

EPUB eBook Converting Schemes for Improving User Interactions

Namhui Lee[†] · Jai-Hoon Kim^{**} · Kangseok Kim^{***}

ABSTRACT

To access PDF documents on an electronic book, PDF documents need to be converted into EPUB which is a standard format of the electronic book. When converting a PDF document into EPUB format, we need to convert color representations from CMYK into RGB representation. It is possible to give a visual effect and a user interaction using a video and JavaScript supported by EPUB format. The schemes for converting from PDF to EPUB are studied in this paper. (1) The first study is to carry out not to lose the color conversion effect using an ICC profile. (2) The second one is a layout configuration in the conversion process. (3) The third one is to highlight a specific content such as quiz platform to provide interactive visual effect for electronic book readers. Finally, in this paper we will show the usability of EPUB based eBook converting scheme through user study.

Keywords : EPUB, PDF, eBook, ICC Profile, Interactive Effect

사용자의 인터렉션 향상을 위한 EPUB eBook 변환 기법

이 남 희[†] · 김 재 훈^{**} · 김 강 석^{***}

요 약

기존 책 문서인 PDF문서를 전자책에서 사용 가능하도록 전자책의 표준 형태로 변환하는 과정이 필요하다. PDF문서를 전자책의 표준 형태인 EPUB으로 변환할 때, 인쇄 색상 표현방법인 CMYK를 디지털 색상 RGB 형태로 변환하는 과정에서 형태의 차이로 인해 색감이 제대로 변환되지 못하는 문제점이 있다. 그리고 EPUB은 동영상, 자바스크립트 등을 이용해서 시각적인 효과를 줄 수 있으며, 사용자와의 상호작용이 가능하다. 본 연구는 PDF를 EPUB으로 변환 시 발생할 수 있는 문제점에 대해서 연구를 하였다. (1) 첫 번째는 색감을 잃지 않도록 ICC 프로파일을 이용한 변환 연구를 수행하였다. (2) 두 번째는 변환 과정에서 레이아웃 구성에 대한 연구를 수행하였다. (3) 세 번째는 전자책 독자와 상호작용할 수 있는 시각적인 효과를 제공하기 위하여, 텍스트 중 특정 부분을 인식하여 시각적인 효과를 주는 연구를 수행하였으며 기존 책에서의 퀴즈를 추출하여 변환 시 자바스크립트를 이용한 퀴즈 플랫폼 프로토타입을 구현하였다. 또한 PDF문서를 EPUB에 기반 한 전자책 문서로 변환 후 사용자 평가를 통해 변환유용성 및 편의성을 검증하였다.

키워드 : EPUB, PDF, 전자책, ICC 프로파일, 인터랙티브 효과

1. 서 론

출판 시장의 흐름은 종이에 인쇄하여 책과 같은 형태의 출판물 시대에서 벗어나, 최근 IT 기술의 발달로 개인 스마트폰, 태블릿, 노트북 등을 이용해 앱 스토어 같은 북 스토어에서 온라인 디지털 형태로 출판되는 추세로 바뀌어 가고 있다. 전자책(eBook) 출판 형태는 두 가지로 나뉘는데 첫째

는 모바일 애플리케이션 형태의 앱북과 두 번째는 전자 문서로 제작하여 별도의 뷰어를 통해 읽을 수 있는 전자책 문서 형태이다[1]. 앱북은 개인 디바이스에만 국한되어 확장의 어려움이 있어, 현재는 전자책 문서 형태를 선호하는 시장이 형성되어 가고 있다.

현재까지 출판된 책들을 다시 전자책으로 만드는 과정은 PDF(Portable Document Format)나 종이로 된 책을 사람이 수작업을 통하여 제작하고 있다. 따라서 기존에 책을 제작했던 작업 시간과 인건비용이 재투자 되어 낭비되는 부분이 많다. 또한, 전자책 초창기에는 PDF문서 자체를 보여주는 전자책 시장이었지만, 다양한 콘텐츠를 담아내기에는 PDF문서가 미디어를 처리하는데 한계가 있다. 이러한 문제를 해결하기 위하여 현재 PDF를 EPUB(Electronic Publication)[2]의 형태로 변환하는 도구가 개발되어 전자책 콘텐츠 제작자들이 사

※ 이 논문은 미래창조과학부의 2015년 고용계약형 SW석사과정 지원 사업을 지원받아 수행한 결과임.
† 정 회 원 : 아주대학교 지식정보공학과 IoT전공 석사
** 정 회 원 : 아주대학교 사이버보안학과 교수
*** 정 회 원 : 아주대학교 사이버보안학과 부교수
Manuscript Received : December 24, 2015
First Revision : October 12, 2016
Second Revision : November 2, 2016
Accepted : November 2, 2016
* Corresponding Author : Kangseok Kim(kangskim@ajou.ac.kr)

용하고 있다. 그러나 폰트가 추출 되지 않는 문제, 글자 누락, 색감 저조, 표 추출 누락 등으로 완벽히 변환되지 않아 아직은 미흡한 실정이다. 그리고 EPUB 3.0이 다양한 콘텐츠의 생성을 통해 인터랙티브 효과를 주어 사용자의 시각적인 만족과 사용자가 직접 참여할 수 있는 연구가 진행되고 있다[3].

본 논문에서는 PDF에서 EPUB으로 변환을 위하여 다음과 같은 연구를 진행하였다.

- 폰트 추출에 있어서 인쇄 색상인 CMYK 에서 웹 기준 색상 표현 RGB로 변환 시 색감 보존 추출 연구: 변환 시 문제점인 색감처리 부분을 개선하기 위하여 CMYK에서 RGB 색상으로 ICC(International Color Consortium: 국제컬러협회) 파일을 이용한 변환 방법
- PDF에서 EPUB으로 변환 시 레이아웃 구성에 대한 구현: 서로 다른 레이아웃 구조 비교 분석 및 추출을 통한 EPUB 표준에 맞게 파일을 배치하고 본문 내에 위치 계산방법
- 상호작용 효과: EPUB의 특징인 상호작용 효과를 단순히 고정적인 글자에서 각 페이지 마다 시각적인 효과를 주기 위해 제목과 같은 강조할 문구를 찾아 효과를 줄 수 있는 알고리즘 연구와 PDF 내의 퀴즈 문구를 분석하고 퀴즈 폼 자동완성 기능 연구

본 논문은 위의 연구를 통하여 인쇄 색감에 동일한 색감의 표현을 위한 연구를 수행하였으며, 문서 변환 시 전체 페이지의 특정 글자 부분에 상호작용 효과를 주어 시각적 효과를 도출하기 위한 연구를 수행하였으며, 또한 단순히 책을 보는 것을 넘어서 퀴즈를 직접 풀면서 책과의 상호작용을 통한 참여 효과를 주기 위한 연구를 수행하였다.

2. 배경연구

PDF는 전자 문서로 문자, 도형, 그림, 글꼴 등을 표현할 수 있다. 대부분의 문서를 표현할 수 있고 압축 및 암호화를 통해 변조가 용이하지 않고, 아도비 사에서 PDF문서를 볼 수 있는 아크로벳 리더를 무료로 배포하여 기업에서 외부 문서나 개인은 제출 문서 형태로 이용하고 있다[4].

EPUB은 다양한 기기에 맞도록 최적화된 콘텐츠를 볼 수 있게 자동공간조정(Reflowable), 고정 레이아웃(Fixed Layout)을 지원한다. 또한 HTML5, CSS3.0, Javascript의 사용이 가능하여 다양한 언어, 동영상, 음악, 상호작용효과, 폰트내장, MathML 등을 표현하고 기능 구현이 가능하다[5].

오픈지디지트[6]에서 개발된 'ViewPorter PDF to EPUB 3.0 Converter' 변환 툴은 기존의 인쇄물 전자 저장 방식인 PDF를 전자책 표준인 EPUB 3.0 포맷으로 변환해 주는 도구이다. 본 연구에서 사용된 'ViewPorter PDF to EPUB 3.0 Converter'의 변환 과정은 Fig. 1과 같다. PDF문서의 변환 시 먼저 폰트와 글꼴 정보를 추출하고 추출이 되지 않은 폰트는 OS 시스템 내의 내장 폰트와 이름이 같은 폰트 정보

를 가져온다. 두 번째로는 각 이미지를 추출하고 글자 이외에 도형이나 배경 색을 배경이미지로 추출하여 image 폴더에 저장한다. 세 번째로 글자를 추출하게 되는데 글자 추출 시 폰트 타입, 폰트 이름, 자간, 행간, 글자 크기 등의 정보를 같이 추출한다. 네 번째로는 각 추출된 정보로 기본 디자인 정보를 가지는 CSS 파일과 콘텐츠 정보를 가지는 XHTML 파일을 생성하고, 추출된 정보를 토대로 opf 등 EPUB 구조 파일을 생성한다. 마지막으로 추출되고 생성된 파일을 컨테이너에 담는 과정을 거치는데 일반적으로 zip 압축 방식을 통해 epub 확장자를 가지는 파일을 생성한다.

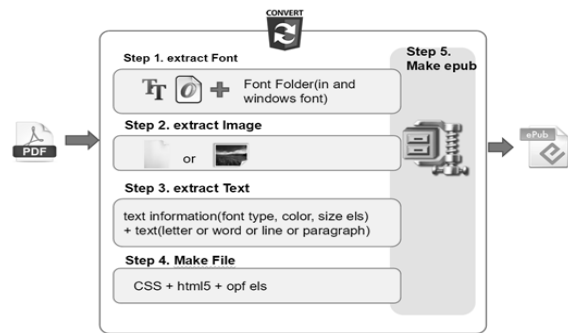


Fig. 1. Conversion Process from PDF to EPUB 3.0 Tools

3. 관련 연구분석

3.1 CMYK에서 RGB로의 색상 변환 분석

색 혼합에는 여러 방식이 있는데 본 연구에서 분석할 방식은 사이안(Cyan), 마젠타(Magenta), 노랑색(Yellow), 검정(Black)을 섞어서 명도와 채도를 낮추는 CMYK 감산혼합과 빛의 삼원색을 이용한 RGB 가산혼합 방식을 사용하였다.

CMYK 감산 혼합은 사이안, 마젠타, 노랑, 검정의 값을 혼합하면서 하얀색 이외에 색들을 표현한다. 실물의 색상과 일치하기 때문에 인쇄와 사진에서의 색을 표기할 때 사용하는 방식이다. RGB 가산 혼합은 빨강, 녹색, 파랑 3가지 색을 조합하여 표현하는데 이 방식은 주로 웹이나 디지털 콘텐츠의 색을 나타내는데 사용된다. 감산혼합 방식을 가산혼합 방식으로 색을 변환하려면 대략 두 가지 방법이 있는데 첫 번째로는 단순계산 방식이고 두 번째는 색 공간을 이용한 방법이다.

단순계산에 의한 변환의 수식[7]은 Fig. 2와 같다. 기존 연구의 수식을 보면 먼저 CMYK에서 검정색을 뺀 CMY로 변환한다. 이 과정에서 검정이 빠지면서 나머지 세 가지 색 값에 1에서 검정색 값을 뺀 값을 곱하여 다시 검정색 값을 더한다. 그 값과 다시 1보다 작은 값을 각 색 값으로 얻는다. 이렇게 CMY 값을 얻은 다음 1에서 사이안 값을 빼 빨강 색 값을, 1에서 마젠타 값을 빼 녹색 값을, 마지막으로 1에서 노랑색을 빼 파랑색 값을 구한다. 이 과정을 통해 RGB 색의 값을 구한다. 그러나 이렇게 단순 값을 얻게 되면 실제로 보는 색과 차이를 보일 수 있다.

CMYK to CMY

Cyan = minimum(1,Cyan*(1-Black) + Black)
 Magenta = minimum(1,Magenta*(1-Black) + Black)
 Yellow = minimum(1,Yellow*(1-Black) + Black)

CMY to RGB

Red = 1-Cyan
 Green = 1-Magenta
 Blue = 1-Yellow

Fig. 2. Simple Conversion Formula from CMYK to RGB

3.2 PDF와 EPUB 3.0의 레이아웃 비교 분석

PDF는 오브젝트 단위의 코드로 구성되며 EPUB은 이미지, 배치파일, 구조파일, 내용 등이 파일, 폴더가 일정한 규칙에 따라 위치하여 압축한 형태이다. Fig. 3은 PDF에서의 'Hello World'를 표현한 구조이며 Fig. 4는 EPUB 3.0으로 'Hello World'를 제작한 구조이다. 폴더 형태로 파일 별로 구조화가 되어 있으며 콘텐츠 내용이 있는 부분은 HTML 문서 형태로 표현된다.

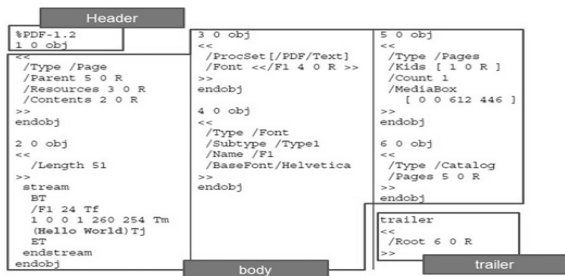


Fig. 3. 'Hello World' Representation Structure in PDF



Fig. 4. 'Hello World' Representation Structure in EPUB 3.0

3.3 EPUB 3.0에서 인터랙티브 효과 기존 연구 분석

EPUB 3.0은 웹처럼 자바스크립트, CSS, 동영상, 이미지 등을 이용한 사용자와의 상호 작용을 지원한다. 기존 연구에는 교육적인 측면을 구현한 사례와 시각적 요소 구현, 구글 지도를 전자책에 삽입하여 지형적인 부분을 표현하기도 했다[8]. [9]에서는 사용자가 참여 가능한 게임적요소인 조각 퍼즐 게임을 전자책에 구현하였다. 실질적으로 사용되고 있

는 콘텐츠에는 동영상 및 음악 재생, 디지털 사진첩, 디지털 교과서 등 많은 분야에서 전자책의 인터랙티브 효과들을 구현하고 있다.

4. PDF를 EPUB으로 변환을 위한 구현방법

본 장에서는 PDF를 EPUB으로 변환 시 발생할 수 있는 색상 및 레이아웃 구성 문제를 해결하기 위한 구현방법 및 EPUB 기반 전자책 독자와 상호작용할 수 있는 시각적인 효과를 주기위한 구현방법에 대하여 간략히 기술하였다. 5장에서 변환 후 구현결과에 대해 자세히 설명할 것이다. 개발 환경은 Table 1과 같다. 개발 언어는 PDF문서를 분석할 수 있는 PDFBox 라이브러리[10]를 이용하기 위해 Java 1.7을 사용하였고 기본 'ViewPorter PDF to EPUB 3.0 Converter'에 기능 추가 형식으로 구현하였다.

Table 1. Development Environment

classification	Detail
Language	Java 1.7, Javascript
Converter	ViewPorter PDF to EPUB 3.0
PDF Library	Apache PDFBox
Epub Editor	ViewPorter Editor 2.0
Epub Viewer	iBooks, Tabonbooks, Radium, ViewPorter Epub Viewer 1.1

4.1 변환 시 CMYK에서 RGB 색상보존 추출

PDF문서가 CMYK 색상의 문서 일 경우 RGB 형태로 변환하기 위하여 ICC가 공표한 표준, 즉 색 입력 장치 또는 색 출력 장치의 특성을 구현하는 데이터 집합인 ICC 프로파일을 이용하였다. 프린터, 모니터 등 각종 입력 장치나 출력장치는 정의한 ICC 프로파일[11]이 있는데 본 연구에서는 PDF를 개발한 아도비 시스템의 Acrobat Reader에서 CMYK 색상을 표현하기 위해 기본으로 설정 되어있는 "Japan Color 2001 Coated" ICC 프로파일과 윈도우즈 디스플레이 색 시스템의 기본인 sRGB ICC 프로파일을 이용하여 구현하였다.

4.2 PDF를 EPUB 3.0으로 변환 시 고정레이아웃 구현

PDF에서 EPUB으로 변환 시 레이아웃 구성에 대한 구현을 위해 서로 다른 레이아웃 구조 비교 분석 및 추출을 통한 EPUB 표준에 맞게 파일을 배치하고 본문 내에 위치 계산방법을 사용하였다.

4.3 EPUB 3.0 인터랙티브 기능 구현

인터랙티브 기능은 변환 과정에서 시각 효과인 Fade 효과 삽입 방법과 사용자 참여가 가능한 퀴즈폼(Quiz Form) 자동완성에 대해 구현하였다. Fig. 5는 변환 부분을 나타내는 예시이다.

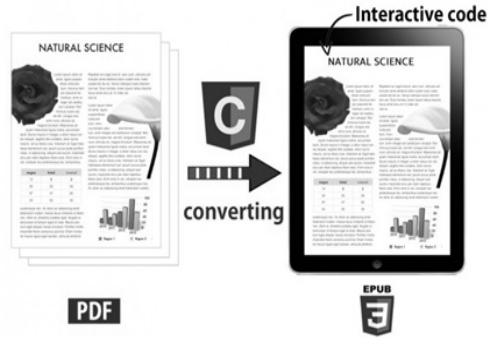


Fig. 5. An Example of Interaction

5. PDF를 EPUB 기반 전자책 변환 분석

본 장에서는 변환 시 CMYK에서 RGB 색상 보존 추출, 고정레이아웃, 및 인터랙티브 기능분석에 대하여 기술하였다.

5.1 변환 시 CMYK에서 RGB 색상 보존 추출

PDF문서가 CMYK 색상의 문서 일 경우 RGB 형태로 변환해야 하는데 변환공식으로 계산 할 경우 색이 일치하지 않는다. 변환 공식에 대입하여 계산했을 경우 Table 2와 같이 값이 계산되며 그 경우 Fig. 6과 같이 육안으로도 색감이 확연히 다르다는 것을 알 수 있다. 그러한 문제를 해결하기 위하여 4장에서 언급되어진 ICC 프로파일을 이용하였다. “Japan Color 2001 Coated” ICC 프로파일과 윈도우즈 디스플레이 색 시스템의 기본인 sRGB ICC 프로파일을 이용하여 색을 표현하기 위해서는 공통된 색 공간을 거쳐야 하는데 각 프로파일들은 CIE (International Commission on Illumination: 국제조명위원회) XYZ 색 공간 값을 계산할 수 있다. CIE XYZ 색 공간은 인간 색채 인지에 대한 연구를 바탕으로 수학적으로 정의된 색 공간의 하나이며 색상 공간에서 유도되는 색도 불변의 색상 공간으로서, XYZ색상 공간을 조도 성분과 색도 성분으로 분리함으로써 얻어진다 [12]. Fig. 7은 색 변환 코드로써, 먼저 추출하고자 하는 텍스트의 색이 CMYK 일 경우 해당 값을 “Japan Color 2001 Coated” ICC 프로파일로 구성된 색을 입력하여 CIE XYZ 값을 도출 한 뒤에 이 값을 “sRGB” ICC 프로파일에 입력하고 RGB 값을 추출하면 색의 손실 없이 변환할 수 있다.

Fig. 8에서 색 공간 값인 CIE XYZ 값의 차이를 그래프로 비교하였다. 여기에서 색 공간(color space)은 CIE RGB, CIE XYZ 등의 색 체계를 3차원으로 표현한 공간 개념이다. 또한 RGB 색상 값에서도 많은 차이가 나타나는데 Table 3을 보면 Cyan의 Green 수치는 무려 93이나 차이를 보이면서 단순 계산과 ICC 프로파일을 이용하는 것에 대한 분명한 차이를 확인할 수 있다. Fig. 9를 보면 색을 추출하고 변환함에 있어서 차이는 있지만 ICC 프로파일을 이용한 변환이 Fig. 6 보다 좋아진 것을 확인할 수 있다.

Table 2. Simple Calculation of Color Values

색상	CMYK (%)	단순계산 (R,G,B)
Cyan	(100,0,0,0)	(0,255,255)
Magenta	(0,100,0,0)	(255,0,255)
Yellow	(0,0,100,0)	(255,255,0)
Red	(0,100,100,0)	(255,0,0)
Green	(100,0,100,0)	(0,255,0)
Blue	(100,100,0,0)	(0,0,255)

Cyan Cyan
Magenta Magenta
Yellow Yellow
Black Black
Red Red
Green Green
Blue Blue

Fig. 6. Simple Conversion Extraction in Text

```
IF colorspace type of text == CMYK THEN
  CIEXYZ = CMYK_ColorSpace.
    toCIEXYZ(colorspace valueof text)
  RGB value = sRGB_colorSpace.fromCIEXYZ(CIEXYZ)
```

Fig. 7. Conversion Pseudo Code

Table 3. Used ICC Profiles

색상	CMYK (%)	단순계산 (R,G,B)	ICC 적용 (R,G,B)
Cyan	(100,0,0,0)	(0,255,255)	(0,162,232)
Magenta	(0,100,0,0)	(255,0,255)	(228,3,129)
Yellow	(0,0,100,0)	(255,255,0)	(253,240,10)
Red	(0,100,100,0)	(255,0,0)	(229,24,35)
Green	(100,0,100,0)	(0,255,0)	(0,155,74)
Blue	(100,100,0,0)	(0,0,255)	(31,51,138)

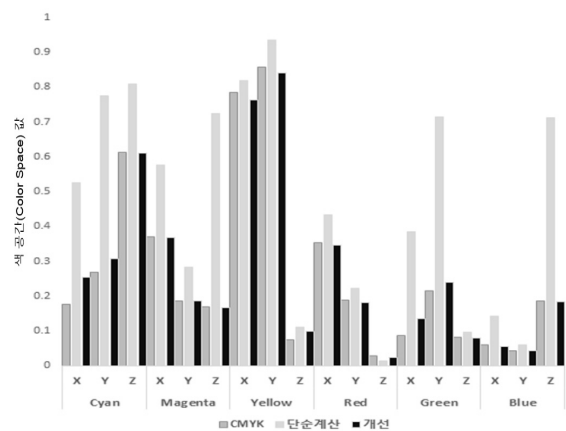


Fig. 8. Comparison to CIE XYZ



Fig. 9. Text Conversion using ICC Profiles

5.2 PDF에서 EPUB 3.0으로 변환 시 고정레이아웃 구현

변환 과정을 거치면 EPUB 3.0의 구조와 파일 목록을 나타내는 content.opf 파일이 Fig. 10과 같이 생성되는데 레이아웃 형태를 나타내는 meta property= "rendition:레이아웃" 속성에 고정형 레이아웃의 값인 'pre-paginated'가 할당되며, 또한 콘텐츠의 방향과 보여지는 방식을 나타내는 속성의 meta property = "rendition:orientation" 와 meta property="rendition: spread"의 값을 'auto'로 할당하여 사용자가 EPUB 3.0을 읽기 위해 사용하는 뷰어 설정에 따라 볼 수 있도록 하였다.

```

1 <?xml version="1.0" encoding="utf-8" standalone="yes"?>
2 <package xmlns="http://www.idpf.org/2007/opf" prefix="rendition: http://www.idpf.org/vocab/rendition/"
3 <metadata xmlns:dc="http://purl.org/dc/elements/1.1/"
4 <dc:identifier id="BookId" urn:uuid:029d2624-1a37-409c-aa36-aa61e9d22497</dc:identifier>
5 <!-- fixed layout options -->
6 <meta property="rendition:layout" pre-paginated</meta>
7 <meta property="rendition:orientation" auto</meta>
8 <meta property="rendition:spread" auto</meta>
9 <meta property="dcterms:modified" 2015-09-05T00:00:00Z</meta>
10 <dc:date id="modification" 2015-09-05</dc:date>
11 <dc:title id="vp_page_1_1" />
12 <dc:creator Orange Digit Inc.</dc:creator>
13 <dc:language en</dc:language>
14 <dc:type 3.0</dc:type>
15 <dc:date xmlns:opf="http://www.idpf.org/2007/opf" opf:event="modification" 2014-03-21</dc:date>
16 </metadata>
17 <manifest>
18 <item href="contents/toc.xhtml" id="toc" media-type="application/xhtml+xml" properties="nav" />
19 <item id="vp_page_1_1.xhtml" href="contents/vp_page_1_1.xhtml" media-type="application/xhtml+xml"
20 <item id="vp_page_1_2.xhtml" href="contents/vp_page_1_2.xhtml" media-type="application/xhtml+xml"
21 <item id="vp_page_1_3.xhtml" href="contents/vp_page_1_3.xhtml" media-type="application/xhtml+xml"
22 <item id="vp_page_1_4.xhtml" href="contents/vp_page_1_4.xhtml" media-type="application/xhtml+xml"
23 <item id="vp_page_1_5.xhtml" href="contents/vp_page_1_5.xhtml" media-type="application/xhtml+xml"
24 <item id="vp_page_1_6.xhtml" href="contents/vp_page_1_6.xhtml" media-type="application/xhtml+xml"
25 <item id="vp_page_1_7.xhtml" href="contents/vp_page_1_7.xhtml" media-type="application/xhtml+xml"
26 <item id="vp_page_1_8.xhtml" href="contents/vp_page_1_8.xhtml" media-type="application/xhtml+xml"
    
```

Fig. 10. Converted content.opf File

$$E(sx, sy) = P(x, y) \times s$$

- s : scale (default 2.0)
- x : horizontal position
- y : vertical position
- P() : PDF File element position
- E() : EPUB3.0 element position

Fig. 11. Formula for Calculating Object Position

각 객체 별 위치 값을 Fig. 11의 수식으로 계산되어 배치 되는데, 배율 값은 기본 2.0이지만 변경이 가능하다. 기본 배율 s에 각 객체의 가로, 세로 값인 x, y에 s를 곱해서

EPUB 3.0 객체 위치 값을 구한다. Fig. 12는 변환 된 각 콘텐츠의 내용에 해당하는 XHTML 파일의 소스코드이다. 'Background Image'로 표시된 부분은 추출된 배경 이미지와 배치 위치 및 크기가 속성으로 들어가고 아래의 'Text' 부분에는 클래스 명에 글자 위치 부분을 나타내는 'text*'와 폰트 속성과 글자크기, 색상 등의 'fnt*'가 명시된다.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <html xmlns="http://www.w3.org/1999/xhtml" xmlns:epub="http://www.idpf.org/2007/ops" xmlns:ibooks="http://vocabularies.
3 <head>
4 <title>xx.html</title>
5 <link rel="stylesheet" type="text/css" href="css/vp_style.css" />
6 <link rel="stylesheet" type="text/css" href="css/page2.css" />
7 <meta name="viewport" content="width=1224, height=1584"/>
8 </head>
9 <body style="margin:0px;padding:0px;">
10 <div style="width: 100%; height: 100%; background-color: white;">
11 
12 </div>
13 <div class="vp_textbox text674 fnt16">Normal</div>
14 <div class="vp_textbox text675 fnt16">Image</div>
15 <div class="vp_textbox text676 fnt16">Caption</div>
16 <div class="vp_textbox text677 fnt16">Generation</div>
17 <div class="vp_textbox text678 fnt16">Title</div>
18 <div class="vp_textbox text679 fnt16">Visual</div>
19 <div class="vp_textbox text680 fnt16">Attention</div>
20 <div class="vp_textbox text681 fnt16">Figure</div>
21 <div class="vp_textbox text682 fnt16">Table</div>
22 <div class="vp_textbox text683 fnt16">Attention</div>
23 <div class="vp_textbox text684 fnt16">Cover</div>
24 <div class="vp_textbox text685 fnt16">Time</div>
25 <div class="vp_textbox text686 fnt16">As</div>
26 <div class="vp_textbox text687 fnt16">The</div>
    
```

Fig. 12. Source Code of xx.xhtml File

Fig. 13은 폰트 정보와 폰트 파일 위치를 나타내는 부분과 각 페이지의 글자 속성 값을 보여주고 있으며, CSS 클래스 값에는 색상, 크기, 폰트, 자간 등이 포함된다. Fig. 14는 각 페이지 안의 배경색 값과 각 글자 별 정렬, 위치 값을 보여주고 있다.

```

1 @font-face {
2 font-family: 'NimbusRomNo9L-Regu'; src: url(../fonts/NotoSans.ttf);
3 .fnt3 {font-size: 15px; font-family: 'NimbusRomNo9L-Regu'; color: RGB(0, 0, 0); left:
4 .fnt4 {font-size: 19px; font-family: 'NimbusRomNo9L-Regu'; color: RGB(0, 0, 0); left:
5 .fnt6 {font-size: 19px; font-family: 'NimbusRomNo9L-Regu'; color: RGB(0, 20, 115); left:
6 .fnt8 {font-size: 17px; font-family: 'NimbusRomNo9L-Regu'; color: RGB(0, 0, 0); left:
7 .fnt9 {font-size: 17px; font-family: 'NimbusRomNo9L-Regu'; color: RGB(0, 20, 115); left:
8 .fnt36 {font-size: 13px; font-family: 'NimbusRomNo9L-Regu'; color: RGB(0, 0, 0); left:
9 .fnt39 {font-size: 11px; font-family: 'NimbusRomNo9L-Regu'; color: RGB(0, 0, 0); left:
    
```

Fig. 13. CSS File for Fonts Information

```

1 body {
2 background-color: white;
3 }
4 .text674 { position: absolute; top: 90.28522px; left: 381.528px; }
5 .text675 { position: absolute; top: 90.28522px; left: 438.80536px; }
6 .text676 { position: absolute; top: 90.28522px; left: 491.0974px; }
7 .text677 { position: absolute; top: 90.28522px; left: 557.35913px; }
8 .text678 { position: absolute; top: 90.28522px; left: 648.5116px; }
9 .text679 { position: absolute; top: 90.28522px; left: 686.8699px; }
10 .text680 { position: absolute; top: 90.28522px; left: 739.5205px; }
11 .text681 { position: absolute; top: 148.3432px; left: 110.88px; }
    
```

Fig. 14. CSS File for Text Color and Text Position

5.3 EPUB 3.0 인터랙티브 기능 구현

인터랙티브 기능은 첫 번째로 기존 발표문서나 웹에 있어서 헤드라인을 강조, 주목을 끌기 위하여 애니메이션 기능이나 나타내기 등과 같은 효과를 주기도 하였다. 기존에 변환 시에는 그러한 효과들을 작업자의 손으로 코드를 삽입해야 했고 그에 따른 시간과 비용은 낭비됐다. 이러한 부분을 변환 툴에 삽입하여 사람이 일일이 하지 않고 한 번에 도구를 이용하여 자동으로 추출, 효과 삽입을 하였다. 그 방법은

문서 변환 시 제목 부분을 찾아내고 그 부분에 애니메이션 기능이나 나타내기 효과를 삽입한다. 문서에서 제목부분이나 강조하기 위한 방법은 글자를 본문보다 크게 설정하거나, 굵게 표시하거나, 검정색이 아닌 다른 색을 부여한다. 위와 같은 부분을 Fig. 15의 코드를 이용하여 찾아내고 따로 분류를 해서 그 부분에 효과를 삽입하도록 하였다. 물론 다양한 형태의 문서가 존재하지만, 그 부분은 변환을 하였다. Fig. 16의 CSS 소스코드는 Fade In 동작 예시 코드이다. 알고리즘으로 찾아낸 글자들을 XHTML 코드 내에서 Fig. 17의 예시처럼 효과가 삽입이 되면 Fig. 18과 같이 동작을 한다.

두 번째로 기존 PDF나 인쇄 문서에 문제풀이 페이지가 포함되어 있는 부분을 스크립트 코드와 본문 내용 교체로 자동으로 퀴즈 폼이 생성되고 포함에 대한 구현이다. 퀴즈 폼 변환은 일단 문서 변환이 된 다음 진행이 되는데, 문서의 페이지가 들어오면 먼저 변환을 한 다음 글자들을 검사해서 퀴즈 폼으로 판단되면 퀴즈에 대한 문제에 블록 단위를 확인 한 뒤에 HTML 소스코드로 변환한다. 그 다음에 위치를 계산하여 퀴즈를 배치한다. 이 과정을 반복 한 뒤에 퀴즈가 검출이 되지 않거나 모든 글자가 검사를 마치면 전체적 XHTML 파일 및 CSS 파일을 생성하고 퀴즈에 해당하는 자바스크립트를 삽입한다.

```
FOR inputText = first to last of allText DO
  IF size of input text > size of maximum text THEN
    secondText = maxText
    maxText = inputText
  ELSE IF size of inputText == size of maxText THEN
    IF colorSum() of inputText <
      colorSum() of inputText THEN
      secondText = maxText
      maxText = inputText
    ELSE
      secondText = inputText
  return maxText
```

Fig. 15. Extracting the Title of Pseudo Code

```
.VP_fade_in {
  background: none;
  position: absolute;
  top: 0px;
  left: 0px;
  width: 1684px;
  animation: VP_fade_in 3s 1;
  -o-animation: VP_fade_in 3s 1;
}
@keyframes VP_fade_in
{
  0%{transform: translateX(0); opacity:0}
  25%{-o-transform: translateX(0); opacity:0}
  50%{transform: translateX(0); opacity:0}
  100%{transform: translateX(0); opacity:1}
}
@-o-keyframes VP_fade_in
{
  0%{-o-transform: translateX(0); opacity:0}
  25%{-o-transform: translateX(0); opacity:0}
  50%{-o-transform: translateX(0); opacity:0}
  100%{-o-transform: translateX(0); opacity:1}
}
```

Fig. 16. CSS Code of 'Fade In'

```
<div class="VP_fade_in">
  <div class="vp_textbox text120 fnt10">Social</div>
  <div class="vp_textbox text121 fnt10">media</div>
  <div class="vp_textbox text122 fnt10">networks</div>
  <div class="vp_textbox text123 fnt10"></div>
</div>
```

Fig. 17. Applying the Code in Body

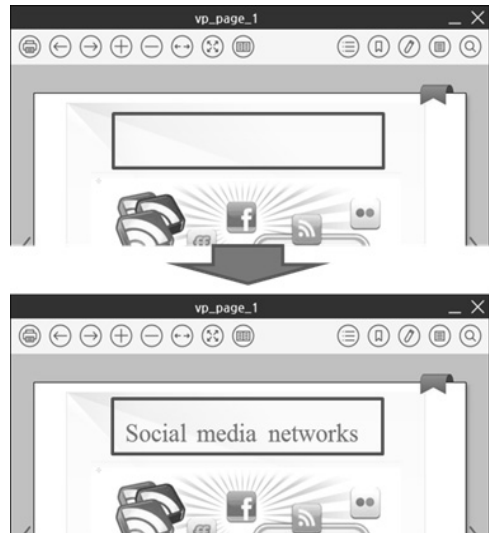


Fig. 18. Applied Content

퀴즈 폼 자동 생성을 이용하고 에디터로 디자인 및 정답 값을 입력해주어야 한다. 약간의 수정과정을 거치면 Fig. 19와 같이 나타난다. 각 퀴즈는 선택이 가능하고 답안 확인 버튼을 누르게 되면 정답의 확인이 즉각적으로 가능하고 다시 풀어볼 기회도 얻을 수 있다. 오답일 경우 정답을 페이지에 띄워 사용자의 확인이 가능하다. 또한 전부 맞출 경우에는 새로운 박스를 띄워 전부 맞았음을 확인할 수 있다. Fig. 20과 같이 인터랙티브 효과를 즉각적으로 확인 가능하다.

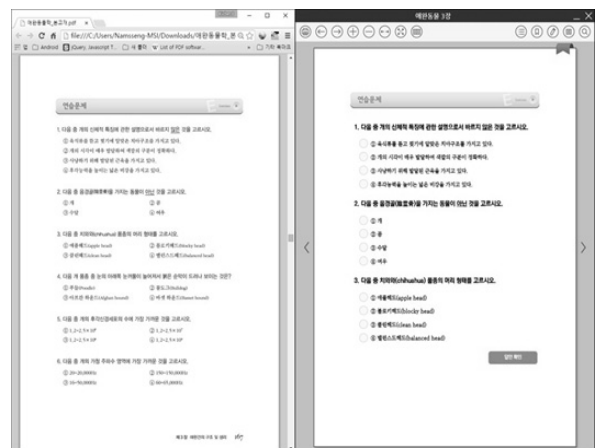


Fig. 19. Quiz Page in PDF (left figure) vs. Quiz Page in EPUB after Conversion (right figure)

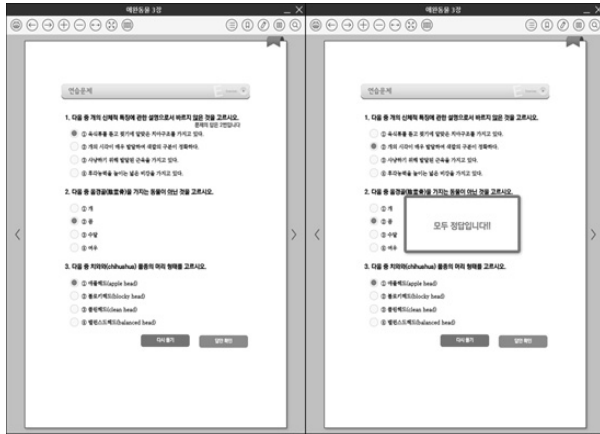


Fig. 20. Case of Incorrect Answer (left figure) vs. Case of Correct Answer (right figure)

6. 변환유효성 및 편의성 평가

본 장에서의 변환유효성 및 편의성 평가는 실제로 PDF에서 EPUB으로 변환하여 전자책을 제작, 수정 등을 하는 실무자의 평가를 통해 변환의 유효성 또는 만족도의 평가 및 개선해야 할 사항을 도출하여 본 연구의 검증 및 변환기법을 향상시키는데 있다[13, 14].

6.1 평가 환경

사용자 평가는 다음과 같이 이루어 졌다.

- 평가대상 : 7년 이상 실무자 1명, 3년 이상 개발자 1명, 1년 이상 실무자·개발자·디자이너 4명
- 평가문항 : 6개 항목
- 평가점수 : 1 - 5점

6.2 평가 결과

Table 4는 PDF에서 EPUB으로 변환에 있어 3가지 기법에 대한 설문 평가 결과를 나타낸다. 각각의 기법에 맞게 문항에 따라서 평균값과 표준편차를 계산하였다. Fig. 21은 세 가지 변환 기법에 따른 Table 4의 질문에 대한 결과를 박스그래프로 나타낸 것이다. 대체적으로 높은 만족도를 보이고 있다. Q.1 질문에 만족도는 4.5에 편차도 0.54로 육안으로 비교 시 만족도는 높은 편이다. Q.2 질문의 레이아웃(layout) 보존에 대한 만족도가 평균적으로 4.167이지만 표준 편차가 높아 향후 추출 기능을 향상시키기 위한 연구가 필요하며, 제목에 대한 인터랙티브 효과의 질문인 Q.3에 평균은 4.5에 표준 편차는 0.54 정도로 알고리즘 적용과 인터랙티브 효과 삽입에 대한 기법연구는 잘 수행되었음을 보인다. 그러나 퀴즈 폼에 대한 프로토타입에 대해서는 이후 수작업으로 보정 작업을 거쳐야하는 부분이 있기에 Q.4의 퀴즈폼 만족도는 제목 부분에 인터랙티브 효과 만족도보다 떨어지며, 전체 인터랙티브의 중요성 질문인 Q.5를 보면 평균

이 4.333이지만 표준 편차가 타 질문 보다 높아 인터랙티브 기능의 중요도는 많은 개인적 견해차가 있음을 나타내고 있다. 사용자 평가에 있어서 전체 점수 평균은 4.333으로 대부분 높은 만족도를 나타내고 있다. 향후 PDF를 EPUB으로 변환 시 필요한 기능들의 구현과 좀 더 다양한 평가항목과 대상에 대한 변환 유효성 및 편의성 평가테스트를 통해 변환기능의 향상뿐만 아니라 변환 시 수작업을 최소화하기 위한 연구를 수행할 것이다.

Table 4. Result of Usability Test for Three Schemes

Questions	Mean	Standard Deviation
Q.1 The CMYK type color of PDF document was well converted to the RGB color of EPUB?	4.500	0.5477
Q.2 The preservation of layout, letters, and image location were extracted properly when transforming PDF document to EPUB?	4.167	0.7528
Q.3 The interactive effect on the caption was well extracted when transforming PDF document to EPUB?	4.500	0.5477
Q.4 How is the transformation to the interactive quiz form of EPUB from the quiz of PDF satisfied?	4.000	0.6325
Q.5 The interactive function in EPUB is necessary?	4.333	0.8165
Q.6 What is the overall satisfaction level of the above functions (Q.1 - Q.5)?	4.500	0.5477
Overall	4.333	0.6408

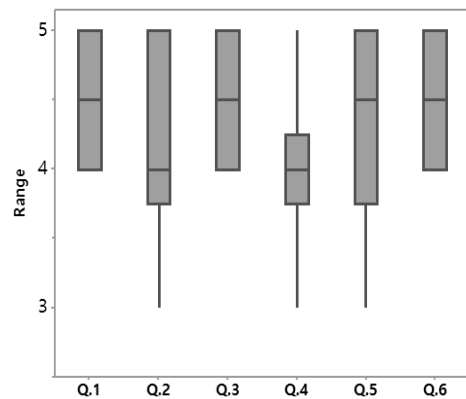


Fig. 21. Box Graph about Result of Usability Test

7. 결론

본 연구는 PDF문서를 EPUB 형태로 변환 시 사람의 수작업에 의한 보정 없이 바로 볼 수 있도록 하는 연구에 목적을 두었다. (1) 인쇄 색상 표현 기준인 CMYK를 웹 표준

색상 표현 방식인 RGB 색상으로 변환 시 ICC 프로파일을 이용하여 색감을 보존하기 위한 연구를 수행하였고, (2) PDF와 EPUB의 구조적, 파일 배치 부분에 차이를 비교하여 PDF에서 EPUB으로 변환 시 고정 레이아웃 형태로 형식에 벗어나지 않게 구현 하였으며, (3) 제목 부분의 추출 알고리즘을 통한 상호작용 효과 구현과 문서에 퀴즈 형식의 내용을 추출하여 사용자가 참여할 수 있도록 퀴즈 폼으로의 변환 및 프로토타입 구현을 하였다.

향후, 본 연구를 진행하면서 기존 문서들을 디지털화 하기 위한 다음과 같은 연구를 진행할 것이다. 첫 번째는 기존의 책은 크기가 정해져 있지만 각종 장치들은 크기와 형태가 다양하다. 따라서 다양한 기기들에 맞추기 위한 자동 공간조정(Reflowable) 형태로 변환하기 위한 연구를 수행할 것이고, 두 번째는 글자나 이미지의 인터랙티브 효과뿐만 아니라 상황이나 날씨, 지역에 따른 내용, 이미지 바꿈 효과를 넣어 더 다양한 형태로 변환 및 생성하기 위한 연구를 수행할 것이다.

References

[1] Hyun-suk Moon, "The Production of Interactive App Books using Digital Publishing Technique," *Journal of Digital Design*, Vol.13, No.2, pp.441-449, 2013.

[2] EPUB: IDPF (International Digital Publishing Forum) Standard [Internet], <http://idpf.org/epub>.

[3] Joon-Woo Park, Jong-Wook Won, "A Study on Interactive App Book Design for the Increase of User Pleasurability," *Journal of Digital Design*, Vol.14, No.1, pp.141-150, 2014.

[4] Adobe Systems Incorporated, PDF Reference Sixth Edition, Adobe Systems Incorporated, CA, pp.90-98, 2006.

[5] Eui Seok, Chung, "EPUB 3.0," *TTA Journal*, Vol.144, pp.55-58, 2012.

[6] Orange Digit Inc., Viewporter[®] Converter 2015 [Internet], <http://viewporter.com/viewporter-converter/>.

[7] Ford, Adrian and Alan Roberts, "Colour Space Conversions," Westminster University, London, pp.1-31, 1998.

[8] Accessing Google Maps from EPUB, [Internet], <http://idpf.org/forum/topic-533>.

[9] J. Bidarra, M. Figueiredo and C. Natálio, "Interactive Design and Gamification of ebooks for Mobile and Contextual Learning," *International Journal of Interactive Mobile Technology*, Vol.9, No.3, pp.24-32, 2015.

[10] The Apache Software Foundation, Apache PDFBox - A Java PDF Library 2015 [Internet], <https://pdfbox.apache.org/>.

[11] Kyung-Chul Song and Sang-Hoon Kang, "A Study on the Color Proofing System Development for High Quality Color Prints," *Journal of the Korean Printing Society*, Vol.22, No.2, pp.55-72, 2004.

[12] Gwang-Gook Lee et al., "Comparisons of Color Spaces for Shadow Elimination," *Journal of Korea Multimedia Society*, Vol.11, No.5, pp.610-622, 2008.

[13] Sung-Sik Lee, "A Study on Design Concept of Computer OS Interface as the Agent on Ubiquitous Computing Environment," *Journal of Korean Society of Design Science*, Vol.20, No.4, pp.191-202, 2007.

[14] Sung Jung Cho and Yeun Bae Kim, "Intelligent Mobile Human-Computer Interaction Based on Machine Learning Algorithms," *Korean Institute of Information Scientists and Engineers*, Vol.25, No.3, pp.46-51, 2007.



이 남 희

e-mail : namsseng@nate.com
 2011년 한성대학교 멀티미디어공학과(학사)
 2016년 아주대학교 지식정보공학과(석사)
 현 재 (주)포도소프트 연구원
 관심분야: IoT, 클라우드 컴퓨팅, 모바일보안



김 재 훈

e-mail : jaikim@ajou.ac.kr
 1997년 Texas A&M University 컴퓨터공학(박사)
 1998년~현 재 아주대학교 사이버보안학과 교수
 관심분야: 분산시스템, Cyber-Physical 시스템, 모바일 컴퓨팅



김 강 석

e-mail : kangskim@ajou.ac.kr
 2007년 인디애나대학교 컴퓨터공학(박사)
 2010년~2016년 아주대학교 지식정보공학과 연구교수
 2016년~현 재 아주대학교 사이버보안학과 부교수
 관심분야: 클라우드 컴퓨팅, 유비쿼터스 컴퓨팅, 모바일 보안, IoT 협업시스템