# Master's Thesis

Title

# Improving 3D Printer Accessibility for People with Visual Impairments by Reading Menus and Scrolling Text

Primary Examiner: Prof. Dr. Naoki Mori

Examiners: Prof. Dr. Koichi Kise

Associate Prof. Dr. Katsufumi Inoue

Associate Prof. Dr. Masakazu Iwamura

(No. BGA23098) Naoya Tagawa

Department of Core Informatics

Graduate School of Informatics

Osaka Metropolitan University

# Acknowledgements

I would like to express my deep gratitude to Associate Professor Masakazu Iwamura for his direct guidance in this research. He not only gave me advice on research but also gave me a lot of advice that served as a guide for life.

I would like to thank Professor Kazunori Minamitani for his advice, especially on research guidelines.

I would like to thank Lecturer Yuzuko Utsumi, Associate Professor Motoi Iwata, and Professor Koichi Kise for their advice during the research presentation.

I would like to thank Ms. Onishi, Yoh Otsuka, Banri Kakehi, Tokio Uchida, Yuto Nakao, Yuichiro Iwasita, the Intelligent Media Processing. Research Group office staff, everyone in the Supporting for People with Visual Impairments Group, and other lab members for their support in the lab and everyday life.

Last but not least, I would like to express my deepest gratitude to my family.

# Contents

# List of Figures

## 1 INTRODUCTION

*"Objects such as snowflakes, castles, and butterflies have become more than just words when explored as a 3D print."* This impressive sentence appears in the very beginning of [17], which reports the activities of See3D [28], a non-profit organization that manages the printing and distribution of 3D-printed models for people who are blind. Similar activities include [2, 3]. 3D printing has immense potential to enhance the lives of people with visual impairments (PVI). 3D-printed tactile objects are invaluable sources of information for PVI, enabling them to comprehend shapes and other details through touch that words alone cannot convey [2, 28]. These objects can also serve as effective educational tools, which greatly benefit students with visual impairments [3, 21].

However, there is little literature (e.g., Minatani [21]) and few Web sites (e.g., Round Table [1]) on the use of 3D printers by PVI to print 3D models. There are several benefits for PVI using 3D printers [21]. The most significant is the ability to independently produce desired objects. 3D printing enables PVI to create the objects they want on their own without depending on sighted people to interpret their descriptions, which can often be challenging and inaccurate. Another key benefit is the potential for personalized assistive devices. 3D printers allow individuals with visual and other impairments to design and create customized assistive technologies tailored to their specific needs. This is crucial as traditional assistive devices often do not meet individual requirements due to the relatively small disabled population.

Therefore, we aim to make 3D printers accessible for PVI. Methods that allow PVI to read text on home appliance displays have been extensively researched [6, 12–14, 18, 24, 25, 27, 30]. The major concerns of previous research include how to find the location of the text, how to take a good-quality photograph of the text, and how to read the text well. However, our work addresses completely different issues: scrolling text and menus. We implemented reading functions for them, which, to the best of our knowledge, is a novel contribution. Our method for scrolling text is simple yet effective, making it easily applicable to other OCR engines in smartphone apps and wearable devices. We implemented a prototype system for accessing a 3D printer console for its potential needs of PVI (e.g., [22]). A user study involving 13 PVI (five blind and eight with low vision) confirmed the effectiveness of the functions implemented over conventional smartphone apps and wearable devices.

The contributions of this paper are summarized below.

(1) To the best of our knowledge, we are the first to implement a function to read scrolling text. Our method is simple yet effective, making it easily applicable to other OCR engines in smartphone apps and wearable devices.
(2) To the best of our knowledge, we are the first to implement a function that reads the selected menu item.
(3) Through a user study with 13 PVI, we demonstrate that the proposed functions significantly enhance the ability of PVI to operate the 3D printer console.

## 2 RELATED WORK

### 2.1 Screen readers

Methods of reading text on home appliance displays for PVIs have been extensively researched. The major concerns of previous research include how to find the location of the text, how to take a good-quality photograph of the text, and how to read the text well. A pioneering work by Morris *et al.* [25] has proposed the attachable screen reader using a web camera for liquid crystal display (LCD) or light-emitting diode (LED) screens in home appliances. Text detection and recognition are performed using markers and template matching, respectively. Their attachable screen reader provides three modes of operation that read all the information, significant portions, and changes that appear on the screen. They pointed out the importance of scrolling text (referred to as *scrolling display* in their paper), which they considered in future work. It is curious that the word *screen reader* is used only in this paper and does not appear in subsequent papers. Fusco *et al.* [13, 14] have proposed a smartphone app called *Display Reader* that uses markers to find display text and helps PVIs to ask remote sighted assistants to read the text using Be My Eyes [4]. Finnegan *et al.* [12] have proposed a method to read seven-segment digits on the displays of blood glucose meters and blood pressure monitors using image processing and computer vision techniques. Tekin *et al.* [30], Rasines *et al.* [27], and Kasar [18] also propose similar methods using image processing and computer vision techniques. Moreira *et al.* [24] and Boonnag *et al.* [6] have proposed methods to read the display text of medical devices using object detection techniques. In addition, Lee *et al.* [19] have pointed out the necessity of recognizing display text in the context of remote-sighted assistance.

Guo *et al.* [15] have proposed an interactive tool, called *VisLens*, that helps PVIs to access flat interfaces, such as microwave control pads. In VizLens v2, they add a function to read text on the displays. After preprocessing the text image, they apply Tesseract OCR [31], an open-source OCR engine, to read the text. Although they call the proposed interactive tool an *interactive screen reader*.

As we see above, to the best of our knowledge, we are the first to make it possible to read the scrolling text and menu items using attachable screen readers, while the importance of the scrolling text is pointed out in [25].

### 2.2 3D printing for people with visual impairments

Researchers have repeatedly emphasized the importance of involving people with disabilities who need assistive technology in the development of their assistive tools (e.g., Hurst *et al.* [16], Bennett [5] in the cases of PVIs). However, current commercial and open-source computer-aided design (CAD) tools that support viewing, authoring, and editing of 3D models are mostly visually reliant and limit access for PVIs [29]. To overcome these limitations, Siu *et al.* [29] has implemented an accessible 3D modeling workflow, called *shapeCAD*, which allows users with visual impairments to complement programming of 3D models while obtaining haptic feedback using a 2.5D tactile shape display. Lieb *et al.* [20] have presented a system that gives blind modelers an audio-haptic preview of their 3D object using an inexpensive haptic system with a single end effector. Hurst *et al.* [16] have presented two software prototypes that enable novices to quickly and easily create 3D printable physical objects: *VizTouch* automatically generates 3D printable tactile graphics [7], and *Easy Make Oven* is an interactive surface for scanning, editing, and creating physical objects. Minatani, a blind researcher, demonstrated the potential for accessible 3D modeling by designing a 3D CAD model of the leaning tower of Pisa using a programmable CAD processing system [23].

In contrast to the attention given to the development of assistive tools, there is little literature (e.g., Minatani [21]) and few websites (e.g., Round Table [1]) on the use of 3D printers by PVIs to print 3D models. There are several benefits for PVIs using 3D printers [21]. The most significant is the ability to independently produce desired objects. 3D printing enables PVIs to create the objects they want on their own without depending on sighted people to interpret their descriptions, which can often be challenging and inaccurate. Another key benefit is the potential for personalized assistive devices. 3D printers allow individuals with visual and other impairments to design and create customized assistive technologies tailored to their specific needs. This is crucial as traditional assistive devices often do not meet individual requirements due to the relatively small disabled population.

Therefore, Minatani has reported a practical method for PVIs to independently produce a 3D CAD model using a 3D printer [22]. However, the process reported in this paper is a fortunate success case, where the printing process was completed without any issues. Typically, when you encounter a problem or need maintenance—such as when the filament runs out or when the printer needs calibration—you must operate the 3D printer's console by looking at the display. Currently, the only practical way for PVIs to manage this is by using smartphone apps and wearable devices to read the text displayed on the display, which is extremely cumbersome and time-consuming. There is a website that introduces practical information for PVIs to use 3D printers [1]. The suggested method on this site is remembering all menu items (or listing them beside the printer) and counting to select the desired item using the physical click knob. However, the menu structure is too complex to remember. In addition, some settings require turning a knob to determine a value, which cannot be handled by simply memorizing the menu hierarchy. To alleviate the above situations, the proposed system aims to facilitate easier operations of the 3D printer console for PVIs, which can be applied to other home appliances.

### 2.3 Central Visual Field Loss And Scrolling Text

Regarding the effectiveness of character display methods for PVI, E. N. Fine *et al.* [11] have suggested that dynamically displayed text is faster and easier to read than static text for people with central visual field loss (CVL). A representative example of a dynamic display method is RSVP (rapid serial visual presentation), proposed by Mary C. Potter [26] is a method to reduce eye movement and improve readability by displaying characters continuously at the same position on the screen at high speed. The horizontally scrolling text has also been attracting attention as a method of dynamic text display. Farah Akthar *et al.* [10] have compared the reading performance of subjects with CVL by presenting the same text in three ways: static, horizontal scrolling, and RSVP. The horizontal scrolling presentation

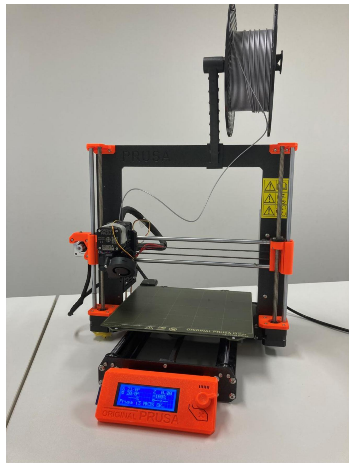


Fig. 1. Overall picture of the Original Prusa i3 MK3.

resulted in the best, with an average comprehension of 79.46% and a misreading rate of 1.35%, while the RSVP presentation was the worst, with only a comprehension of 46.22% and a misreading rate of 11.39%. The static text presentation showed a comprehension of 52.97% and a misreading rate of 2.13%. In addition, Robin Walker [32]has developed a tablet reading aid app called *MD_evReader* for users with CVL. The app aims to improve reading accuracy by displaying text that scrolls one line at a time. When comparing static and dynamic (scrolling) presentations on a tablet, they found that although there was no significant difference in reading speed or comprehension, the reading error rate was significantly lower for the dynamic presentation. These research suggests that a horizontal scrolling display method is useful for people with CVL. However, no approach currently reads scrolling text aloud to PVI. Therefore, one of the proposed system's functions, *a function to read scrolling text*, is believed to be a new means of effectively conveying information on a display to PVI, including those with CVL.

## 3 PROPOSED SYSTEM

### 3.1 Overview

The proposed system is tailored for the Original Prusa i3 MK3, a consumer-grade 3D printer, as shown in Fig. 1. The 3D printer has a console panel, as shown in Fig. 2, which consists of an LCD on the left and a control knob with a reset button directly under the

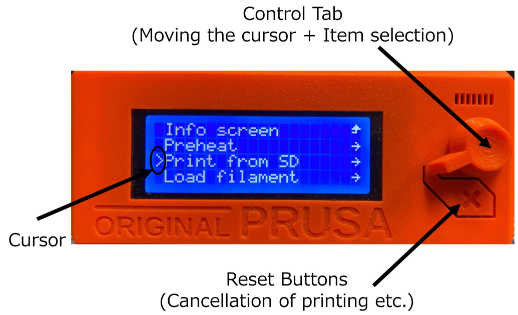



Fig. 2. The console of a 3D printer, which consists of a liquid crystal display (LCD) on the left and a control knob with a reset button directly under the control knob on the right.

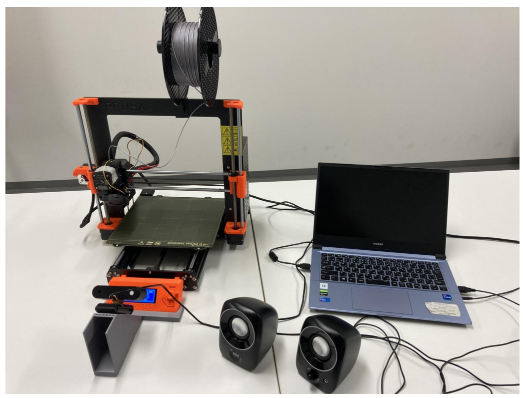
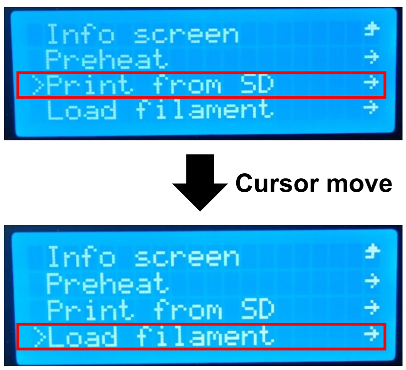


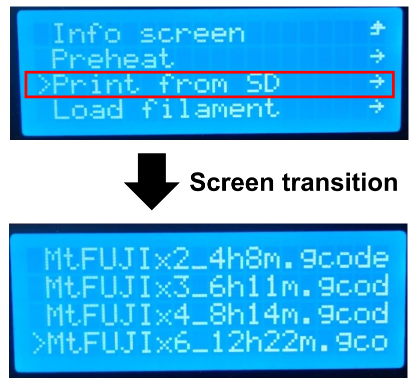
Fig. 3. Overview of the proposed system, consisting of a camera mounted on the stand, a speaker (which is not contained in the photo), and a laptop computer.

control knob on the right. The knob can be turned and pushed. When turning the right knob clockwise or counterclockwise, a clicking sound is played, and the cursor moves one step downward or upward. Pushing the knob makes a selection. The LCD displays white text on a blue background, showing four lines vertically and 20 characters horizontally. The LCD is surrounded by a black frame, which is encased in an orange plastic housing. Fig. 3 shows an overview of the proposed system attached to the 3D printer. The system consists of a web camera, a speaker, and a laptop computer. The web camera is mounted on the plastic stand printed by the 3D printer, as shown in Fig. 4. The stand, printed using the 3D printer itself, ensures that the camera is positioned to capture the display head-on when fitted into the console.

Fig. 4. Stand for fixing the camera




(a) Menu item pointed to by cursor




(b) Screen transition

Fig. 5. Information on the menus: (a) When the cursor is moved, the menu item pointed to by the cursor should be read out immediately. (b) When the screen transition occurs, all the information displayed on the new screen should be read out immediately.

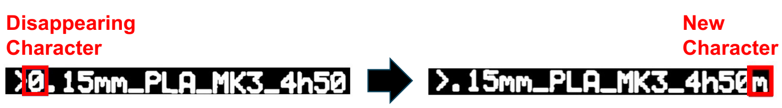



Fig. 6. Text horizontally scrolls if the length of the text exceeds 19 characters. In the scroll, the leftmost character disappears, and a new character appears on the right.

### 3.2 Reading functions

We will discuss how to implement reading functions for the menu and scrolling text below.

*3.2.1 Reading function for menu.* Reading menus presents three main challenges:

(1) Cursor position detection

As shown in Fig. 5(a), the text that should be read out is adjacent to the cursor.

Therefore, the first step is to detect the cursor and then identify the text adjacent to it. To avoid irritating users, the system reads out the text only when the cursor moves. Therefore, when the movement of the cursor is detected, the system determines its position and reads out the item pointed to by the cursor, allowing users to understand the selected item. This process is realized by character recognition (see Section 3.3.3) and its simple post-process to find the adjacent text. When reading out, the system speaks "The cursor points to" before the item pointed to by the cursor.

(2) Vertical scroll detection
If the number of menu items exceeds the number of vertical lines on the display, some menu items will extend beyond the visible screen area. When the cursor is at the top of the screen, and the user attempts to move it further up, the screen will scroll down to reveal the new item unless the cursor is already pointing to the first item in the menu list. The vertical scroll is similar to the behavior of the horizontal scrolling text (see Fig. 6), except that it does not automatically scroll like the horizontal scrolling text. In this case, the cursor position remains the same before and after scrolling, but the text lines shift downward one by one. In other words, even if the cursor does not appear to be moving, it is possible to detect the movement of the cursor and read out the item pointed to by the cursor by detecting whether scrolling has occurred. Therefore, vertical scrolling must be detected. The same principle applies when the screen scrolls upward. This process is realized by change detection (see Section 3.3.2).

(3) Screen transition detection
Transitioning to a new level in the menu hierarchy should be detected. As shown in Fig. 5(b), pressing the knob causes a transition to a new screen displaying more detailed information about the selected item. Detecting this transition is crucial for providing users with the appropriate information about their current location within the menu structure and guiding them through the navigation process. This process is also realized by change detection (see Section 3.3.2).

*3.2.2 Reading function for horizontal scrolling text.* Reading scrolling text poses a unique challenge since the entire text cannot be seen at once, as shown in Fig. 6. Therefore, we

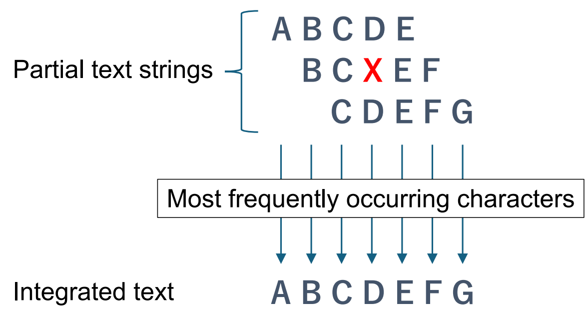



Fig. 7. Basic concept of text merging scheme for horizontal scrolling text. After aligning the partial text strings, the most frequently occurring character is selected at each position to construct the integrated text.

need to recognize the visible portion of the text in each frame (say, partial text string) and concatenate them to form a complete message. We refer to this concatenated text across multiple frames as the *integrated text.* Although text recognition techniques are generally accurate, they are inherently prone to errors. Therefore, our method must be robust against recognition errors while avoiding the consumption of large computational resources. To address these requirements, we propose a simple and efficient method. Fig. 7 shows the basic concept of our method. Our method first aligns the partial text strings and then selects the character that occurs most frequently at each position, which forms the integrated text. However, unlike the situation in the figure, the partial text string sequentially arrives one by one. Therefore, our method sequentially concatenates partial text strings by alternating between two key processes: text string alignment and text concatenation. In the text string alignment process, the method finds the most appropriate position to align the partial text string from the current frame with the integrated text. This ensures that the new partial text string is correctly positioned in relation to the already concatenated text. Once the text is aligned, the text concatenation process merges the new text string with the integrated text. To ensure robustness against recognition errors, we employ a voting mechanism during the merging process. That is, we prepare a voting table for each position and take the character that occurs most frequently as the representative of the position. This voting system helps minimize the impact of individual recognition errors on the final integrated text. By iteratively applying these two processes, our method effectively reads scrolling text while maintaining efficiency and robustness.

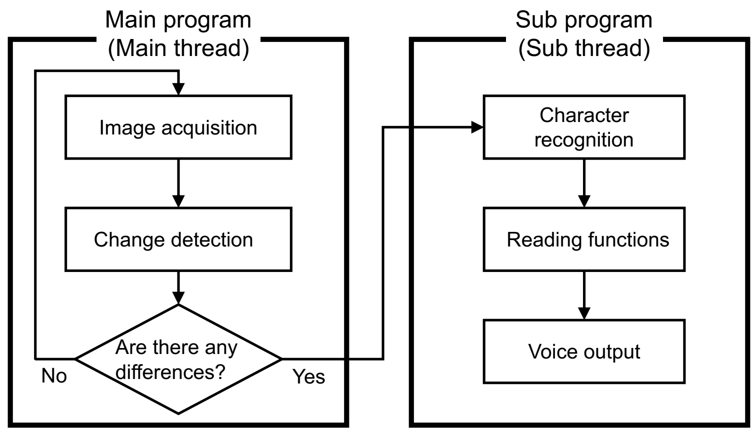



Fig. 8. Flow diagram of the proposed system

### 3.3 Implementation details

For real-time processing, the program is divided into a main program and a sub-program, which run in two threads. As Fig. 8 shows the flow diagram of the proposed system, the main program obtains images (see Section 3.3.1) and detects differences (see Section 3.3.2), while the sub-program performs character recognition (see Section 3.3.3), executes reading functions (see Section 3.2), and voice output (see Section 3.3.4). The proposed system is implemented in Python.

*3.3.1 Obtaining binary text image.* The target 3D printer exhibits fine fluctuations in luminance on the display within a certain period of time. As a result, taking differences between frames leads to detecting changes due to the luminance values, even when the display has not changed. To stabilize the luminance values, the average image of the last ten display snapshots is calculated, resulting in the image shown in Fig. 9(a). To perform character recognition, the display area, consisting of white text and a blue background, has to be extracted from the image. The proposed system identifies the blue display area by a simple thresholding technique and replaces the remaining areas with black pixels, as shown in Fig. 9(b). Subsequently, to emphasize the characters, binarization is applied to the image. Fig. 9(c) presents the binarized version of the image in Fig. 9(b).

*3.3.2 Change detection on the screen.* Screen changes are detected in two stages. The first stage detects changes in the entire screen by comparing the average image of the


(a) The average image of the last ten display snapshots. (b) Display area with white text and a blue background. (c) Binarized text regions.

Fig. 9. Obtaining binary text regions from display snapshots.

previous ten frames with the newly captured image. If the difference exceeds a threshold, the screen is considered to have changed. If a screen change is detected in the first stage, the second stage detects changes in each row to identify vertical rolling. This is done by comparing each row of the average image with each row of the newly captured image in the brute-force manner. If the difference is below a threshold, a matching row is considered to be found. This two-stage process allows the system to determine the number of rows the screen has scrolled vertically or whether the entire screen has changed simultaneously.

*3.3.3 Character recognition.* Before performing character recognition, character segmentation is performed first. When the main program detects a difference, it crops the image of the row where the difference occurred and passes the cropped image to the sub-program. The sub-program then calculates the horizontal and vertical projection profiles of the cropped image to detect characters. By analyzing the peaks and valleys in the projection profiles, the boundaries of individual characters can be identified within the cropped image. This character segmentation step is crucial for isolating and extracting characters accurately, enabling subsequent character recognition processes.

Character recognition is performed using logistic regression, where the pixel intensity values of the images are used as explanatory variables, and the characters written in the images serve as the target variable. The characters displayed on the 3D printer's screen consist of 84 different types, including digits, uppercase and lowercase alphabets, special indicators, and symbols, including the cursor “>.” To train the model, a few dozen images of each character were collected from the screen and augmented to increase the dataset, resulting in a total of 84,357 images. The dataset was split into training sets (70%) and test sets (30%). The trained model is then used to perform logistic regression, outputting the class labels of recognized characters. The trained model performed 99.96% recognition accuracy on the test sets.

*3.3.4 Voice output.* Voice output is generated by creating an audio file based on the text generated in reading functions (see Section 3.2) and then playing it. When a new difference image is received from the main thread, the system interrupts the currently playing audio and starts playing the newly generated audio file. This ensures a quick response to user operations. The *pyttsx3* library is used to create the audio files, while the *pyaudio* library is employed to play them.

## 4 USER STUDY

To evaluate the usability, benefits, and drawbacks of our prototype system, we conducted a user study with 13 PVIs. We started with a pre-study interview to collect participants' demographic information and understand the participants' experience with smartphone apps and wearable devices that help to obtain visual information. Then, we conducted two sessions to evaluate two reading functions of our prototype system, comparing with the participants' preferred method from among their eyes for those with low vision and the smartphone apps and wearable devices they selected. Finally, we conducted a post-study interview to supplement our quantitative results and to gain further insights into user preferences and needs.

### 4.1 Method

*4.1.1 Preparation.* The 3D printer used in our study supports English, Czech, German, Spanish, Italian, and Polish languages. However, all participants are native Japanese speakers and may not be familiar with the aforementioned languages (no participants indicated confidence in these languages during our pre-study interview). To accommodate this, we set the language of the 3D printer to English and prepared both English and Japanese speakers for speech synthesis in the proposed method. In addition, prior to the experiment, we asked the participants to select their preferred speaker and adjust the speech speed to their preferred setting.

To compare with our prototype system, we asked the participants to select their preferred methods as their conventional methods. They mostly select smartphone apps they use in their daily lives. However, some participants did not use such apps. Therefore, in such a case, we let them use our smartphones where Seeing AI and Envision AI are installed. We allowed participants who had the sight to use their own eyes if they selected it as their comfortable way.

*4.1.2 Session 1: Menu selection tasks.* We prepared three different operation tasks, as shown in Table 1, which require 1-, 2-, and 3-step operations, respectively. The tasks started with the first item on the first menu screen, i.e., the "Info screen," and the operation task was completed when the user selected the designated item. Fig. 10 exemplifies an operation task. To set the language as shown, participants must select "Settings," "Language," and "English" in this order. Before each task, we informed participants of the items to select and the order to follow.

We asked participants to perform the tasks using the proposed method and the conventional method. As a combination of three tasks and two methods, there were six experiments in total. The order of these experiments was randomly shuffled for each participant. The time required to complete the operation was measured. If participants felt that they could not complete the task even with more time, or they felt they were so tired, we allowed them to quit the task prematurely to reduce the participant's workload. In such a case, we regarded the time taken to complete the task as the upper limit (i.e., 30 minutes).

The 3D printer console will time out and return to the default screen if it is not operated for a while. Therefore, the experimenter performed appropriate operations to avoid timeouts. However, this was not necessary for the proposed method. For tasks spanning multiple screens, we immediately pointed out any wrongly selected item, asked the participant to return to the previous screen, and resumed the experiment. When using the conventional method, the printer name engraved below the LCD could be read by the smartphone app. If the participant requested, the experimenter covered this text with a piece of paper.

After the completion of the experiment, a questionnaire about the experiment was taken.

Table 1. Order of operations for tasks. Participants were asked to select the designated menu items in the specified order.

| Task | Order of operations |
|---|---|
| Task 1 | *Load filament* |
| Task 2 | *Calibration* ⟶ *Wizard* |
| Task 3 | *Settings* ⟶ *Select language* ⟶ *English* |


Fig. 10. Designated operations of Task 3 in Session 1.

*4.1.3 Session 2: Reading horizontal scrolling text.* In this session, participants are asked to hear the file names on an SD card that gradually scrolls horizontally over time using the proposed system and the conventional method separately. We asked a question, "Do you think a function that concatenates horizontally scrolling text into a single sentence would help you understand the file name that scrolls horizontally?" to answer on a five-point Likert scale.

Initially, we had planned to ask participants to estimate the file names using two methods respectively and compare the required times and accuracies. However, due to the extreme difficulty of using the conventional method for this task, we decided to abandon this plan.

### 4.2 Participants

We conducted a user study with 13 PVIs: seven males and six females, whose ages ranged from 34 to 77, with an average age of 51.8 years. Five participants had complete blindness, while eight had low vision. Table 2 summarizes the demographic information of the participants.

Table 2. Participants' demographic information. BL and LV denote blindness and low vision, respectively. A number in the visual impairment column represents visual acuity. The *speaker* and *speed* columns indicate whether an English or Japanese speaker reads the text and how many words are spoken per minute.

| ID | Age | Sex | | Visual impairment | Speaker | Speed |
|---|---|---|---|---|---|---|
| VI-1 | 62 | F | LV | Left: 0.01, Right: 0.01 | English | 150 |
| VI-2 | 53 | F | LV | 1.0 in both eyes, but central vision is almost zero | Japanese | 150 |
| VI-3 | 54 | F | BL | | Japanese | 150 |
| VI-4 | 60 | M | LV | Left: 0.02, Right: 0.02 | English | 150 |
| VI-5 | 44 | M | BL | | Japanese | 150 |
| VI-6 | 38 | M | LV | Right: 0.01 | English | 150 |
| VI-7 | 59 | F | LV | Visual acuity when walking, visual acuity at about 3 meters is 0.5 | Japanese | 250 |
| VI-8 | 60 | F | BL | | English | 150 |
| VI-9 | 77 | M | LV | Left: 0.02 | Japanese | 150 |
| VI-10 | 34 | M | BL | | Japanese | 230 |
| VI-11 | 34 | F | LV | Left: 0, Right: 0.02 | English | 150 |
| VI-12 | 43 | M | LV | Left: 0, Right: 0.04 | Japanese | 130 |
| VI-13 | 55 | M | BL | | Japanese | 150 |

### 4.3 Result

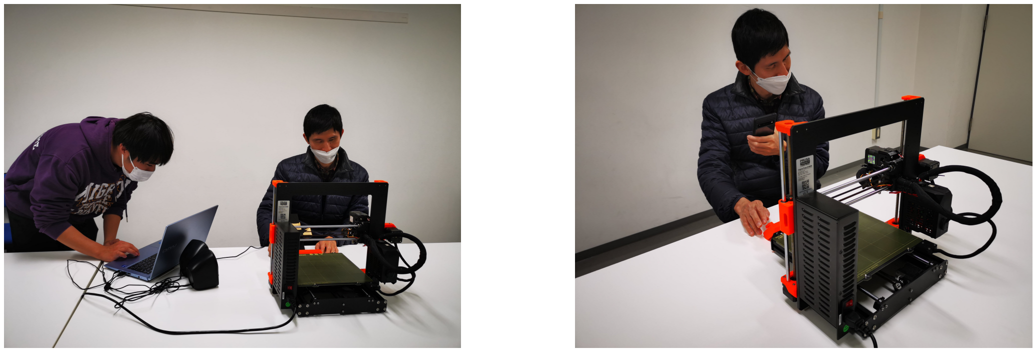
Table 2 shows the participants' preferred speaker and speech speed settings on the right. Table 3 shows participants' preferred methods (i.e., conventional methods) and how they used these methods. Snapshots during the experiments are shown in Fig. 11.

*4.3.1 Session 1: Menu selection tasks.* We compare the time taken to complete the tasks of selecting specific menu items, as described in Section 4.1.2. Fig. 12 shows the time distributions in both methods for each task. The figure indicates that the proposed method completed all tasks in a shorter time than the conventional method. This is because the proposed method always reads out the item pointed to by the cursor without requiring adjustment of the camera angle. On the other hand, the conventional method required the participants to adjust the camera angle by themselves. Furthermore, the smartphone apps participants used had difficulty in recognizing the cursor, hence requiring multiple attempts to recognize it before the app read out the cursor. Some participants judged

Table 3. Conventional methods participants attempted in the user study and how they used these methods.

| ID | Methods or apps | How to use |
|---|---|---|
| VI-1 | Sullivan+ (Text Recognition) | The app repeats the recognition until it recognizes the cursor “>.” When it reads out the cursor, she knows which item the cursor is currently pointing to. |
| VI-2 | iPhone camera transcription function | Repeat the process below to memorize the order of the items and guess the location of the cursor from what is read out.<br>1. Turn the knob to move the cursor.<br>2. Character recognition by the app.<br>3. Copy and paste to the memo app. |
| VI-3 | Sullivan+ (AI mode) | Same as VI-1. |
| VI-4 | Seeing AI (Short Text Channel) | Same as VI-1. |
| VI-5 | Be My AI | Same as VI-1. However, If the app did not recognize it, he asked the AI, “What is the cursor pointing to?” |
| VI-6 | Seeing AI (Short Text Channel) | Same as VI-1. |
| VI-7 | Own eyes | She could read the text on the display, so she operated it while looking at the display. |
| VI-8 | Be My AI | Same as VI-5. |
| VI-9 | Seeing AI (Document Channel) | Same as VI-1. |
| VI-10 | Envision AI (Scan Text) | Same as VI-1. |
| VI-11 | Envision AI (Scan Text) | Same as VI-1. At the same time, she was trying to remember the order of the items being read out. |
| VI-12 | Own eyes | When he brought his face closer to the display, he could see the text information. |
| VI-13 | Envision AI (Instant Text) | Same as VI-1. |

that the smartphone app did not read out the cursor. Therefore, they tried to memorize the order of the items by saving what the app read in a memo app. However, there were exceptional cases. VI-7 and VI-12 had low vision and could see the text on the display with their own eyes, so they operated while looking at the display. In their cases, the



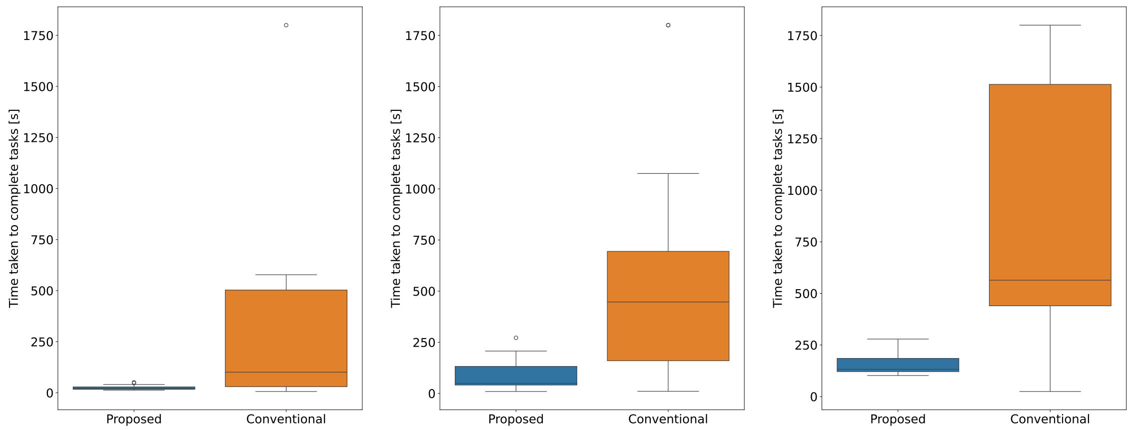
(a) Using the proposed method (b) Using a conventional method

Fig. 11. Snapshots during the experiments.

conventional method (i.e., their own eyes) required a shorter time to complete the tasks than the proposed method.

Table 4 shows the mean and standard deviation for each task for each method, with the p-value calculated by the one-sided Mann-Whitney U test. In this experiment, since the number of samples was small and normality could not be assumed, we conducted a Mann-Whitney U test to see if there was a significant difference between the two methods. As a result, the p-value was less than 0.05 for all tasks, which can be said that the time required to complete the tasks in the proposed system was significantly shorter than the conventional method. Carefully looking at the table, you may notice that the standard deviations in the conventional method were relatively large. This is because some participants gave up the tasks prematurely when using the conventional method, and we regarded their time taken to complete the tasks as 30 minutes.

After the experiment was completed, we conducted interviews to investigate the usability of the proposed system and the conventional method. First, we asked the question, "*Which method did you find easier to operate?*" Fig. 13 shows the participants' responses. Except for two people who were able to see the text on the display, all other people answered that the proposed system was easier to operate. Table 5 summarizes the advantages and disadvantages of both methods raised by the participants. As an advantage of the proposed method, many participants found that it surely reads out the item pointed to by the cursor. Other positive comments about the proposed method include the following: "*The camera is fixed, which eliminates the hassle of adjusting the angle of view.*" and "*If I*




(a) Task 1 (b) Task 2 (c) Task 3

Fig. 12. Session 1: Time taken to complete menu selection tasks.

Table 4. Session1 results: mean and standard deviation. The p-value was calculated using a one-sided Mann-Whitney U test.

| Task | Proposed | Conventional | p-value |
|---|---|---|---|
| | Mean ± SD | Mean ± SD | |
| Task 1 | **25.5** ± 12.7 | 340.0 ± 494.1 | 0.01563 |
| Task 2 | **88.8** ± 81.2 | 581.3 ± 626.5 | 0.02010 |
| Task 3 | **153.7** ± 49.5 | 905.4 ± 684.8 | 0.001458 |

*turn the knob back, it will read the part I missed again."* As negative comments about the proposed method, many said, *"It's annoying that it reads 'The cursor points to' every time it is recognized."* In addition, a negative comment about the proposed method was *"I would like the recognition accuracy to be a little higher."* In contrast, positive comments about the conventional method include the following: *"The entire screen is read out, so it's easy to understand what's currently on the screen."* and *"I was able to recognize it better than I expected."* The negative comments about the conventional method include the following: *"The accuracy of character recognition is poor," "It takes too much time and effort to adjust the camera's angle of view."* and *"I don't know what the cursor is pointing at because it doesn't read the item pointed to."* There were more negative comments than positive comments about the conventional method. Overall, the proposed method was highly evaluated, and the results showed that the conventional method was extremely difficult.

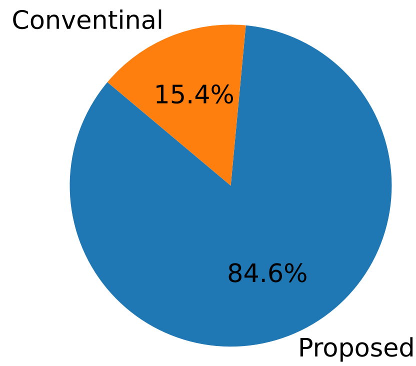



Fig. 13. Session 1 "Reading menus": This pie chart shows the answers of question, *"Which method did you find easier to operate?"* Except for two participants who were able to see the information on the display, all other people answered that the proposed system was easier to operate.

*4.3.2 Session 2: Reading horizontal scrolling text.* We present the results of an interview to see if the proposed system could be used to understand file names on an SD card that scrolls horizontally over time, as described in Section 4.1.3. Fig. 14 shows the answers to the question, *"Do you think a function that concatenates horizontally scrolling text into a single sentence would help you understand the file name that scrolls horizontally?"* Seven participants (53.8%) strongly agreed, three (23.1%) agreed, two (15.4%) answered neutral, and one (7.7%) disagreed. Positive comments include the following: *"It's very nice to be able to read information that I couldn't read before."* and *"My visual recognition can't keep up with scrolling text, so having a feature like this would be very helpful."* Negative opinions include the following: *"I don't like that there's no signal while the text is scrolling. I might think it's a malfunction."* and *"Even if this feature existed, the text would be too long. I felt that I could not fully understand it."* In particular, many participants mentioned, *"I don't like it when there is no signal while the text is scrolling."* As a way to improve this, we received advice that included: 1. playing a beep sound while scrolling and 2. reading out the visible part of the text while scrolling instead of waiting until the scroll is finished. VI-7 and VI-12, who used their own eyes in Session 1 because they could see the text on the display, told us that reading the scrolling text was very difficult for them because of luminance change and narrow field of view, respectively. It is noteworthy that explaining the concept of scrolling text to some congenitally blind participants was challenging due to their lack of prior experience.

Table 5. Positive and negative comments from the participants after Session 1 was over.

| | Proposed | Conventional |
|---|---|---|
| Positive | • Read the item the cursor points to.<br>• Simple operation.<br>• Camera is fixed.<br>• Read as soon as the screen changes.<br>• They say, "The cursor points to ~," so it's easy to see where the cursor is pointing.<br>• Rewind one and read it again. | • Maybe I can do it if I get used to it.<br>• The whole thing is read out.<br>• They gave me more textual information than I expected.<br>• Easy to reread. |
| Negative | • The "cursor point to" is annoying.<br>• Sometimes feedback takes time.<br>• Read a little more precisely. | • Takes too long.<br>• Many misidentifications.<br>• Do not read the item the cursor points to.<br>• Takes a lot of time and effort to photograph.<br>• Hold up the camera, take a picture, turn the knob, listen to the audio... It is difficult to operate as in the following.<br>• Support needed. |

## 5 DISCUSSION

### 5.1 How to improve the proposed method

Feedback from participants in the user study provided valuable insight on how to improve the proposed method. The suggestions include reducing silent durations, improving response speed, and reading only the information pointed to by the cursor when navigating menus. Interestingly, these desired improvements align closely with the behavior of screen

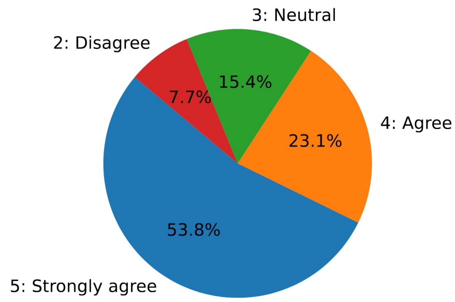



Fig. 14. Session 2 "Reading scrolling text": This pie chart shows the answers to the question, *"Do you think a function that concatenates horizontally scrolling text into a single sentence would help you understand the file name that scrolls horizontally?"* Seven participants (53.8%) strongly agreed, three (23.1%) agreed, two (15.4%) answered neutral, and one (7.7%) disagreed.

readers. It is uncertain whether this preference stems from the participants' familiarity with screen readers or because screen readers have been well-developed over time.

### 5.2 Applying the proposed method beyond 3D printers

The proposed method is tailored for a specific model of 3D printers, which may raise concerns about its applicability to other models or home appliances. Although a custom setup is required for each target device, efforts can be made to reduce adjustments. The following functions need to be adapted for broader applicability. First, the character recognition engine in our system is trained on character images obtained from the 3D printer screen. To generalize the OCR part, we may rely on the latest techniques in scene text recognition. Second, differences in screen layout must be addressed. As the layout of the screen varies between models and appliances, manual customization may be necessary. However, labor can be reduced by modularizing functions. Advances in generative AI, such as ChatGPT [8] and Claude-3 [9], could automate this problem to some extent. For example, if the manual for the target model is distributed in PDF format, it would be possible to customize the system with minimal human intervention by loading the manual into a generative AI. We look forward to such a future where AI can significantly reduce the effort required for customization.

## 6 CONCLUSION

Our goal is to empower PVI to create 3D tactile models independently. This autonomy allows them to produce desired objects without relying on sighted individuals, overcoming the challenges and inaccuracies of verbal descriptions. The biggest bottleneck for PVI in using 3D printers independently is the inability to read text on their display. Therefore, this paper specifically focuses on making 3D printers accessible by addressing the challenge of reading scrolling text and menus. Our method for scrolling text is simple yet effective, making it easily applicable to other OCR engines in smartphone apps and wearable devices.

We conducted a user study with 13 PVI, comparing our prototype system with their preferred methods, including smartphone apps, wearable devices, and their own eyes if they have sight. In the experiment of the reading function for menus, most participants completed the tasks faster using our prototype system and preferred it, except for those who used their own eyes. In that for scrolling text, all participants, including those who used their own eyes, recognized the advantages of our prototype system and preferred it. Some participants expressed their desire to see the proposed reading function for scrolling text implemented in smartphone apps. Some suggested improvements to the user interface of the prototype system, such as eliminating silent durations and reducing the response time after user operations. Interestingly, most of their requests aimed to make the behavior of the system similar to that of screen readers.

## REFERENCES


[1] 3D model selection, design and printing by people who are blind or have low vision — Round Table [n. d.]. . Retrieved April 24, 2024 from https://printdisability.org/about-us/accessible-graphics/3d-printing/blvmakers/

[2] 3D4SDGs: 3D Models for People with Visual Impairment [n. d.]. . Retrieved July 1, 2024 from https://3d4sdgs.net/

[3] 3D4VIP: 3D Printing for Education for Visually Impaired Students [n. d.]. . Retrieved July 1, 2024 from https://www.blista.de/EU-Projekt-3D4VIP

[4] Be My Eyes [n. d.]. . Retrieved April 3, 2024 from https://www.bemyeyes.com/

[5] Cynthia L. Bennett. [n. d.]. A toolkit for facilitating accessible design with blind people. 120 ([n. d.]), 16–19. https://doi.org/10.1145/3178412.3178415

[6] Chiraphat Boonnag, Piyalitt Ittichaiwong, Wanumaidah Saengmolee, Narongrid Seesawad, Amrest Chinkamol, Saendee Rattanasomrerk, Kanyakorn Veerakanjana, Kamonwan Thanontip, Warissara

Limpornchitwilai, and Theerawit Wilaiprasitporn. [n. d.]. PACMAN: A Framework for Pulse Oximeter Digit Detection and Reading in a Low-Resource Setting. 10, 15 ([n. d.]), 13196–13204. https://doi.org/10.1109/jiot.2023.3262205

[7] Craig Brown and Amy Hurst. [n. d.]. VizTouch: automatically generated tactile visualizations of coordinate spaces. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction* (2012-02) *(TEI'12)*. ACM. https://doi.org/10.1145/2148131.2148160

[8] ChatGPT [n. d.]. . Retrieved April 25, 2024 from https://chat.openai.com/

[9] Claude-3 [n. d.]. . Retrieved April 25, 2024 from https://claude.ai/

[10] Ahalya Subramanian Simon Liversedge Robin Walker Farah Akthar, Hannah Harvey. 2021. A comparison of reading, in people with simulated and actual central vision loss, with static text, horizontally scrolling text, and rapid serial visual presentation. *ournal of Vision November 2021, Vol.21, 5.* (2021).

[11] Peli E. Fine EM. 1995 Oct. Scrolled and rapid serial visual presentation texts are read at similar rates by the visually impaired. *J Opt Soc Am A Opt Image Sci Vis.1995 Oct;12(10):2286-92.* (1995 Oct).

[12] E. Finnegan, M. Villarroel, C. Velardo, and L. Tarassenko. [n. d.]. Automated method for detecting and reading seven-segment digits from images of blood glucose metres and blood pressure monitors. 43, 6 ([n. d.]), 341–355. https://doi.org/10.1080/03091902.2019.1673844

[13] Giovanni Fusco, Ender Tekin, Nicholas A. Giudice, and James M. Coughlan. 2015. Appliance Displays: Accessibility Challenges and Proposed Solutions. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '15)*. ACM Press. https://doi.org/10.1145/2700648.2811392

[14] Giovanni Fusco, Ender Tekin, Richard E. Ladner, and James M. Coughlan. 2014. Using computer vision to access appliance displays. In *Proceedings of the 16th international ACM SIGACCESS conference on Computers and Accessibility (ASSETS '14)*. ACM Press. https://doi.org/10.1145/2661334.2661404

[15] Anhong Guo, Xiang 'Anthony' Chen, Haoran Qi, Samuel White, Suman Ghosh, Chieko Asakawa, and Jeffrey P. Bigham. [n. d.]. VizLens: A Robust and Interactive Screen Reader for Interfaces in the Real World. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (2016-10) *(UIST '16)*. ACM. https://doi.org/10.1145/2984511.2984518

[16] Amy Hurst and Shaun Kane. [n. d.]. Making "making" accessible. In *Proceedings of the 12th International Conference on Interaction Design and Children* (2013-06) *(IDC '13)*. ACM. https://doi.org/10.1145/2485760.2485883

[17] Caroline Karbowski. [n. d.]. See3D: 3D Printing for People Who Are Blind. 23, 1 ([n. d.]). https://doi.org/10.14448/jsesd.12.0006

[18] Thotreingam Kasar. [n. d.]. *Recognition of Seven-Segment Displays from Images of Digital Energy Meters.* Springer Singapore, 1–10. https://doi.org/10.1007/978-981-13-2514-4_1

[19] Sooyeon Lee, Rui Yu, Jingyi Xie, Syed Masum Billah, and John M. Carroll. [n. d.]. Opportunities for Human-AI Collaboration in Remote Sighted Assistance. In *27th International Conference on Intelligent User Interfaces* (2022-03) *(IUI '22)*. ACM. https://doi.org/10.1145/3490099.3511113

[20] Sebastian Lieb, Benjamin Rosenmeier, Thorsten Thormählen, and Knut Buettner. [n. d.]. Haptic and Auditive Mesh Inspection for Blind 3D Modelers. In *Proceedings of the 22nd International ACM SIGACCESS Conference on Computers and Accessibility* (2020-10) *(ASSETS '20)*. ACM. https://doi.org/10.1145/3373625.3417007

[21] Kazunori Minatani. 2017. An Analysis and Proposal of 3D Printing Applications for the Visually Impaired. In *Harnessing the Power of Technology to Improve Lives, Proceedings of the 14th European Conference on the Advancement of Assistive Technology, AAATE Conf. 2017, Sheffield, UK, September 12-15, 2017 (Studies in Health Technology and Informatics, Vol. 242)*, Peter Cudd and Luc P. de Witte (Eds.). IOS Press, 918–921. https://doi.org/10.3233/978-1-61499-798-6-918

[22] Kazunori Minatani. 2017. *Finding 3D CAD Data Production Methods that Work for People with Visual Impairments.* Springer International Publishing, 548–554. https://doi.org/10.1007/978-3-319-58750-9_76

[23] Kazunori Minatani. 2023. *A Practical CAD Method for the Visually Impaired: A Case of Modeling the Leaning Tower of Pisa.* Springer Nature Switzerland, 440–450. https://doi.org/10.1007/978-3-031-35681-0_29

[24] Lucas P. Moreira. 2022. Automated Medical Device Display Reading Using Deep Learning Object Detection. https://doi.org/10.48550/ARXIV.2210.01325

[25] Tim Morris, Paul Blenkhorn, Luke Crossey, Quang Ngo, Martin Ross, David Werner, and Christina Wong. [n. d.]. Clearspeech: A Display Reader for the Visually Handicapped. 14, 4 ([n. d.]), 492–500. https://doi.org/10.1109/tnsre.2006.881538

[26] Mary C. Potter. 1984. *Rapid Serial Visual Presentation (RSVP).* Routledge.

[27] Irati Rasines, Pedro Iriondo, and Ibai Díez. [n. d.]. *Real-Time Display Recognition System for Visually Impaired.* Springer Berlin Heidelberg, 623–629. https://doi.org/10.1007/978-3-642-31534-3_91

[28] See3D [n. d.]. . Retrieved July 2, 2024 from https://see3d.org/

[29] Alexa F. Siu, Son Kim, Joshua A. Miele, and Sean Follmer. [n. d.]. shapeCAD: An Accessible 3D Modelling Workflow for the Blind and Visually-Impaired Via 2.5D Shape Displays. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility* (2019-10) *(ASSETS '19)*. ACM. https://doi.org/10.1145/3308561.3353782

[30] Ender Tekin, James M. Coughlan, and Huiying Shen. [n. d.]. Real-time detection and reading of LED/LCD displays for visually impaired persons. In *2011 IEEE Workshop on Applications of Computer Vision (WACV)* (2011-01). IEEE. https://doi.org/10.1109/wacv.2011.5711544

[31] Tesseract OCR [n. d.]. . Retrieved April 21, 2024 from https://github.com/tesseract-ocr/tesseract

[32] Harvey H Riazi A Anderson SJ Walker R, Bryan L. 2016 Apr 7. The value of Tablets as reading aids for individuals with central visual field loss: an evaluation of eccentric reading with static and scrolling text. *Ophthalmic Physiol Opt. 2016 Jul;36(4):459-64.* (2016 Apr 7).