

부채꼴 모델에 기반한 CSS3 극좌표계 서식 모듈의 설계

심승민*, 임순범**

Design of CSS3 Polar-Coordinate Layout Module based on Fan Model

Seung-Min Shim*, Soon-Bum Lim**

ABSTRACT

Most web documents are written in Cartesian coordinates, so the study of vertical arrangement of text has been well organized, while the study of circular arrangement is very early. With the recent development of circular display devices, the demand for circular arrangement of texts is increasing. Thus, we proposed a CSS3 extended specification of polar coordinate layout for the circular placement of text. First, we defined the concept of fan model for the text arrangement in polar coordinate which is corresponding to box model in Cartesian coordinate. And, we described new definition on the directions of sentence, paragraph and text orientation in polar coordinate. Based on this new definitions, we developed the extended specification consisting of three parts. A part for setting the fan model, a part for setting directions, and a part for setting typesetting properties. To verify the feasibility of the proposed specification in current web browsers, a preprocessor was developed and sample contents were examined. We compared the code length of the sample contents implemented using other JavaScript library CssWarp.js so as to verify the efficiency of the proposed specification.

Key words: CSS3, Polar Coordinate, Fan Model, Box Model, Sentence Direction, Paragraph Direction, Text Orientation

1. 서 론

주로 인터넷 브라우저에서 사용되고 있는 웹 문서에서 텍스트나 그림은 HTML로 정의되고 이들 객체의 공간적인 배치는 CSS(Cascading Style Sheet) 형식의 스타일시트를 통하여 이루어진다. 현재 사용하고 있는 CSS3는 여러 개의 모듈로 구성되어 있어서 매우 다양한 새로운 기능을 추가하는 작업을 용이하게 할 수 있다[1]. CSS3 모듈에는 기본설정에 대한 문법, 텍스트 및 폰트 속성 설정, 텍스트의 다양한

배치, 디스플레이 기기의 속성, 다양한 시각적 효과 등 수십 개로 구성되어 있다.

대부분의 CSS3 모듈은 직각 방향의 사각형 디스플레이 공간에서 텍스트 및 그림 객체를 배치하고 다양한 애니메이션 및 전환 효과를 적용한다는 가정 하에 사양이 개발되어 왔다. CSS를 이용하여 수직방향으로 텍스트를 배치하기 위하여 Erika J. Etemad는 서로 다른 편집 방식을 적용하는 다양한 언어의 문자 배열 방식 및 오리엔테이션 설정 방법을 체계적으로 분석하였다. 그 연구결과로 텍스트의 공간 배치

* Corresponding Author: Soon-Bum Lim, Address: (04310) Sookmyung Women's University, Cheongpa-ro 47-gil 100, Yongsan-gu, Seoul, Korea, TEL: +82-2-710-9424, E-mail: sblim@sookmyung.ac.kr
Receipt date: Jan. 4, 2018, Approval date: Jan. 18, 2019
* Dept. of IT Engineering, Sookmyung Women's University (E-mail: 511mintlab@gmail.com)

** Dept. of IT Engineering and Research Institute of ICT Convergence, Sookmyung Women's University
* This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2016-0-00166, Technology Development Project for Information, Communication, and Broadcast)

의 기본 개념이 되는 박스 모델을 제시하였고[2], 이 개념을 중심으로 CSS3 표준 사양에서 텍스트 배치의 기본이 되는 CSS3 Writing-mode 표준을 개발하였다[3].

텍스트의 수직 배치에 대한 연구와 사양은 충분히 발전되어 왔지만 원형으로 배치하는 연구나 사양은 매우 초기 단계이다. 최근 들어서 원형 디스플레이 기기의 출현에 따라 웹기반 플랫폼에서 원형 디스플레이 기기를 지원하기 위한 사양이 제안되었으나 간단히 원형 테두리에 텍스트를 배치하는 경우만 포함하고 일반 텍스트를 원형의 극좌표계에 자유로이 배치하여 문단을 편집하는 기능까지는 지원하지 않고 있다.

본 연구의 최종 목표는 웹 표준 문서에서 스타일 시트를 사용하여 텍스트를 원형으로 극좌표계(Polar Coordinate System)에 배치하는 것을 용이하게 하도록 CSS3 사양을 확장하여 문서작성에 적용하는 것으로 하였다. 선행 논문[4,5]에서 기존 원형 텍스트 배치 사례들로부터 극좌표계 텍스트 배치에 필요한 요구사항을 분석하였고, 본 논문에서는 기존 연구[6]의 결과 중에서 CSS3 극좌표계 서식 모듈의 확장 기능에 대한 개념과 구체 사양을 상세히 소개한다.

극좌표계에 텍스트를 다양한 속성으로 배치할 수 있는 CSS를 설계하기 위하여 3절에서 CSS3 부채꼴 모델을 제안하고, 4절에서 이 모델을 기반으로 극좌표계에 텍스트를 자유로이 배치하는 CSS3 확장사양을 제시한다. 5절에서는 본 논문에서 제안하는 CSS3 확장사양이 타당한지 실험한 결과를 보여준다.

2. CSS3 사양 확장 관련 기존 연구

본 절에서는 CSS 사양의 개발 과정에서 직교좌표계의 CSS 텍스트 배치 사양의 근간이 된 연구와 텍스트 관련 CSS3 모듈을 소개한다. 그리고 원형 디스플레이를 위한 CSS3 확장 사양과 기존의 원형 텍스트 배치를 위한 자바스크립트 라이브러리를 소개한다.

2.1 Robust Vertical Text Layout

CSS3 Writing-mode Module[3]이 나오기 전에 CSS에서 텍스트 배치 사양을 정리하기 위하여 Erika J. Etemad가 연구 결과를 “Robust Vertical Text Layout”[2]에 발표하였다. 텍스트를 수직 방향으로

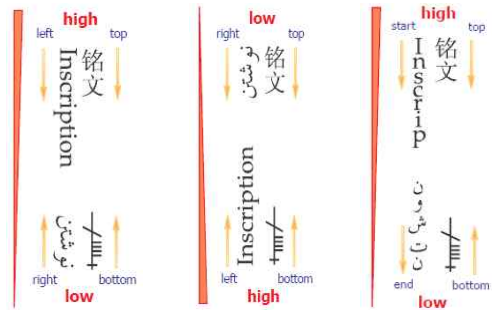


Fig. 1. Text orientations in each sentence directions [2].

배치하는 경우에 서로 다른 편집 방식을 적용하는 각 언어의 문자들이 섞여있는 문서를 편집하기 위한 일반적인 조판 방식을 정리하고, 이에 적용할 수 있는 박스 모델을 제시하였다.

Fig. 1에서 보듯이 아랍어는 텍스트를 아래에서 위로 배치하기도 하며, 한문은 위에서 아래로 배치한다. CSS Writing-mode Module이 나오기 전에는 수직으로 텍스트를 배치하는 시스템이 없었기 때문에 배열 방향이 다른 문자가 등장할 때마다 문자의 방향을 직접 변경해야 했다. 이 연구는 이런 문제점을 해결하기 위하여 문자 고유의 내재된 특성을 정리하여 자동으로 여러 방향의 문자가 섞인 텍스트를 수직으로 배치할 수 있는 시스템을 설계하도록 하였다.

2.2 텍스트 관련 CSS3 모듈

모듈 개념을 도입한 CSS3 사양은 각종 출력 디바이스의 요구에 따라서 필요한 CSS3 모듈만을 탑재하여 실행하거나 해당 모듈만 신속하게 업데이트할 수 있게 되어 있다. 또한 모듈 단위의 개발은 스타일 시트 사양 개발을 용이하게 해 준다. W3C의 스타일 시트 홈페이지에서 CSS3를 구성하는 모듈의 종류와 현재 개발 상황을 확인할 수 있으며, 이 중에서 본 연구와 관련된 CSS3 모듈인 CSS3 Text Module Level 3, CSS3 Writing Mode Level 3에 대해 소개한다.

CSS Text Module Level 3[7]은 CSS에서의 텍스트 조작 방법에 대해 정의하는 모듈로, 텍스트의 서식을 설정하는 데 필요한 속성들을 제공한다. 여기서는 줄바꿈, 정렬, 공백 처리, 글자 변형 등을 다루고 있다.

CSS Writing-mode Mode Level 3[3]은 다양한 국가들의 서로 다른 텍스트 입력 방식을 지원하기 위한

모듈이다. 예를 들면 라틴어나 인디어나 같이 왼쪽에서 오른쪽으로 흐르는 텍스트, 히브리어, 아랍어와 같이 오른쪽에서 왼쪽으로 진행되는 텍스트, 그리고 아랍어같이 가로쓰기와 세로쓰기가 혼용된 경우와 아시아권 언어와 같이 세로쓰기 등이 있다.

2.3 CSS Round Display

CSS 사양을 원형 기기에 적용하기 위해 최초로 시도한 연구로는 LG전자에서 제안한 “CSS Round Display Level 1”[8]이 있다. 최근 스마트폰 등의 첨단 기기에서 사용되는 원형 디스플레이 화면의 기능을 지원하기 위하여 제안하였다. Fig. 2와 같이 텍스트나 짧은 문장의 콘텐츠를 원형 공간에 맞도록 배치하고 원형 테두리를 그려줄 수도 있다. 그러나 긴 문장이나 여러 줄로 구성되는 문단을 원형 경로에 따라 각 글자를 배치하는 기능은 포함하고 있지 않다.

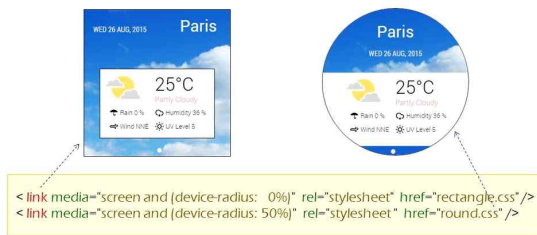


Fig. 2. Example of CSS round display.

2.4 자바스크립트 라이브러리

기존의 CSS나 벡터 그래픽을 위한 SVG(Scalable Vector Graphics) 같은 웹 표준을 이용하여 곡선 경로에 텍스트를 배치하기도 하는데, 텍스트를 일일이 분리하여 위치와 각도를 계산해야 하므로 결과를 얻는데 매우 번거롭게 된다. 이에 곡선이나 원호를 따라 텍스트를 배치하는 ArcText.js, CssWarp.js 등의 자바스크립트 라이브러리가 제공되고 있다.

ArcText.js[9]는 이미지 도안이나 로고 등에서 사용되는 간단한 텍스트를 원형 경로에 배치하는 기능을 제공한다. CssWarp.js[10]는 조사한 극좌표계 텍스트 배치 자바스크립트 라이브러리 중 가장 많은 설정 사양을 제공한다. 이 중, 원에 관련된 속성을 설정하는 사양으로 radius(반지름), center(중심점), angle(시작각도), align(정렬), textPosition(문장 위치)이 있고, 부수적으로 rotationMode(글자 오리엔테이

션 회전여부), indent(들여쓰기)도 설정할 수 있다.

3. 부채꼴 모델과 텍스트 방향성 정의

선행 논문[5]에서는 텍스트를 원형으로 배치하기 위한 CSS3 요구사항을 정리하였고, 본 절에서는 이들 요구사항을 분석한 결과를 바탕으로 텍스트를 극좌표계에 배치하는데 기본적으로 필요한 개념인 부채꼴 모델을 정의하고 텍스트 배치에 필요한 속성을 모델링하였다. 개념 모델은 2016년 선행 연구[4]에서 제안한 바 있는데, 본 논문에서는 이전 부채꼴 모델에 대한 정의에서 미비한 점을 수정 보완하였다.

3.1 부채꼴 모델(Fan Model)

현재 HTML 문서에 적용하는 기존의 CSS는 직교좌표계를 기본으로 하고 있으며, 텍스트나 그림 등 모든 HTML 요소들은 직교좌표계 내에서 사각형 영역에 배치된다. 이 때 각 요소가 차지하는 사각형 영역을 설정하기 위한 속성과 개념을 박스 모델(Box Model)이라고 명명되어 있다. CSS 박스 모델의 영역은 크기가 다른 여러 개의 중첩된 사각형으로 구성되며, Table 1에서 보듯이 콘텐츠 요소가 차지하는 콘텐츠 영역부터 시작하여 박스의 내부 영역은 padding, 박스의 테두리가 차지하는 영역은 border, 박스의 외부 영역은 margin이라는 속성 이름으로 정의되어 있다. 콘텐츠 영역의 크기는 width와 height 속성으로 지정할 수 있으며, 위치는 position 속성으로 위

Table 1. The concept of CSS box model

display type	block, inline, inline-block, etc.
area	width, height
position	position attributes are used to select positioning method, and the box is placed at the left-top corner.
basic concept	

치지정 방식을 선택한 후 전체 박스 영역의 왼쪽 꼭대기(left-top) 점을 기준으로 배치한다. 표현하고자 하는 요소가 문장 내에 위치하는지 독립된 공간을 차지하는지는 display 속성에서 inline 혹은 block 이라는 값으로 표기한다.

선행 연구 및 본 논문에서 제안하는 CSS 부채꼴 모델(CSS Fan Model)은 HTML 구성요소의 원형 배치를 위하여 극좌표계(Polar Coordinate System)에서 콘텐츠를 배치하고 그들이 차지하는 영역을 설정하기 위한 개념 모델이다. Table 2에서 보듯이 극좌표계에서 콘텐츠의 영역은 사각형 가로/세로의 폭(width)/높이(height) 대신에 부채꼴의 각도(fan-angle)/지름크기(fan-radius)로 정의되며, 이 개념에 맞추어 부채꼴 내부영역, 경계선 영역, 외부영역은 박스 모델에 상응하게 존재한다. 기존 박스 모델에서 사각형 영역을 지정할 때에는 left, right, top, bottom의 속성을 이용하였지만 부채꼴 모델에서는 Table 2에서 보듯이 극좌표계의 부채꼴 모양에 맞도록 left, right 대신 start-edge, end-edge로 top, bottom 대신 out-edge, in-edge로 정의하도록 하였다. 박스 모델은 사각형 모양이므로 좌상단점(left-top) 기준으로 위치를 지정하였지만, 부채꼴 모델은 부채꼴의 중심점(fan-center)을 기준 위치로 지정하여 배치한다. display 속성을 지정하는 데에는 박스 모델에서 사용되는 block 유형과 inline 유형에 상응하도록 부채꼴 모델에서는 fan-block 유형과 fan-inline 유형을 설정하도록 하였다.

Table 2. The concept of Fan model

display type	fan-block, fan-inline
area	fan-angle, fan-radius
position	position attributes are used to select positioning method, and the fan is placed at the fan-center point.
basic concept	

3.2 극좌표계에서 수평과 수직

직교좌표계의 수직과 수평 개념이 극좌표계에서 그대로 적용되지 않으므로 수직과 수평의 개념을 새로이 정의해야 한다. CSS3 Writing-mode 사양에서는 텍스트의 방향을 가로로 진행되는 수평(horizontal) 텍스트와 세로로 진행되는 수직(vertical) 텍스트로 구분한다. 극좌표에서 수평과 수직은 직교좌표계에서 보는 것과는 많이 다르다. 예를 들어, Table 3에서 (b)Horizontal-2의 사례는 직교좌표계 관점에서 보면 수직 텍스트인 것처럼 보이며, (d)의 사례에서도 일부 수직 텍스트는 수평인 것처럼 보인다. 이와 같은 혼돈을 줄이기 위하여 극좌표계에서 수평 방향과 수직 방향에 대한 개념을 새롭게 정의하였다. 본 연구에서는 극좌표계의 수평 방향은 원호를 따라 진행되는 방향이라고 정의하였고, 속성값은 cw(시계 방향) 혹은 ccw(반시계 방향)을 가질 수 있다. 극좌표계의 수직 방향은 중심점과 원호 사이를 진행하는 방향이고, 속성값으로 inward(바깥에서 안쪽으로) 혹은 outward(안에서 바깥쪽으로)을 가질 수 있는 것으로 정의하였다.

3.3 극좌표계에서 가로쓰기와 세로쓰기

CSS3 사양의 Writing-mode Module에서 정의되는 텍스트의 방향에는 Table 4에서 보듯이 문장에서

Table 3. Definition of horizontal and vertical directions

(a) Horizontal direction in polar coordinate	
definition	direction along the arc
values	cw, ccw
examples	
(b) Vertical direction in polar coordinate	
definition	direction between the arc and the center
values	inward, outward
examples	

Table 4. Directions in Cartesian coordinate system

directions	ltr, rtl, tb, bt
(a) horizontal writing	
(b) vertical writing	
Note :	

글자가 진행되는 문장의 방향(inline direction), 여러 개의 글줄이 진행되는 문단의 방향(block direction), 그리고 각 글자가 서있는 방향인 글자의 방향(glyph orientation) 세 가지로 구분된다. 이들 세 가지 종류의 방향을 지정하는 데에는 ltr(왼쪽에서 오른쪽으로), rtl(오른쪽에서 왼쪽으로), tb(위에서 아래로), bt(아래에서 위로) 진행되는 값을 가질 수 있다.

극좌표계에서는 이와는 다르게 문장과 문단의 방향으로 적용할 수 있는 값으로는 Table 3에 소개하였던 cw, ccw, inward, outward 방향이 있으며, 글자 방향을 위해서는 upright(문장의 직각 방향)과 sideways(문장의 진행 방향)를 정의하였다. 직교좌표계에서는 글자의 오리엔테이션이 문장의 진행과 직각 혹은 반대 방향이라도 ltr, rtl, tb, bt의 네 가지 방향으로 지정할 수 있지만 극좌표계에서는 Table 5의 사례에서 보듯이 방향 지정에 혼란이 올 수 있다. 이에 따라 수직 방향으로 글자가 서있는 방향을 upright, 수평 방향으로 글자가 누워있는 방향을 sideways라고 정의하였다. 극좌표계에서 새롭게 정의된 방향을 적용하여 가로쓰기와 세로쓰기의 개념을 체계적으로 분류하여 Table 5와 같이 극좌표계의 방향성을 정리하였다.

4. CSS3 극좌표계 서식모듈의 설계

이 절에서는 앞에서 정의한 부채꼴 모델 개념을 스타일시트에서 사용할 수 있도록 설계된 CSS3 극

Table 5. Directions in Polar coordinate system

directions	cw, ccw, inward, outward, upright, sideways
(a) horizontal writing with upright orientation	
(b) horizontal writing with sideways orientation	
(c) vertical writing with upright orientation	
(d) vertical writing with sideways orientation	
Note :	

좌표계 서식모델의 확장사양을 설명한다. CSS3 극좌표계 서식모델의 확장 사양은 부채꼴 모델을 설정하는 속성, 극좌표계의 방향성을 설정하는 속성, 극좌표계 텍스트의 인쇄조판을 설정하는 속성의 세 부분으로 구성된다.

4.1 부채꼴 모델 설정을 위한 CSS3 확장사양

부채꼴 모델을 설정하는 기본 속성으로는 디스플레이 표시형태, 부채꼴의 각도인 너비, 반지름인 높이, 그리고 부채꼴 위치의 기준점을 설정하는 사양이 있다. Table 6에서는 디스플레이 모델 설정에 관련된 기존 CSS 사양과 본 연구의 확장사양을 비교하여 보여준다. CSS3 확장사양에서 확장속성을 구분하기 위해 '-fan-'이라는 접두사를 이용하였다.

Table 6. CSS previous and extended specifications for setting the display model

setting	Related specifications in CSS3 box mode	Extended specifications for CSS3 fan model	default
display type	display : block inline inline-block	display : block inline inline-block -fan-block -fan-inline	필수
width	width : <length>	-fan-angle : <angle> auto	auto
height	height : <length>	-fan-radius : <number> auto	auto
reference point	position : absolute relative static fixed, left : <length>, top : <length>	position : absolute relative static fixed, left : <length>, top : <length>	static

4.1.1 표시 형태 선택 : display

기본적으로 텍스트를 극좌표계로 배치하려면, HTML 기본 요소의 표시 형태를 극좌표계 형태로 바꿀 수 있는 사양이 필요하다. HTML의 모든 요소는 각각 기본 출력 형식이 정해져있는데, 예를 들어 요소는 inline 형식이고, <p> 요소는 block 형식이다. 이 때 사용자는 HTML 기본 출력 형식을 변경하여 다양한 화면을 구현할 수 있는데, 이를 지원하는 속성이 display 속성이다. 따라서 본 연구에서는 display 속성에 속성값을 추가하여 사용자가 출력 형식을 극좌표계 형식으로 바꿀 수 있도록 하였다.

display값은 요소의 표시 형태를 바꾸기 때문에 이 값에 따라 다른 사양 값에도 영향을 주게 된다. 요소의 컨테이너 크기를 지정하는 width나 height 값이 대표적인데, inline 요소는 자기 자신의 컨테이너를 가지지 않기 때문에 부모 요소의 컨테이너를 따르고, block 요소는 자기 자신의 컨테이너를 가질 수 있기 때문에 사용자 설정 값을 사용하기 때문이다. 따라서 inline 요소에 width나 height 값을 지정해도 적용되지 않으며, 대신 상위 노드의 컨테이너 값이 적용된다. Fig. 3을 보면, 기존 직교좌표계의 inline 요소는 기본적으로 컨테이너의 기준이 상위 노드이며, 상위 노드가 없으면 브라우저의 크기를 기준으로 한다. 반면, block 요소는 자기 자신의 컨테이너를 가질 수 있기 때문에 사용자가 설정한 값을 기준으로 하고, 사용자 설정 값이 없을 때만 상위노드의 컨테이너를 기준으로 한다.



Fig. 3. The case of inline and block for display attribute.

이에 상응하도록 극좌표계에서도 -fan-inline 요소인 경우, 무조건 상위노드의 컨테이너 크기를 따르도록 설계하였다. 가로쓰기일 때 -fan-inline 요소의 너비 기준은 상위 노드의 -fan-angle(a_2)값이고, 상위노드가 없는 경우 원의 각도 최대치인 360°를 사용한다. 세로쓰기일 때 높이 기준은 역시 상위노드의 반지름 값인 -fan-radius를 사용하고, 상위노드가 없으면 -fan-radius 값이 auto로 지정되었을 때의 자동 설정 반지름을 기준으로 한다.

4.1.2 너비 지정 : -fan-angle

직교좌표계에서 가로쓰기의 문장 너비와 세로쓰기의 문단 너비를 결정하는 속성은 width이다. 극좌표계에서는 문장이나 문단의 너비를 각도가 결정한다. 이에 따라 극좌표계에서 요소의 너비를 설정할 수 있도록 기존 width 사양에 부합하는 -fan-angle 사양을 정의하였다.

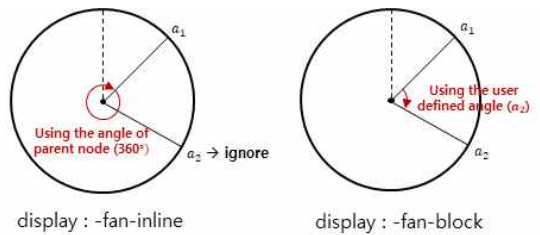


Fig. 4. The base of angles in case of -fan-inline and -fan-block.

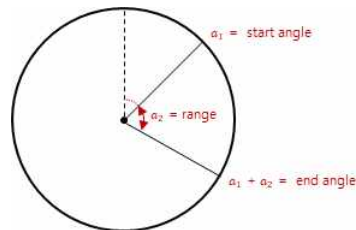


Fig. 5. The value of -fan-angle.

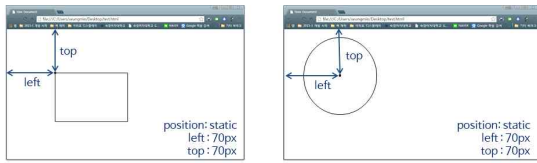


Fig. 6. Positioning in each coordinate system.

CSS Values and Units Module[11]에 따르면, CSS의 각도는 Fig. 5에서 보듯이 12시 방향 축을 기준으로 시계 방향(+)으로 진행되므로, `-fan-angle` 사양도 12시 방향을 0°로 지정하여, 양수이면 시계 방향으로 음수이면 반시계 방향으로 설계하였다. `<angle>`은 각도를 지정하는데, 시작각도만 지정하거나 시작각도와 끝각도를 모두 지정하는 방식이 있다. `auto`로 지정할 경우, 각도는 `display` 값에 따라 다르게 지정이 되는데, 기본적으로 (0deg, 360deg)로 설정된다. `width` 사양은 기존에 `inline` 요소를 제외한 모든 요소에 적용되었으므로, 이에 부합하도록 `-fan-angle`도 `-fan-block` 요소에만 적용된다.

4.1.3 높이 지정 : `-fan-radius`

직교좌표계 요소의 높이를 결정하는 속성은 `height`인 반면, 극좌표계에서 요소의 높이는 반지름이 결정한다. 따라서 극좌표계 요소의 반지름을 설정할 수 있는 `-fan-radius` 사양을 설계하였으며, 블록 요소에만 지정이 가능하다. `-fan-radius` 사양의 `<number>`는 반지름의 길이를 지정하고, `auto`값은 반지름을 자동 지정하는데, 모든 텍스트가 배치될 수 있는 최소한의 반지름으로 설정된다. 최소값은 텍스트의 높이이며, 글줄이 추가될수록 값이 추가되며, 최대값은 원의 반지름이 된다.

4.1.4 기준점 지정 : `position`

극좌표계 요소를 화면에 배치하기 위해서 중심점의 값을 설정할 필요가 있다. 그런데 극좌표계 요소의 배치는 직교좌표 내에서 이루어지므로, 기존의 `position` 속성을 그대로 사용하되, Fig. 6에서와 같이 배치의 기준이 박스 모델의 좌상단(left-top)점이 아닌 중심점(center)이 되도록 설계하였다.

CSS Positioned Layout Module[12]에 따르면, `position`값이 `static`인 경우, 기존의 흐름에 따라 요소를 배치하므로, 직전 요소가 끝나는 부분에 중심점을 배치하게 된다. `relative`로 설정되면, 정상적인 흐름에서 배치될 자신의 위치를 기준으로, 지정한 오프셋(left 또는 top 등)만큼 상대적으로 배치하게 된다. `absolute`를 사용하면 부모 요소의 위치를 기준으로 오프셋만큼 상대적으로 배치되며, `fixed`는 브라우저 윈도우를 기준으로 지정한 오프셋만큼 거리에 배치한다.

4.2 극좌표계 방향 설정을 위한 CSS3 확장사양

극좌표계에서 텍스트의 방향을 설정하기 위한 사양으로는 Table 7에 열거되어 있는 문장 진행 방향, 문단 진행 방향, 글자 오리엔테이션 방향 설정 속성이 있다.

4.2.1 문장 진행 방향 : `-fan-direction`

극좌표계에서 문장의 진행 방향을 지정하기 위해 `-fan-direction` 사양을 설계하였다. 직교좌표계에서 세로쓰기의 문장 방향은 Fig. 7에서 보는 바와 같이, 무조건 위에서 아래로 흐르기 때문에 기존 `direction` 사양은 가로쓰기에만 해당이 된다. 즉, `ltr`, `rtl` 값만 존재한다. 극좌표계에서 세로쓰기의 `tb`은 밖에서 안쪽으로 진행될 수도 있고, 안에서 바깥쪽으로 진행될 수도 있다. 따라서 극좌표계에서는 세로쓰기도 고유의 문장 방향을 가질 수 있으므로, `ltr`에 상응하는 `cw`

Table 7. CSS previous and extended specifications for setting the directions

directions	Related specifications in CSS3 box model	Extended specifications for CSS3 fan model	default
sentence	<code>direction : ltr rtl</code>	<code>-fan-direction : v-inward v-outward cw ccw</code>	cw
paragraph	<code>writing-mode : horizontal-tb vertical-rl vertical-lr</code>	<code>-fan-writing-mode : v-cw v-ccw horizontal-auto</code>	horizontal-auto
text orientation	<code>text-orientation : mixed upright sideways-right sideways-left sideways use-glyph-orientation</code>	<code>-fan-text-orientation : upright sideways sideways-cw sideways-ccw</code>	outward

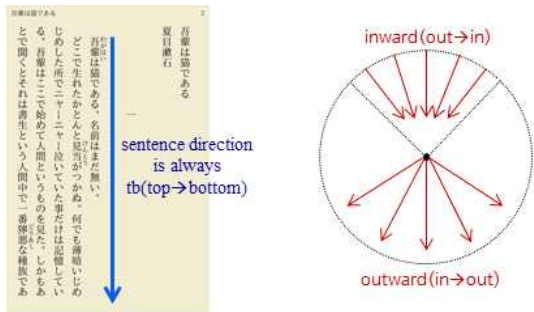


Fig. 7. Sentence directions in vertical writing.

와 rtl에 상응하는 ccw값 외에도, v-inward, v-outward 속성값을 추가하였다.

4.2.2 문단 진행 방향: -fan-writing-mode

-fan-writing-mode 사양은 극좌표계의 문단 진행 방향을 지정할 수 있는데, 가로쓰기일 때 문단은 무조건 위에서 아래로 진행되기 때문에 따로 설정하지 않고, horizontal-auto값을 자동 지정되는 방식으로 설계하였다. horizontal-auto 값으로 설정하면, Fig. 8에서와 같이 문장 방향이 시계 방향(cw)인 경우 문단은 tb으로 진행되기 위해 밖에서 안쪽(inward)으로 배치되고, 반시계 방향(ccw)인 경우, 문단은 안에서 바깥(outward)으로 진행된다.

4.2.3 글자 오리엔테이션 방향: -fan-text-orientation

모든 글자 체계는 언어권에 따라 고유한 글자 오리엔테이션을 가진다. 현대의 글자 체계는 크게 세 가지로 나뉘는데, 예를 들어 라틴어나 아라비아어 등은 수평 텍스트만 사용하며, 몽골어 등은 수직 텍스트만 쓰고, 한글이나 한자, 일어 등은 수평 텍스트와

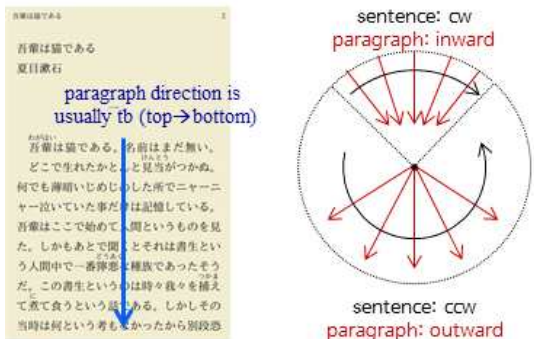


Fig. 8. Paragraph directions in horizontal writing.

수직 텍스트를 모두 사용한다. 수평 텍스트만 사용하는 언어는 세로쓰기를 할 때, 글자 오리엔테이션도 함께 회전되므로, text-orientation 사양을 통해 오리엔테이션 방향을 바꿀 수 있다. 직교좌표계에서 text-orientation 사양의 기본값은 mixed로, 수평 텍스트와 수직 텍스트가 함께 쓰이는 경우, 언어마다 다른 글자 오리엔테이션을 그대로 적용한다. upright 값은 Fig. 9에서처럼 언어에 상관없이 글자가 윗(up) 방향을 가리키게 한다.

극좌표계에서는 upright 값을 기본값으로 하며, 직교좌표계와 달리 극좌표계에서 upright의 방향은 중심점을 기준으로 밖에서 안쪽 방향 또는 안에서 바깥쪽 방향을 의미한다. Fig. 10에서 (b)는 가로쓰기에서 upright 방향을, (c)에서는 세로쓰기에서 upright 방향을 보여주고 있다. sideways-cw는 글자 오리엔테이션을 시계 방향으로 회전시키고, sideways-ccw는 반시계 방향으로 회전시키며, sideways 값은 가로쓰기의 경우 문장 방향(direction)과 같은 방향으로 회전시키고 세로쓰기의 경우 문장 방향과 직각으로 글자를 회전시킨다.

4.3 극좌표계 인쇄조판 설정을 위한 CSS3 사양 확장

인쇄조판 속성 설정에 해당하는 기능에는 Table 8에 열거된 들여쓰기, 가로정렬, 공백표시, 세로정렬을 지정하는 사양이 있다.



Fig. 9. Text orientations for vertical writing in Cartesian coordinate system [2]

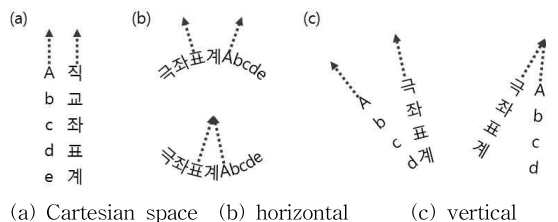


Fig. 10. Upright text orientations.

Table 8. CSS previous and extended specifications for typesetting attributes

setting	Related specifications in CSS3 box model	Extended specifications for CSS3 fan model	default
indentation	text-indent : <length> <percentage>	-fan-indent : <length>	0
alignment	text-align : start end left right center	-fan-align : ceeter start-edge end-edge	left
white spacing	white-space : normal pre nowrap pre-line pre-wrap	-fan-white-space : normal pre-line pre-wrap	normal
vertical alignment	vertical-align : <length> <percentage> baseline text-top text-bottom sub super middle top bottom	-fan-vertical-align : <length> <percentage> baseline text-top text-bottom sub super	baseline

4.3.1 들여쓰기 : -fan-indent

요소의 첫 번째 행의 들여쓰기를 지정하는 사양으로 -fan-indent 사양을 추가하였다. 길이를 지정하는 방식을 사용하는데, 기존의 text-indent 사양에서는 속성값을 음수로 지정하면 내어쓰기 형태가 되기 때문에, 이에 부합하도록 -fan-indent도 음수로 지정할 경우 내어쓰기가 되도록 설계하였다.

4.3.2 가로정렬 : -fan-align

-fan-align 사양은 극좌표 요소 내 텍스트의 수평 정렬을 지정한다. 글자의 수평 정렬을 지정할 때 사용하는 text-align을 참조하여 설계하였는데, 기존 text-align은 left, right 등의 값을 쓰는 반면, 부채꼴 모델에서 left는 start-edge, right은 end-edge의 용어를 쓰므로, left, right 대신 start-edge, end-edge의 값을 사용하도록 하였으며, center는 그대로이다. Fig. 11과 같이 start-edge 정렬인 경우, 시작각도를 기준으로 정렬되며, end-edge 정렬인 경우, 끝각도를 기준으로 정렬된다. center값이면 가운데를 기준으로 정렬된다.

4.4.3 공백표시 : -fan-white-space

사용자가 HTML 요소 내에 입력한 공백을 표시할 수 있는 사양이 white-space이다. 극좌표계 텍스트 배치에서도 공백을 처리할 수 있도록 -fan-white-space 사양을 설계하였다. 기존의 white-space 사양은 normal, pre, nowrap, pre-wrap, pre-line의 다섯 가지 값을 가질 수 있었으나, wrap이 안 되는 nowrap이나 pre값은 범위를 넘어설 때 자동 개행이 되지 않고 스크롤바를 사용하기 때문에 구현의 어려움이 있다. 따라서 자동 개행이 되는 normal, pre-wrap,

pre-line값만 사용하여 설계하였다.

4.4.4 세로정렬 : -fan-vertical-align

직교좌표계에서 행 높이를 기준으로 텍스트의 베이스라인을 수직 정렬하는 사양에는 vertical-align 속성이 있다[13]. 극좌표계에서도 -fan-vertical-align 사양을 확장하여 텍스트를 세로 방향에서 정렬할 수 있게 하였으며, 세로정렬 속성값은 기존의 vertical-align 값과 동일하게 하였다.

5. 사용성 평가 및 분석

본 연구에서 제안하는 CSS3 확장사양을 적용하여 제작한 웹 문서가 제대로 표현되는지 기존 브라우저에서도 확인하기 위하여 자바스크립트로 전처리기를 구현하였다[4,6]. CSS3 확장사양 전처리기는 -fan- 접두사가 붙은 확장사양을 기존의 CSS 문법

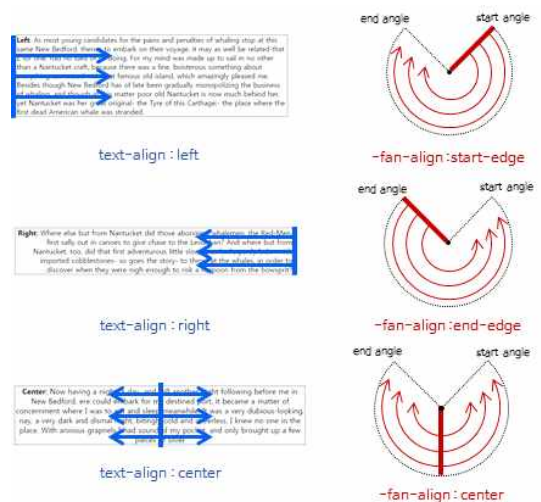


Fig. 11. Alignment in each coordinate system.

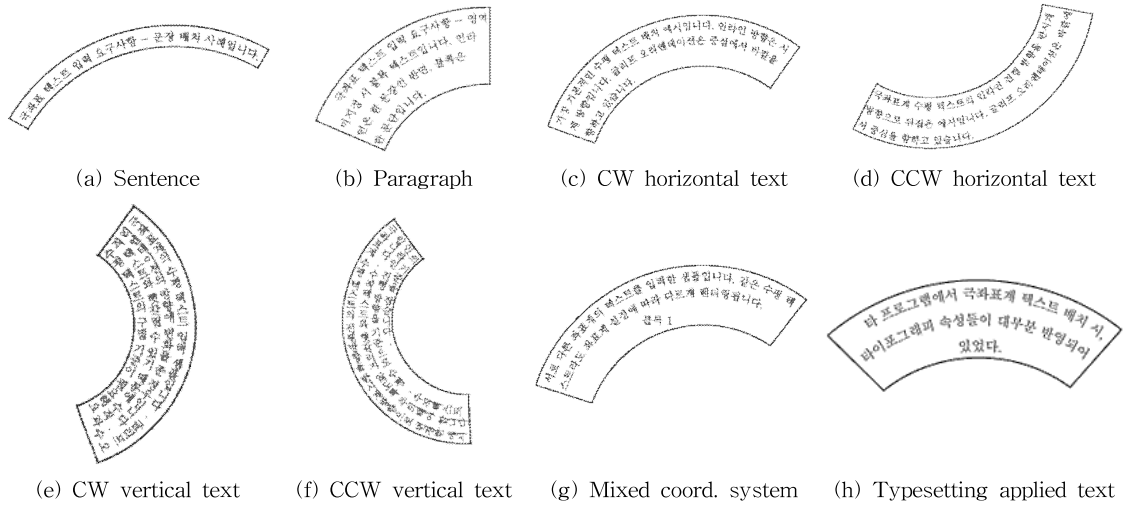


Fig. 12. Sample Text for feasibility test on CSS3 extended specifications.

으로 변환하여 현재의 브라우저에서 볼 수 있도록 결과를 출력한다.

본 절은 확장 CSS3 사양의 효율성을 검증하기 위하여 확장사양을 기존의 극좌표계 텍스트 배치를 돕는 자바스크립트 라이브러리 중 가장 우수한 결과를 보여주는 CswWarp.js와 비교하여 사용성 평가를 진행하였다. 이를 위하여 확장 CSS3 사양과 자바스크립트 라이브러리 CswWarp.js를 이용하여 Fig. 12에서 보여주는 샘플 콘텐츠를 작성하여 웹브라우저에서 실행 결과를 확인하고, 두 가지 코드의 토큰 수를 비교하였다. 이 때 샘플 콘텐츠에서 (e)와 (f)의 세로 쓰기는 CswWarp.js로는 표현하기 곤란하지만 제시된 CSS 확장 사양으로는 무리없이 표현이 가능하였다. 확장사양 전처리거나 CswWarp.js 모두 표준 자바스크립트를 사용하여 구현되었으므로 크롬, 익스플로러, 사파리 등 웹브라우저에서 같은 결과를 볼 수 있었다.

동일한 샘플 텍스트를 표현하는 자바스크립트 라이브러리 소스코드와 확장 CSS3 사양의 소스코드를 의미 있는 정보를 지니는 토큰 단위로 분리하여 그 개수를 평가하였다. 그 결과, 검사한 6개의 샘플 모두 기존 자바스크립트 라이브러리보다 본 연구에서 제공하는 확장 CSS3 사양을 사용할 때 토큰 수가 확연히 줄어든 것을 확인할 수 있었다. 문단을 작성할 때 확장사양의 토큰 수는 자바스크립트 라이브러리의 토큰 수보다 0.82배로 감소하여 가장 큰 효율을 보였

으며, 시계 방향 가로쓰기나 반시계 방향 가로쓰기에서도 낮은 감소율을 보였으나, 모두 문단을 작성하는 점에서는 공통적이다. 또한 타이포그래피 적용 예시도 비슷한 감소율을 보였다. 이를 통해 본 확장 CSS3 사양은 기존의 방법보다 문단을 작성하는 데 매우 효과적인 것으로 나타났다. 반면, 평범한 한 줄 짜리 극좌표계 문장을 작성할 때나 서로 다른 좌표계를 표현할 때에는 코드의 감소량이 비교적 적긴 하지만, 절반 이상 줄일 수 있었다.

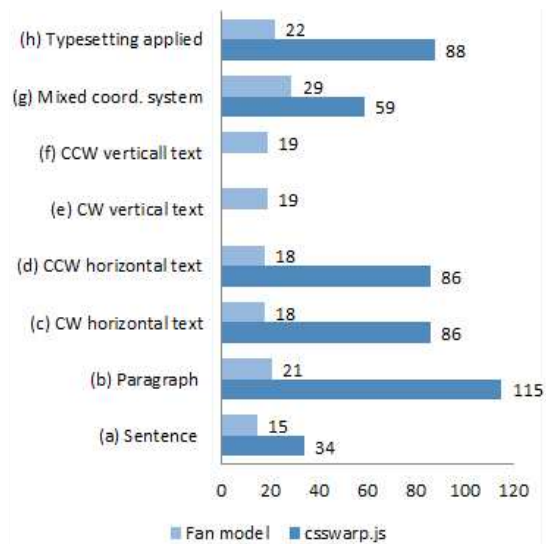


Fig. 13. Comparison of the number of tokens.

6. 결 론

웹 문서에 적용되는 대부분의 스타일시트는 직교좌표계를 기반으로 하고 있으므로 텍스트의 원형 배치가 용이하지 않았다. 이에 본 연구에서는 직교좌표계에서 텍스트가 차지하는 영역을 박스 모델로 정의하였듯이 극좌표계에서 텍스트가 차지하는 영역을 부채꼴 모델로 정의하였다.

이러한 부채꼴 모델의 개념에 기반하여 CSS3를 확장한 CSS3 극좌표계 배치 모듈의 사양을 설계하였다. CSS3 극좌표계 확장 사양은 부채꼴 모델의 설정, 방향 설정, 인쇄조판 속성 등 세 그룹으로 나누어 극좌표계 텍스트 배치의 요구사항을 충실히 반영하여 설계하였다. 그리고 현재의 브라우저에서 확장사양의 효율성을 검증하기 위하여 전처리기를 개발하여 샘플 콘텐츠를 통해 확인하였다.

본 연구에서 제시하는 확장사양이 W3C 표준으로 채택되어 웹브라우저에서 구현된다면 기존의 직교좌표계의 다양한 편집 및 출력 결과가 다양한 원형 디스플레이에서도 가능해질 것이다. 또한 데이터 시각화 및 프리젠테이션 분야에서도 직각 텍스트와 원형 텍스트 배치를 적절히 사용한다면 지금보다 훨씬 다양한 편집 결과를 만들 수 있을 것이다[14]. CSS3의 수많은 모듈과도 호환이 될 수 있으므로 이들과 연동하여 극좌표계 공간에서 텍스트의 애니메이션, 화면 전환 등 다양한 시각적 효과가 가능해질 것이다.

REFERENCE

- [1] W3C, Cascading Style Sheets Home Page, <https://www.w3.org/Style/CSS/> (accessed Nov., 10, 2018).
- [2] E.J. Etemad, "Robust Vertical Text Layout," Unicode Technical Note #22, Unicode Consortium, April 25, 2005, <http://unicode.org/notes/tn22/> (accessed Nov., 10, 2018).
- [3] W3C, CSS Writing Modes Level 3, <http://www.w3.org/TR/css-writing-modes-3/> (accessed Nov., 10, 2018).
- [4] S.M. Shim and S.B. Lim, "Design of CSS3 Extensions for Polar-Coordinate Text Layout in Web Documents," *Korean Institute of Information Scientists and Engineers Transactions on Computing Practices*, Vol. 22, No. 10, pp. 537-545, 2016.
- [5] S.M. Shim and S.B. Lim, "Case Studies and Requirement Analysis for the Circular Layout of Text," *The Korean Journal of Animation*, Vol. 14, No. 4, pp. 53-71, 2018.
- [6] S.M. Shim, *CSS3 Extension for Polar-Coordinate Text Layout and its Preprocessor*, Master's Thesis of Sookmyung Women's University, 2016.
- [7] W3C, CSS Text Module Level 3, <http://www.w3.org/TR/css3-text/> (accessed Nov., 10, 2018).
- [8] W3C, CSS Round Display Level 1, <http://www.w3.org/TR/css-round-display-1/> (accessed Nov., 10, 2018).
- [9] ArcText.js Demos, <http://tympanus.net/Development/Arctext/> (accessed Nov., 10, 2018).
- [10] Dirkweber, Cswarp.js: 'Warp' HTML Text Around an Arbitrary Path, <https://github.com/dirkweber/csswarp.js>. (accessed Nov., 10, 2018).
- [11] W3C, CSS Values and Units Module Level 4, <https://www.w3.org/TR/css-values-4/> (accessed Nov., 10, 2018).
- [12] W3C, CSS Positioned Layout Module Level 3, <https://www.w3.org/TR/css-position-3/> (accessed Nov., 10, 2018).
- [13] MDN, CSS Property - Vertical-Align, <https://developer.mozilla.org/ko/docs/Web/CSS/vertical-align> (accessed Nov., 10, 2018).
- [14] J. Choi, S.Y. Kil, and S.B. Lim, "Development of Web-Based Interface Tool for Map Data Visualization," *Journal of Korea Multimedia Society*, Vol. 20, No. 8, pp. 1216-1223, 2017.



심 승 민

2013년 숙명여자대학교 멀티미디어학과 졸업(학사)

2016년 숙명여자대학교 멀티미디어학과 졸업(석사)

관심분야: 웹표준, 웹/모바일 멀티미디어 응용, 전자출판, User Interface



임 순 범

1982년 서울대학교 계산통계학과 (학사)

1983년 한국과학기술원 전산학과 (석사)

1992년 한국과학기술원 전산학과 (박사)

1989~1992년 (주)휴먼컴퓨터 창업 (연구소장)

1992~1997년 (주)삼보컴퓨터 프린터개발부 부장

1997~2001년 건국대학교 컴퓨터학과 교수

2001년~현재 숙명여자대학교 IT공학과 교수

2006년 University of Colorado 방문교수

관심분야: 컴퓨터 그래픽스, 웹/모바일 멀티미디어 응용, 디지털 방송, 전자출판(폰트, 전자책, XML 문서), User Interface