A Series of Simple Processing Tools for PDF Files for People with Print Disabilities

Shunsuke Nakamura, Kento Kohase, and Akio Fujiyoshi

Graduate School of Science and Engineering, Ibaraki University, Hitachi, Japan
{20nm726r,20nm711h,akio.fujiyoshi.cs}@vc.ibaraki.ac.jp

Abstract. This paper presents simple processing tools for PDF files for people with print disabilities. They consist of the following three tools: “PDFcontentEraser”, “PDFfontChanger” and “PDFcontentExtracter.” PDFcontentEraser is a tool to remove a certain type of elements in a PDF file. PDFfontChanger is a tool to change a selection of fonts in a document. PDFcontentExtracter is a tool to retrieve the components of a PDF file.

Keywords: Print disability · Accessibility · PDF documents · Universal design font

1 Introduction

In our daily life, we often read digital documents on the web. Official documents from governments, event guides, floor maps of station, and operating manuals of electronic appliances are usually available in the Portable Document Format (PDF). Therefore, the existence of hard-to-read PDF documents is a big issue for all people, especially for people with print disabilities. A print disability is a difficulty or inability of reading printed material because of a visual, physical, perceptual, developmental, cognitive, or learning disability. Figure 1(a) is a typical example of hard-to-read PDF document. The characters stay on the background image, the gradations of color are too steep, and the selection of fonts can be improved. Fortunately, if we process the document, a readable document can be obtained as shown in Fig. 1(b). This study presents simple tools to process PDF files for people with print disabilities.

Editing software for PDF files has not been widely used. There are commercial tools for editing PDF files, such as “Adobe Acrobat Pro” and “Adobe Illustrator.” These tools are capable of editing PDF files freely at a high quality. However, they are not developed for general people, but for DTP engineers. Their prices are very high, and a lot of training is necessary to master them. There are some free tools for handling PDF files such as PDFBox application [1] and PDFtk server [2]. However, they are not user-friendly and does not have enough function for editing to produce readable PDF files.
The authors have experience of handling PDF files because the authors’ laboratory has been producing Multimodal Textbooks [3,4] for students with print disabilities. Multimodal Textbooks are paper-based textbooks with audio support utilizing invisible 2-dimensional codes and digital audio players with a 2-dimensional code scanner. They were used by 1,110 students in 2018 in Japan. The simple processing tools for PDF files presented in this paper are a reformulation of software developed for the production of Multimodal Textbooks.

The simple processing tools for PDF files are designed for general users including people with print disabilities. They consist of the following three tools: “PDFcontentEraser”, “PDFfontChanger” and “PDFcontentExctracter.” They are developed in Java using Apache PDFBox library and run on Windows. The usage is very simple. When they are installed on a PC, icons are placed on the desktop as shown in Fig. 2. A PDF file is processed by a drag-and-drop operation to these icons.

PDFcontentEraser is a tool to remove a certain type of elements in a PDF file. A PDF file mainly consists of characters, images, paths, and shadings. This tool can selectively erase all elements of a certain type. PDFfontChanger is a tool to change a selection of fonts in a document. As a default setting, all fonts are replaced by “Universal Design Font for Digital Textbooks” (UD font) [5] developed by Morisawa Inc. The converted document (Fig. 1(b)) is obtained by erasing all images and all shadings using PDFcontentEraser and by changing...
all fonts to the UD font from the original document (Fig. 1(a)). PDFcontentExtractor is a tool to retrieve the components of a PDF file. It can output a retrieved information in XML format.

To evaluate the usability of the simple tools, we measure the working times for conversions of a PDF file using the simple tools and Adobe Acrobat Pro. With the simple tools, the conversions were finished much faster without any mistakes.

2 Simple Processing Tools

The simple processing tools for PDF files are developed on Java with the Apache PDFBox library and run on Windows.

2.1 PDFcontentEraser

This tool is to remove a certain type of elements in a PDF file. As shown in Fig. 3, main elements of a PDF file are characters, images, paths, and shadings. When this tool is installed on a PC, 4 icons are placed on the desktop (the upper row of Fig. 2). The way to use this tool is just a drag-and-drop operation of a PDF file to one of the icons. An output file is created in the same folder with a new file name (the original file name + “.wo_” + the initial letter of the erased elements). If we want to delete more than one type of elements, we can continue the drag-and-drop operations.
When a person wants to remove annoying background under characters in a document, the tool can remove the images and shadings on the background. In addition, the tool helps for productions of large-printed teaching materials. When characters stay on images overlappingly, the tool can erase all characters over images, and we can obtain clean images from a PDF file.

PDFcontentEraser is a convenient tool for all people to make a readable PDF files. It has been used to create large-printed textbooks for low vision students in Japan.

2.2 PDFfontChanger

This tool is to change a selection of fonts in a document. The usage of the tool is also a drag-and-drop operation to the icon. As a default setting, all fonts are replaced by “Universal Design Font for Digital Textbooks” (UD font) [5] developed by Morisawa Inc. (Fig. 4). From the “Windows 10 Fall Creators Update” in 2017, UD font becomes one of standard fonts in Windows 10 with Japanese language pack. If UD font is not installed on a PC, the tool uses Arial font instead.

![Fig. 3. Main elements of a PDF file](image)

![Fig. 4. Sample of Glyph of UD font, Arial, and Georgia](image)

Though we can change a selection of fonts using Adobe Acrobat Pro, page-layouts of the document might be broken sometimes due to differences of glyph
metrics of fonts. PDFfontChanger can replace fonts without any change of page-layouts because each character is replaced by the same character of another font with the same size at the same position.

PDFfontChanger should be used by someone incapable of reading specific types of fonts. In Japan, there are some children who cannot read Mincho font. It has been used to change fonts of large-printed textbooks for low vision and dyslexic students in Japan.

Unlike a conversion to PDF/UA (PDF/Universal Accessibility), the tool does not change the reading order of an original PDF file because this tool is mainly designed for a person who wants to read a document on a screen or a sheet of printed paper. Elementary and junior high school teachers may also want to use this tool. After creating printed materials to be distributed, all the fonts can be replaced for some students who have difficulty of reading particular fonts.

2.3 PDFcontentExtractor

This tool is to retrieve the components of a PDF file. It is a command line tool. It can extract images, character information (position, bounding box, font name, Unicode, and color) and the shapes of paths. This tool can output in the following three forms: (1) character information, image location and path location in XML, (2) image files in PNG, and (3) the shapes of paths, the shapes of characters and the locations of images in SVG (Fig. 5).

PDFcontentExtractor can be used for the development of new PDF processing tools by others.

![Character information](image)

**Fig. 5.** Types of outputs of PDFcontentExtractor

3 Evaluation

In order to evaluate the usability of the simple processing tools, we compare the time needed to convert a PDF file by using the simple tools and Adobe Acrobat Pro.
3.1 Method

For experimental subjects, 3 university students were employed. They are familiar with using Acrobat Pro.

They were asked to convert the PDF file shown in Fig. 1(a) to the new one shown in Fig. 1(b) by deleting all images and shadings and replacing all font to UD font.

As shown in Fig. 6, the task can be done using PDFcontentEraser and PDFfontChanger as follows. First, drag-and-drop the file to the “Erase Image” icon and the “Erase Shading” icon. Next, drag-and-drop the output file to the “Font to UD” icon. Then, the requested PDF file is obtained.

Using Acrobat Pro, on the other hand, there are two ways to delete elements in a PDF file: (1) clicking items directly on the document view and pushing the delete key, and (2) showing the content panel, and selecting items there. Changing a selection of fonts can be done by selecting text objects on the document view and choosing a new font from a drop-down list in the side panel. When a selection of fonts is changed, the layout of some text objects becomes broken and needs to be fixed.

Fig. 6. Operations using processing tools
3.2 Result

The result is shown in Table 1. All subject can convert the document faster with the simple tools than Acrobat Pro. When they used Acrobat Pro, all subjects choose the document view to select items. They made some mistakes using Acrobat Pro; Subject A erased a text object and left a shading object, Subject B selected a wrong font and forgot to fix the layout of a text object, and Subject C deleted a necessary path object.

|             | Working time | Number of mistakes |
|-------------|--------------|--------------------|
|             | Acrobat Pro  | Simple tools       |
| Subject A   | 1 min 31 sec | 12 sec             |
|             | 2            | 0                  |
| Subject B   | 44 sec       | 10 sec             |
|             | 2            | 0                  |
| Subject C   | 1 min 10 sec | 12 sec             |
|             | 1            | 0                  |

4 Conclusion

This paper has presented simple processing tools for PDF files. These tools were developed to be used by all people, especially for people with print disabilities and their supporters. From the result of the evaluation, the usability of the simple tools was demonstrated. The simple processing tools for PDF files are available on the website of authors’ laboratory:

http://apricot.cis.ibaraki.ac.jp/PDFtools/

In the future, we would like to explore collaboration between the simple tools and PDF/UA converters.

References

1. Apache PDFBox. https://pdfbox.apache.org/
2. PDF Labs PDFtk. https://www.pdflabs.com/tools/pdftk-the-pdf-toolkit/
3. Fujiyoshi, A., Fujiyoshi, M., Ohsawa, A., Ota, Y.: Development of multimodal textbooks with invisible 2-dimensional codes for students with print disabilities. In: Miesenberger, K., Fels, D., Archambault, D., Peñáz, P., Zagler, W. (eds.) ICCHP 2014. LNCS, vol. 8548, pp. 331–337. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08599-9_50
4. Takaira, T., Tani, Y., Fujiyoshi, A.: Development of a unified production system for various types of accessible textbooks. In: Miesenberger, K., Bühler, C., Penaz, P. (eds.) ICCHP 2016. LNCS, vol. 9758, pp. 381–388. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-41264-1_52
5. Morisawa Inc., Universal Design Fonts (in Japanese). https://www.morisawa.co.jp/fonts/udfont/