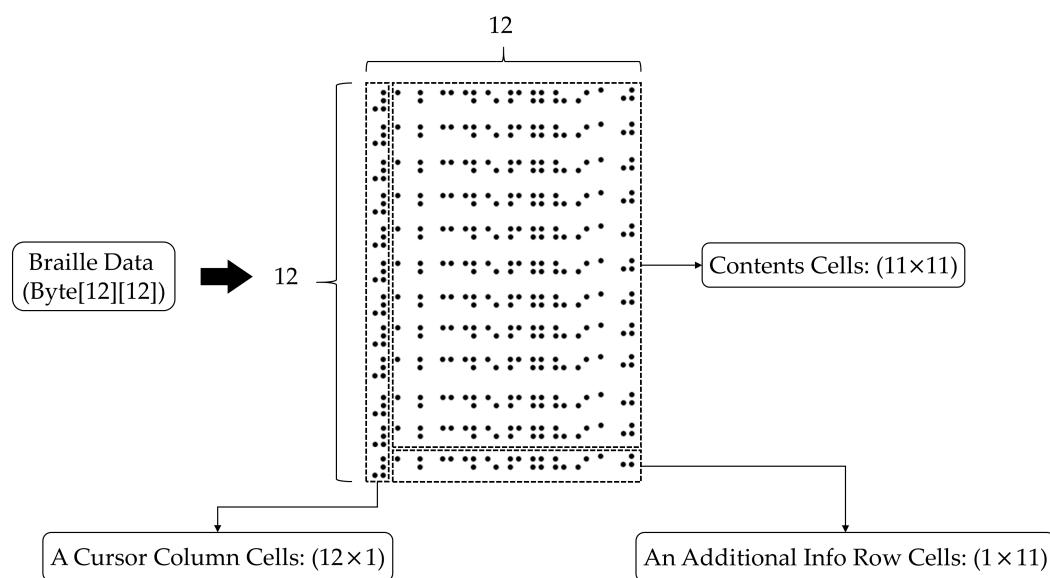

Figure 11. Example of a braille display presented in this study.

3.4.2. Setting Options for Reading the eBook by Audio, and Shortcut Function for the eBook Reader

The keyboard design and allocation of buttons are not so different from those shown in Figure 3. The arrows located on the upper left side can be used to increase or decrease the volume of the MP3 or TTS. By long-clicking these, the speed of the TTS can be increased or decreased. Right beside the multiarray display, there are keys to navigate between pages, i.e., to go to the next or previous page. In addition, there are six function (F) keys, a backspace key, an enter key, a space key, the up/down/left/right keys, a shift (S) key, a control (C) key, a power on/off (P) key, a TTS (T) key, and a volume control key. The respective keys perform various roles such as playing or pausing TTS/audio files and saving/fetching/deleting bookmarks.

An audio is the content feedback that is as important as braille. In this study, if a cursor was moved and touched a cell line in the case of list, the user could listen to the corresponding content by using the TTS function through the smartphone; in the case of a large text such as a book, the part that the user is currently reading is read by the TTS. The VI user can listen to the audio at any time using the setting of TTS saved by him/herself. In the TTS setting, the sound volume and speed can be controlled. The speed can be set to 0.1 times minimum and 2 times maximum. The sound volume is set the same as that of MP3, and it can be controlled by using the multimedia sound volume setting basically provided in the Android OS. Furthermore, an MP3 player is implemented so that if there is a parsed MP3 file in a corresponding eBook, it can be selected when reading the book. When an MP3 file is played, the recorded audio is played and on the braille display, the play and pause marks are expressed so that the user can understand the playing status. Moreover, in additional contents braille display, additional information of corresponding braille page and book is extracted and outputted.

Furthermore, by using the bookmark function, the user can save a braille page of the corresponding text in a book. Favorites is similar to bookmarks, but, in this case, a book can be saved and opened immediately. A bookmark or favorite is saved in each bmk (bookmark) file format and can be deleted by the user, if required. The compression function compresses all the eBooks in background operation and by saving it in the memory buffer temporarily, it enables the user to read the book without any problem of retrieval. If the book that the user was reading is closed, it is automatically compressed again, and the book that the user has been reading is automatically saved by using Shared-Preference, a data saving method of the Android OS. This is a simple method that uses a database to save the book in the Key-Value format, and until the application is deleted, it is kept as an internal XML file of the application [41].

3.4.3. Braille Image Conversion on Braille Display for the VI

There have been collaborated studies on braille figures, which presented novel methodologies for translating texts and graphics to braille. Figure 12 explains the translated braille image in a multiarray braille display based on such studies [19–21]. Generally, braille graphic experts consider three factors while constructing braille graphics: image complexity classification, a tactile graphic translation of low- or high-complexity images, and image segmentation. For image complexity classification, simple graphic images are defined by graphs and tables. On the other hand, pictures and paintings are distinguished as intricate images for processing. A comparison of the gray level histograms for simple and complex images shows a clear distinction between the two. Generally, simple images such as charts and graphs have drastic changes in their gray level pixels unlike complex images that have slight changes between their gray level pixels. Therefore, high-complexity images cannot be expressed in braille perfectly; and extracting and displaying a central object on the braille display is an efficient method.

Especially, the main reason for extracting only the central object is explained in Table 4. According to Park et al. [19], blind people find it difficult to recognize braille images without central objects. Because VI users usually begin from the top-left side of the braille display and move towards the bottom-right side, it is difficult to understand the high-complexity images in the braille display at a glance, although they are translated very well. In addition, they could not distinguish other image features such as color and texture through touch. Therefore, the edges of the central object,

which can be conversed by braille, were extracted and presented on the braille device. The central object calculates the color differences in one image and labels them for extraction. Additionally, it is assumed that the central object is located at the center of the image, and is detected through the following steps: (i) simplify color values by quantization; and (ii) obtain the color similarities and extract color values divided by each color in the image.

Generally, based on the photography and characteristics of the triangulation, the largest area within these labeled areas is designated as a central label. However, the central object may not be able to capture this properly because it has a small area in the central label area. In this context, especially because of the large area of the landscape, a horizontal or vertical aspect of the object can lead to errors that cannot be captured correctly by the central object. This is done by calculating the variance by the size and average of the horizontal or vertical pixels of the central object located in the central label area and matching with the decision-making values. In this study, the decision-making values were directly extracted by five tactile graphic experts. The tested figures are in textbooks for high school students. Subsequently, the edges of the central object area were converted to binary data and the borders of the binary figure were extracted.
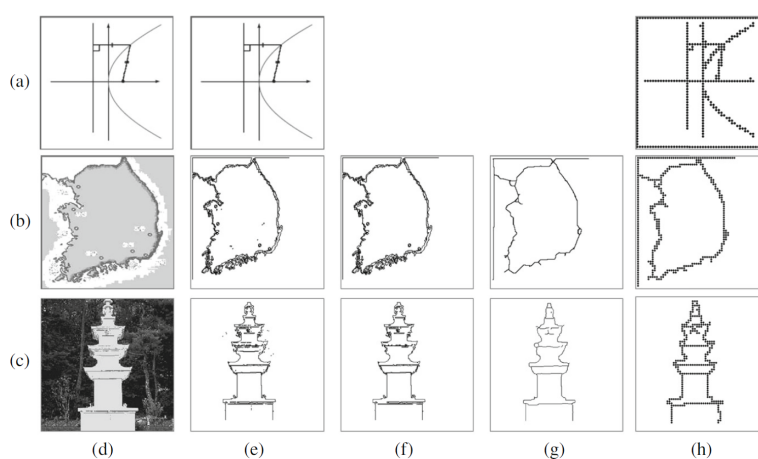

Figure 12. Steps in the conversion of a graphic image to braille cells: Row (**a**) translation of a graph (a low-complexity image); Row (**b**) translation of an illustration (a high-complexity image); Row (**c**) translation of a photograph (a high-complexity image); Column (**d**) results of quantization; Column (**e**) results of object detection; Column (**f**) results of first simplification; Column (**g**) results of second simplification; and Column (**h**) final output in the form of braille cells [19].

Table 4. Survey results on tactile graphic recognition characteristics of the VI (15 blind, 10 with low vision) [19].

Item	Recognition Characteristics
Image details	Expressing excessive detail can cause confusion in determining the direction and intersection of image outlines. Therefore, the outline in both low-and high-complexity images should be expressed as simply as possible to increase the information recognition capabilities.
High-complexity image with a central object	For an image containing a primary object, the background and surrounding data should be removed and only the outline of the primary object should be provided to increase recognition capabilities.
High-complexity image without a central object	For an image without a primary object, such as a landscape, translating the outline does not usually enable the VI to recognize the essential information.

In this binary image, there are some irregular sharp lines in the contours, which are unnecessary and disturb the lines of the edges; these should be simplified because of the recognition problem outlined in Table 4. Expansion and segmentation were conducted on the binarized image to create a

simplified contour image. Next, the corner points were extracted using the Shi–Tomasi algorithm [42]. Thus, the unnecessary corner points were removed to simplify the contour image. This simplification method increases the recognition rate of the central object. According to the survey (Table 4), expression of the tactile graphics must be simple, failing which the information might not be recognized by the VI users.

Normally, since image sizes vary in most cases, a few of considerations, such as the size of the designated braille display and up/down scaling of the central object, must be taken into account. Moreover, while calculating the size and number of the cells, the vertical dots are twice as many as the horizontal dots (4×2 dots). However, although we designed braille cells with eight dots, we tested them as six dots and considered the vertical dots as 1.5 times the horizontal dots (3×2 dots). In addition, in this study, various braille display sizes have been considered for printing letters and figures: ranging from the small size of 7×7 cells to the current size of 12×12 cells, The braille display in this study was aimed to fit in the palm of a hand like a smartphone. Therefore, the 12×12 cell display was considered as the appropriate braille size involving a small keyboard. However, the resolution for the braille display is important for the VI in terms of image recognition.

This can be easily understood through a comparison with the resolution of an image/video viewed by a non-disabled person: for example, it is like the difference between watching a soccer game in Standard Definition (SD) resolution and High Definition (HD) resolution. (An SD image has 720×480 -pixel resolution, which is expressed as 345.6 K pixels, whereas a HD image is expressed as 1049 K pixels.) The number of braille cells in the display device used in this study is 12×12 but, as shown in Figure 11, only 11×11 cells are used because the contents are expressed only on the contents' cells. Therefore, an image can be expressed with a total of 968 dots composed of 121 braille cells (eight dots per braille cell, and in the case of six dots, the number of dots equals 726). Because of this, braille cell resolutions and the image quality are proportional, but there is a trade-off with the recognition rate of the user, and the supplying price of the braille display. In this study, grounding of braille and fingers was considered. If a braille cell size is approximately 10×5 mm, for a 2D braille display, then the recommended number of braille cells is greater than 7×7 for recognition of symbols or objects. In this study, we attempted to use many multiarray formats ranging from 5×5 to 12×12 to express various contents, but it has no choice to choose the maximum braille display size. In this case, many braille cells make remarkable conversion.

4. Implementation and Results

This study developed an efficient 2D multiarray braille display that can replace heavy and inefficient devices and express the content simply and clearly for the VI users. We attempted to eliminate the disadvantages of the existing devices for the VI and provide convenient use while maintaining the as many features as possible of the regular devices for VI users. We also paid close attention to the interaction between VI users and their assistants/guardians, and to enable smooth information sharing. Thus, we developed a braille display software that can share information with a smartphone. However, we could explain the implementation only through the images of the braille display screen because of the design copyright issues of the braille device that is under development. In addition, we obtained much practical feedback from the VI for developing the proposed system [19–21]. Thus, we focused on the braille display software only and presented this system thoughtfully.

In this study, we used a combination of a tablet and smartphone to replace the braille display, as shown in Figures 6 and 7. The tablets used were Samsung Galaxy Tab S and Galaxy Tab S2 8.0, and the smartphones used were Samsung Galaxy S7 Edge, S6, Note 5, and LG G4. Their operating systems were all Android 6.0 Marshmallow, and the development was carried out using the minSDKVersion of 19 of the Android API. For the SDK, Android Studio was used. In addition, the development was carried out on a laptop computer with Intel i3 CPU, 4 GB RAM, and SSD of 500 MB/sec read/write speed. The sample files used for the DAISY books were provided by DAISY Consortium and the EPUB sample files were provided by PRESSBOOK [36,43].

Figure 13 shows the text and its corresponding braille translation while exchanging data through the Braille OS via a Bluetooth connection between the tablet and smartphone. Figure 13 shows the braille contents that a user reads on the tablet, which is a text from DAISY v2.02 translated to braille, and the tablet screen is modified to show only the contents of the multiarray braille. Figure 13 shows screenshots of the tablet and smartphone screens, which is an example of the previously explained braille pad and Korean contents translated into braille. Figure 14 shows examples using an eBook of EPUB format. Figure 14a shows examples using an eBook in the EPUB format. In Figure 14a, we see that all chapter lists of the sample eBooks are normally outputted. In addition, Figure 14b shows that a text content of the eBook is normally outputted via the text player.
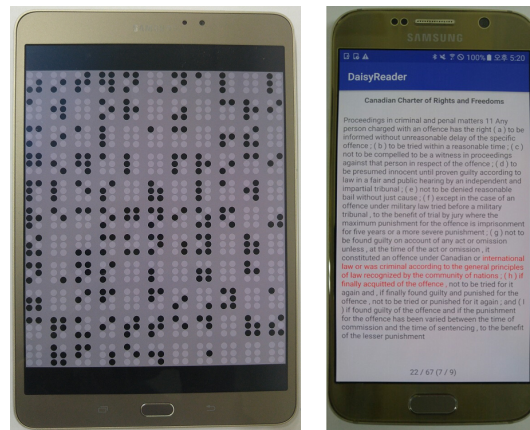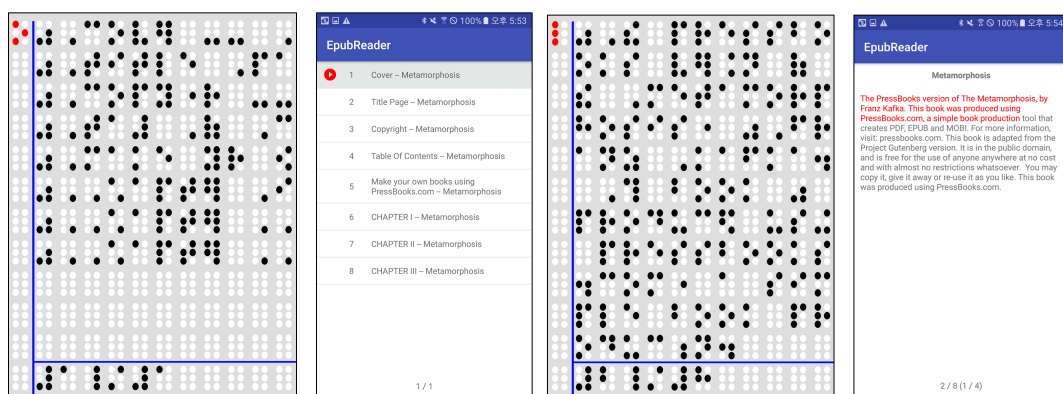

(**a**) English
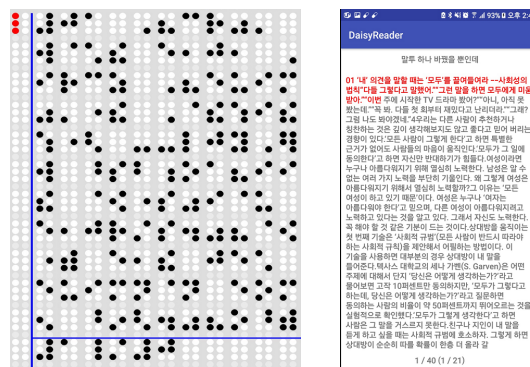
(**b**) Korean

Figure 13. Text player with DAISY v2.02 (English) and DAISY v3.0 (Korean) eBook.

(**a**) Chapter list of an EPUB file

(**b**) Text player of an EPUB file

Figure 14. Sample eBook in EPUB format.

Figure 15, using a smartphone for easy understanding, shows the sequence of the processes performed when a VI user is reading a book. In the main page, search for books, open last book, or favorites can be selected. When open last book or favorites is selected, the text player is executed immediately; and when search for books is selected, the book list is opened, and this shows the list of books in the smartphone storage. Moreover, if the searched book contains audio files, the MP3/text mode can be selected, and the respective mode is executed. In this study, they are operated based on chapters (usually MP3 files are divided chapter-wise). According to the selected part of the book, the book content is played.
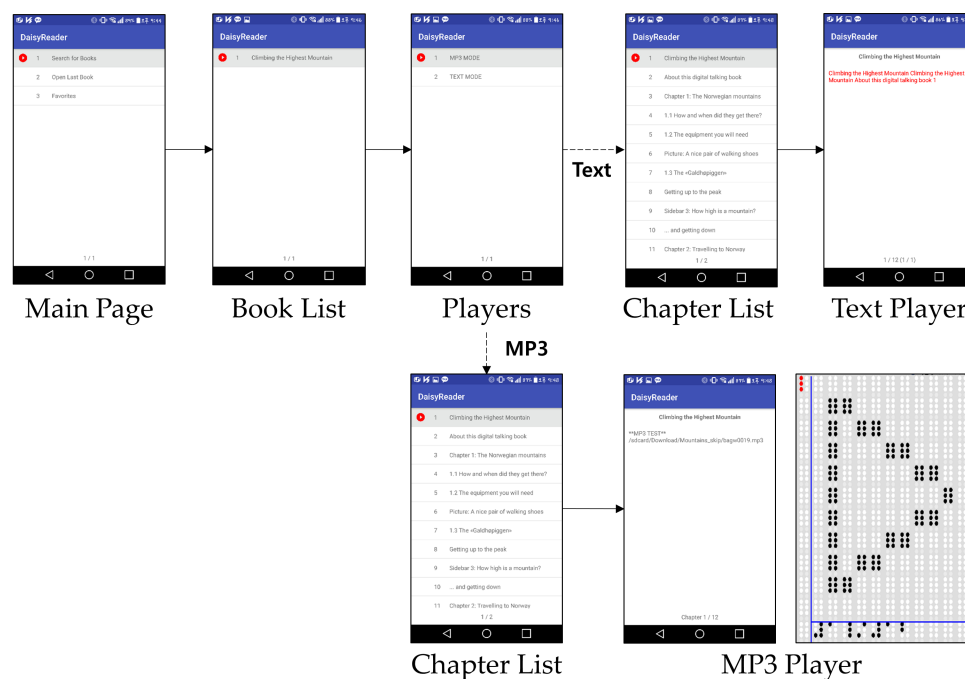

Figure 15. Process of the eBook reader application

Park and Jung et al. [5,19–23], who are co-researchers of this study, performed investigations pertaining to the processing of images in books: when a user presses a certain key in the image viewer, the book image list is displayed in response. In this study, a tag is attached to an image file in a chapter that a user is currently reading, it is tagged and extracted, and an image file list is created. When the user enters the image viewer through the list, image-oriented objects are extracted and displayed. As shown in Figure 16, there are a/an graph, illustration, and photograph displayed with high-resolution 2D multiarray braille display.

Table 5 compares related works. Some of them focus on people who are either blind or have a reading disability. We see that the DAISY format has been investigated in most studies, especially for people with reading disabilities. Unlike EPUB, DAISY is an essential format for the VI users. However, Bornschein et al. [6] and Goncu [12] did not develop the DAISY reader. EPUB has been excluded from most studies, although, as mentioned above, EPUB books are the most representative eBooks and are abundantly available. Thus, it is very good feature supporting EPUB eBooks. Only Bornchein's and the proposed methods consider braille translation and braille figure expression among these studies. In the case of mobility, studies conducted on mobile OS such as Android and iOS show a positive mobility. However, only the method by Bornschein and the proposed method in this study have considered braille translation and braille figure expression among these studies. Furthermore, since studies based on mobile OSs such as Android and iOS have reported positive mobility, this study deeply examined the DAISY reader application for VI users, in addition to supporting EPUB books, printing translated braille, braille figures, device compatibility, and portability through Android-based OS, as shown in Table 5.
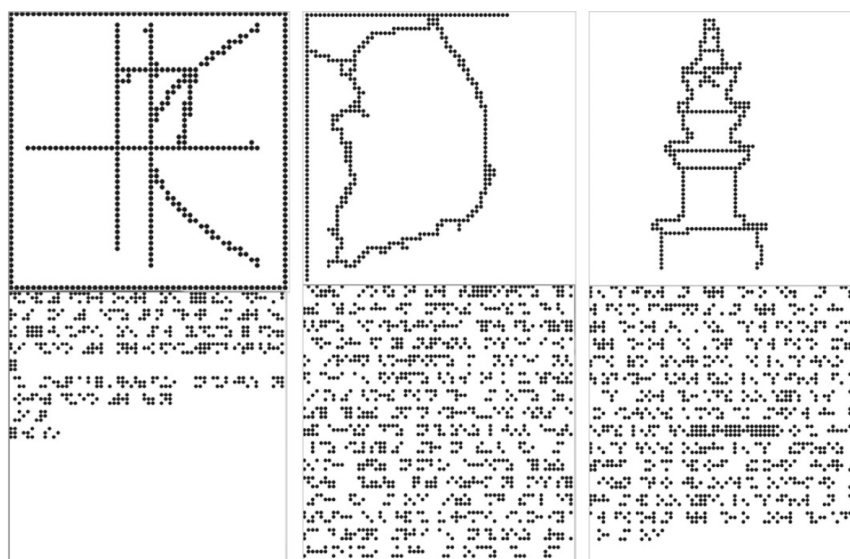
Figure 16. Implementation of figure expression using multi-array braille (50 × 50). Braille pages created after positioning and combining translated braille and tactile graphics [19].

Table 5. Comparison of the introduced assistive applications in this study.

Method	DAISY	EPUB	Braille Translation	Braille Image	Mobility	OS
Bae's [9]	v2.02, v3.0	×	○	×	Weak	Windows
Kim's [10,11]	v2.02, v3.0	×	×	×	Strong	Android
Bornschein's [6]	×	×	○	○	Weak	Windows
Goncu's [12]	×	○	×	×	Strong	iOS
Harty's [13]	v2.02, v3.0	×	×	×	Strong	Android
Mahule's [14]	v3.0	×	×	×	Strong	Android
Ours	V2.02, V3.0	○	○	○	Strong	Android

This study attempted to go beyond the applications listed in Table 5. Unfortunately, however, the braille device has not been developed yet. Nevertheless, to reduce the existing issues in the currently marketed braille displays, which are the original objectives of our study, Table 6 compares the important factors discussed herein. Blitab [16], which has attracted the attention of not only the VI but also the media although it has not yet been marketed, has a mild panning problem as a 2D braille display. Therefore, images can be presented and real-time interaction with other non-disabled people is possible with the screen on the device. However, it is not known whether DAISY and EPUB are supported because there are no details yet. BrailleSense Polaris and U2 [17] are among the best braille displays currently available in the market, offering various compatibilities for audio, video (extracted audio), document, and eBook formats. However, due to the limitations of their 1D braille display, they have the panning problem and, hence, cannot display braille images. However, they have stronger compatibility than the other devices. Moreover, they require enabling real-time sharing with non-disabled people indirectly using a smartphone, tablet, and PC. BrailleNote Touch 32 braille notetaker [18] has an impressive touchscreen keyboard but suffers from poor panning and cannot display braille images, because of its 1D braille display. However, since the device consists of a tablet touchscreen and a braille pad, it is possible to share information directly with the non-disabled. While DAISY is supported by default, EPUB can be downloaded and used from the Google Play App or through a web browser such as Chrome in Android OS.

Byrd's research [7] developed a braille device with a 2D LED braille display, which is resistant to the panning problem. Since this device is in 2D and has a multiarray cell structure ranging from 4×40 to 12×30, display of braille figures is also possible. In addition, an instant visual feedback is indirectly possible via Bluetooth and a braille display. However, DAISY and EPUB are not currently supported by the device itself. Velázquez's [8] work focused on the braille hardware, which is a condensed braille device that considers the convenience of VI device users and has fewer functions than the previous devices. However, this too has a 1D braille display, and its small size causes panning problems. Thus, braille figures are also difficult to print. Furthermore, it is difficult to support the instant visual feedback as well as the DAISY and EPUB formats. On the contrary, the method proposed in this study not only decreases the panning problem but can also to display braille images because of its 12×12 braille cell design. Additionally, as explained in Section 3, real-time communication through Bluetooth on smartphones and braille displays enables information sharing and visual feedback between the VI and non-disabled users. Above all, this study included all versions of DAISY and EPUB. However, it is still very difficult to allow the VI people to sense images and videos through the braille pad.

Table 6. Comparison of introduced braille displays in this study.

Device	Panning Problem	Braille Image Support	Instant Visual Feedback	DAISY and EPUB
Blitab [16]	Strong	○	Direct	Unknown
BrailleSense Polaris [17]	Weak	×	Indirect	Support
BrailleSense U2 [17]	Weak	×	Indirect	Support
BrailleNote Touch 32 Braille Notetaker [18]	Weak	×	Direct	Support
Byrd's [7]	Strong	×	Indirect	Support
Velázquez's [8]	Weak	×	×	×
Ours	Strong	○	Indirect	Support

In this study, video was expressed only as the cover image of the video, and, as shown in Figure 16, although images were displayed on the braille device simulator, it is not enough to express the details on 11×11. Therefore, further research is required to display photographs and pictures that contain several objects on the braille device using the limited number of braille characters. Furthermore, although this study considered a method for expressing videos on the pad, further research is required to improve features such as the reaction speed of actuators, power consumption, and design, with respect to the braille hardware; moreover, there are many challenges that need be addressed in terms of software as well, such as video sampling, filtering, streaming, and compression. Hence, this study intends to use a method for displaying a typical cover image for videos, which is also difficult because a cover image requires accurate filtering and conversion in addition to the resolution problem of the braille display. Therefore, more studies are required for adequate expression of images or videos on a braille display. Assuming that advancements will be made in hardware through the use of superior technology, the final objective of this study is to output videos on a multiarray braille device.

In addition, with respect to the uses of 2D braille display that have not been fully developed yet, practical feedback from VI users is required through the use of prototype devices. The devices must be tested to determine their ease of use compared to other braille devices and whether the existing disadvantages have been overcome. Furthermore, although this study was conducted for the VI, there have been several studies targeting ordinary persons. Therefore, we wish to emphasize that, in the future, more studies should be carried out on people with disabilities. As described in Section 1, with the growing global population, the number of physically challenged persons is also rapidly growing. Hence, we believe that studies for people with various disabilities will be very valuable and is a desperate requirement of disabled people. We therefore need to strive to make a better society

where disabled and non-disabled people can live harmoniously without much differences in their daily lives.

5. Conclusions

This study proposes a 2D multiarray braille display device for the VI. We developed a method for delivering braille translation and multimedia information for 2D multiarray braille display to overcome the limitations of the 1D conventional braille devices. Besides, since texts are expressed in several lines, the VI users will be able to read eBooks efficiently and quickly. This research also enhances the image expression on the braille display by extracting edges and translating images for the braille. This method provides ways to sense visual media (e.g., videos, pictures, and images) through touch. With the multiarray braille display, diverse multimedia contents (e.g., text, figures, and audio status symbols) can be effectively converted to braille. Furthermore, it allows playing audio files on the smartphone as well as the tablet that replaces the braille device. This novel 2D multiarray braille display technology is very useful for the VI as it increases their accessibility to information and helps to navigate through multimedia contents. From the educational perspective, the 2D mobile braille display can be very useful for obtaining information such as literature, scientific figures, and audio-based education. However, in future works, further studies must be conducted on braille devices and should be tested by VI users. The most preferred research would be methods to display videos on a cutting-edge braille device. Hence, the current challenge is to develop adaptive software with a new concept braille display.

Abbreviations

The following abbreviations are used in this manuscript:

DAISY	Digital accessible information system
VI	Visually impaired
GPS	Global positioning system
RFID	Radio frequency identification
LED	Light emitting diode
NoIR	No infrared (filter)
EPUB	Electronic publication
TTS	Text-to-speech
NCC	Navigation control center
NCX	Navigation control center for extensible markup language
XML	Extensible markup language
SMIL	Synchronized multimedia integration language
HTML	Hyper text markup language
HD	High definition
SD	Standard definition

References

1. Bourne, R.R.; Flaxman, S.R.; Braithwaite, T.; Cicinelli, M.V.; Das, A.; Jonas, J.B.; Keeffe, J.; Kempen, J.H.; Leasher, J.; Limburg, H.; et al. Magnitude, temporal trends, and projections of the global prevalence of blindness and distance and near vision impairment: A systematic review and meta-analysis. *Lancet Glob. Health* **2017**, *5*, e888–e897. [CrossRef]
2. Colby, S.L.; Ortman, J.M. *Projections of the Size and Composition of the US Population: 2014 to 2060: Population Estimates and Projections*; U.S. CENSUS BUREAU: Washington, DC, USA, 2017.
3. He, W.; Larsen, L.J. *Older Americans with a Disability, 2008–2012*; US Census Bureau: Washington, DC, USA, 2014.
4. Elmannai, W.; Elleithy, K. Sensor-based assistive devices for visually-impaired people: Current status, challenges, and future directions. *Sensors* **2017**, *17*, 565. [CrossRef] [PubMed]
5. Park, J.; Sung, K.K.; Cho, J.; Choi, J. *Layout Design and Implementation for Information Output of Mobile Devices Based on Multi-Array Braille Terminal*; The 2016 Winter Korean Institute of Information Scientists And Engineers; Korea Information Science Society: Pyeongchang, Korea, 2016; pp. 66–68.
6. Bornschein, J.; Prescher, D.; Weber, G. Collaborative creation of digital tactile graphics. In Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility, Lisbon, Portugal, 26–28 October 2015; pp. 117–126.
7. Byrd, G. Tactile digital braille display. *Computer* **2016**, *49*,88–90. [CrossRef]
8. Velázquez, R.; Preza, E.; Hernández, H. Making eBooks accessible to blind Braille readers. In Proceedings of the 2008 IEEE International Workshop on Haptic Audio visual Environments and Games (HAVE 2008), Ottawa, ON, Canada, 18–19 October 2008; pp. 25–29.
9. Bae, K.J. A Study on the DAISY Service Interface for the Print-Disabled. *J. Korean Biblia Soc. Libr. Inf. Sci.* **2011**, *22*, 173–188.
10. Won, J.; Lee, H.; Kim, T.-E.; .Lee, J. *An Implementation of an Android Mobile E-book Player for Disabled People*; Korea Multimedia Society: Seoul, Korea, 2010; pp. 361–364.
11. Kim, T.E.; Lee, J.; Lim, S.B. A design and implementation of DAISY3 compliant mobile E-book viewer. *J. Digit. Contents Soc.* **2011**, *12*, 291–298. [CrossRef]
12. Goncu, C.; Marriott, K. Creating ebooks with accessible graphics content. In Proceedings of the 2015 ACM Symposium on Document Engineering, Lausanne, Switzerland, 8–11 September 2015; pp. 89–92.
13. Harty, J.; Holdt, H.C.; Coppola, A.; LogiGearTeam. Android Daisy Reader. 2013. Available online: https://code.google.com/archive/p/android-daisy-epub-reader (accessed on 6 October 2018).
14. Mahule, A. Daisy3-Reader. 2011. Available online: https://github.com/amahule/Daisy3-Reader (accessed on 6 October 2018).
15. Canadian Assistive Technologies Ltd. What to Know Before You Buy a Braille Display. 2018. Available online: https://canasstech.com/blogs/news/what-to-know-before-you-buy-a-braille-display (accessed on 10 January 2019).
16. Blitab Homepage. Blitab Technology. 2017. Available online: https://blitab.com/ (accessed on 25 February 2019).
17. HIMS International. BrailleSense Polaris and U2. 2018. Available online: http://himsintl.com/blindness/ (accessed on 18 October 2018).
18. Humanware Store. BrailleNote Touch 32 Braille Notetaker. 2017. Available online: https://store.humanware.com/asia/braillenote-touch-32.html (accessed on 10 January 2019).
19. Park, T.; Jung, J.; Cho, J. A method for automatically translating print books into electronic Braille books. *Sci. China Inf. Sci.* **2016**, *59*, 072101. [CrossRef]
20. Jung, J.; Kim, H.G.; Cho, J. Design and implementation of a real-time education assistive technology system based on haptic display to improve education environment of total blindness people. *J. Korea Contents Assoc.* **2011**, *11*, 94–102. [CrossRef]
21. Jung, J.; Hongchan, Y.; Hyelim, L.; Jinsoo, C. Graphic haptic electronic board-based education assistive technology system for blind people. In Proceedings of the 2015 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 9–12 January 2015; pp. 364–365.
22. Jeong, I.; Ahn, E.; Seo, Y.; Lee, S.; Jung, J.; Cho, J. *Design of Electronic Braille Learning Tool System for Low Vision People and Blind People*; The 2018 Summer Conference of the Korean Institute of Information Scientists and Engineers; Korea Institute of Communication Sciences: Jeju, Korea, 2018; pp. 1502–1503.

23. Seo, Y.S.; Joo, H.J.; Jung, J.I.; Cho, J.S. Implementation of improved functional router using embedded Linux system. In Proceedings of the 2016 IEIE Summer Conference, Jeju, Korea, 22–24 June 2016; The Institute of Electronics and Information Engineers: Jeju, Korea, 2016; pp. 831–832.

24. Leithinger, D.; Follmer, S.; Olwal, A.; Ishii, H. Shape displays: Spatial interaction with dynamic physical form. *IEEE Comput. Graph. Appl.* **2015**, *35*, 5–11. [CrossRef] [PubMed]

25. Follmer, S.; Leithinger, D.; Olwal, A.; Hogge, A.; Ishii, H. inFORM: Dynamic physical affordances and constraints through shape and object actuation. In Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology, St. Andrews, Scotland, UK, 8–11 October 2013; Volume 13, pp. 417–426.

26. Massachusetts Institute of Technology, School of Architecture + Planning. Tangible Media Group(MIT)—The inFORM System. 2018. Available online: http://tangible.media.mit.edu/project/inform/ (accessed on 31 October 2018).

27. Dunai Dunai, L.; Chillarón Pérez, M.; Peris-Fajarnés, G.; Lengua Lengua, I. Euro banknote recognition system for blind people. *Sensors* **2017**, *17*, 184. [CrossRef] [PubMed]

28. Consortium, T.D. Daisy Consortium Homepage. 2018. Available online: http://www.daisy.org/home (accessed on 6 October 2018).

29. IDPF (International Digital Publishing Forum). EPUB Official Homepage. 2018. Available online: http://idpf.org/epub (accessed on 6 October 2018).

30. Google Developers. Documentation for Android Developers—Message. 2018. Available online: https://developer.android.com/reference/android/os/Message (accessed on 7 October 2018).

31. Google Developers. Documentation for Android Developers—Activity. 2018. Available online: https://developer.android.com/reference/android/app/Activity (accessed on 7 October 2018).

32. Bray, T.; Paoli, J.; Sperberg-McQueen, C.M.; Maler, E.; Yergeau, F. *Extensible Markup Language (XML) 1.1*, 2nd ed.; W3C Recommendation: London, UK, 2006.

33. Daisy Consortium. DAISY 2.02 Specification. 2018. Available online: http://www.daisy.org/z3986/specifications/daisy_202.html (accessed on 7 October 2018).

34. Daisy Consortium. Part I: Introduction to Structured Markup—Daisy 3 Structure Guidelines. 2018. Available online: http://www.daisy.org/z3986/structure/SG-DAISY3/part1.html (accessed on 7 October 2018).

35. Park, E.S.; Kim, S.D.; Ryu, Y.; Roh, H.J.; Koo, J.; Ryu, E.S. *Design and Implementation of Daisy 3 Viewer for 2D Braille Device*; The 2018 Winter Conference of the Korean Institute of Communications and Information Sciences; Korea Institute of Communication Sciences: Jeongseon, Korea, 2018; pp. 826–827.

36. Daisy Consortium. Daisy Sample Books. 2018. Available online: http://www.daisy.org/sample-content (accessed on 10 October 2018).

37. Kim, S.D.; Roh, H.J.; Ryu, Y.; Ryu, E.S. *Daisy/EPUB-Based Braille Conversion Software Development for 2D Braille Information Terminal*; The 2017 Korea Computer Congress of the Korean Institute of Information Scientists and Engineers; Korea Information Science Society: Jeju, Korea, 2017; pp. 1975–1977.

38. Paul Siegmann. EPUBLIB—A Java EPUB Library. 2018. Available online: http://www.siegmann.nl/epublib (accessed on 18 October 2018).

39. Siegmann, P. Epublib for Android OS. 2018. Available online: http://www.siegmann.nl/epublib/android (accessed on 6 October 2018).

40. Hedley, J. Jsoup: Java HTML Parser. 2018. Available online: https://jsoup.org/ (accessed on 7 October 2018).

41. Google Developers. Save Key-Value Data. 2018. Available online: https://developer.android.com/training/data-storage/shared-preferences (accessed on 10 October 2018).

42. Shi, J.; Tomasi, C. *Good Features to Track*; Technical Report; Cornell University: Ithaca, NY, USA, 1993.

43. PRESSBOOKS. EPUB Sample Books. 2018. Available online: https://pressbooks.com/sample-books/ (accessed on 8 October 2018).