

A Study of Displaying Electronic Text with Seamless Pages –Magnifying And Simplifying System for Text EXTension (MaSSTExt)–

Seiji Sugiyama
College of Information Science and Engineering,
Ritsumeikan University, Kusatsu, Shiga, Japan
seijisan@is.ritsumei.ac.jp

Miyuki Shibata
Dept. of Humane Informatics,
Otani University, Kyoto, Japan
neko@res.otani.ac.jp

Atsushi Ikuta
Dept. of Humane Informatics,
Otani University, Kyoto, Japan
a.ikuta@sch.otani.ac.jp

Susumu Saito
Dept. of Humane Informatics,
Otani University, Kyoto, Japan
ssm3110@sch.otani.ac.jp

Seiki Miyashita
Dept. of Humane Informatics,
Otani University, Kyoto, Japan
miyashita@res.otani.ac.jp

Abstract

In this paper, we propose a new system for enhanced viewing of electronic text. It is named “Magnifying And Simplifying System for Text EXTension (MaSSTExt)”. This system for displaying text has the operational feeling like a map system displayed on a web browser. No link buttons are used for changing the other pages in this system because it is made by using Ajax. So, it allows us to display huge text files in only one window or seamlessly display/not-display from index to body or annotation data. Using this system, footnotes, references, tables and figures on the text can be located at optimal positions. Comparing simultaneously two different texts is easy to read. Understanding a whole text with both a bird’s-eye view in its totality and a detailed view can be easy. Simulation software “MaSSTExt” of our prototype system is introduced and the effectiveness of our research is shown.

1. Introduction

1.1. Purpose of present study

Paper media has been used from ancient times. However, it has many disadvantages: retrieving the contents is difficult; the storing space is large; preserving information is difficult and so on [1]. Recently, electronic media has appeared. It has many advantages: retrieving the contents is easy; the storing space is small; preserving information is easy and so on [2]. Nowadays, using electronic texts or digital documents is very convenient. Moreover, using the Internet can display various types of data, and has the environment to show the contents while communicating [3][4]. Then, is paper media already unnecessary? The answer is “no” because paper media still has another advantages. We believe that one of the conveniences of paper media is legibility.

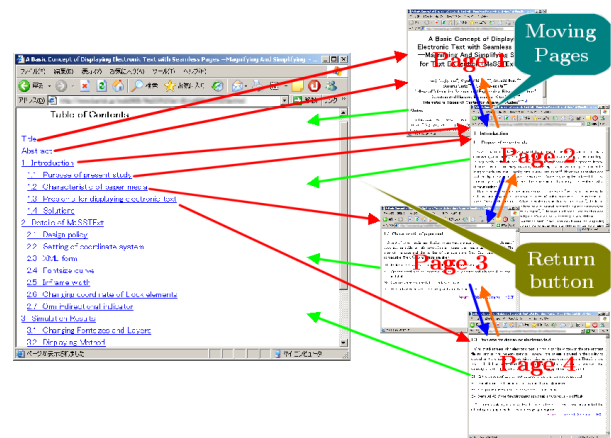


Figure 1. Many link buttons cause confusion

Is electronic text legibility? Many people said that “A book is easier to read than electronic text.”, “I print an electronic text on paper immediately because on a monitor it is not easy to read. And I read the text on paper.”, “I become tired when I read an electronic text.”, “Understanding whole digital documents is very difficult.”. Therefore, it is said that paper media still has advantages due to legibility.

Why is it difficult to read an electronic text? When using a web browser for displaying text, its disadvantages are clear. We believe that the difficulty comes from using many link buttons. In many cases, a long text is divided into many short files as shown in **Figure 1** because it has the limitation of displayable window size or transmission rates. Therefore, the table of contents should be shown and many link buttons are made on the table of contents. Moreover, moving pages sequence is occurring. Return pages sequence is also occurring. Though only 4 pages, it can be seen many link buttons. If using more pages, many link buttons cause confusion. It is thought that using link buttons is natural.

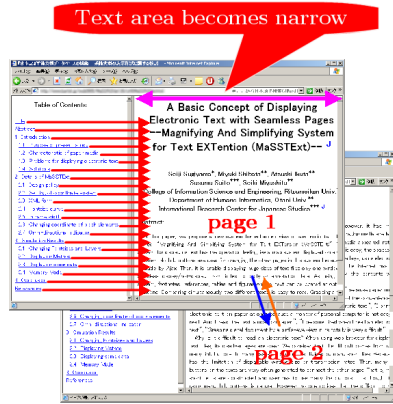


Figure 2. Using frame structure

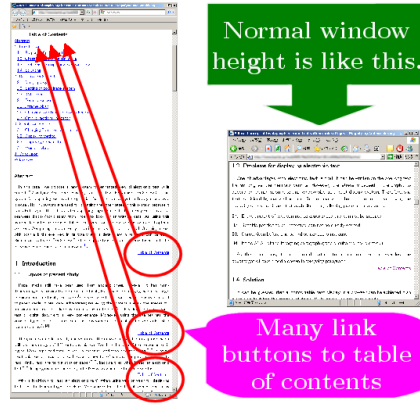


Figure 3. Using one window

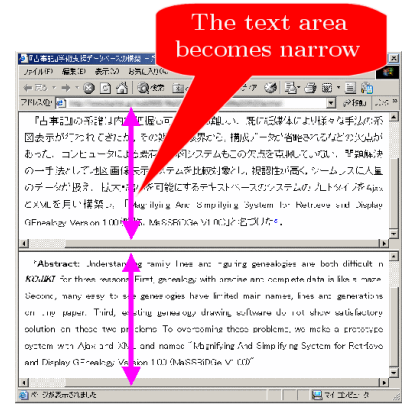


Figure 4. Comparing of two texts

However, no one notices that the difficulty to read comes from using many link buttons. Every time a link button is pushed, the transition is generated. This operation is not seamless. Therefore, the user cannot see the text seamlessly and might get lost from sight. Moreover, it has loss time for generating the transition.

If using frame structure as shown in **Figure 2**, the table of contents becomes legible. However, the text area becomes narrow and text is not legible. The number of link buttons are almost as same as using no frame structure as shown in Figure 1. Moreover, the last text of page 1 and the first text of page 2 cannot be seen at the same time. Therefore, it is also not seamless in this case. On the other hand at the fixed length display like PDF viewer, thumbnail is generated automatically. In such application, to print on paper is natural and text area has zoom function. However, thumbnail is too small and cannot be read in detail, and the size of pages is fixed. That is, text area with variable length cannot be used like HTML in such system. In addition, an electronic book system like i-Pad has appeared and it is explosively sold. However, an e-book style does not have a variable length text on each page. If using e-book system, one advantage of electronic text is lost.

If using one large height window for one text as shown in **Figure 3**, the whole text can be legible. However, the normal window height is often short. So, if scroll down the contents, table of contents cannot be seen. Therefore, many link buttons to table of contents on a intermediate of text is occurring. However, these internal links are not necessary if table of contents is always appeared. Otherwise, if displaying one text with two languages on two frame structure as shown in **Figure 4**, each frame is narrow and cannot be connected to move. It is difficult to display them with the relation.

Link buttons must be used for connecting to the different document's pages. However, we believe that link buttons are not necessary for connecting to the same document's pages. How can such a system be made? If using image data, a map system has no link button for connecting other

images and this system is seamless! However, we want to use text data, and converting text data into image data is a bad sequence for displaying text. Because the volume of text data is smaller than the volume of image data, changing text by using text data is easy, and it is important that it is text data according to the humanities or social affiliated field [5]. Therefore, the system for displaying text should be made by using text data, though constructing the pictures or figures by using image data is good [6].

We will discuss the problems of both paper media and electronic media, and a solution for displaying text without using link buttons with our prototype system "Magnifying And Simplifying System for Text EXTension (MaSSTExt)". "MaSSTExt" has the enhanced viewing that a seamless visible expression for displaying text is achieved by using expansion/reduction of fontsize, display/not-display in each layered structure of the sentence, and movement of text for omni-directional positions.

1.2. Characteristics of paper media

Paper media has physical values that are the size of the paper, thickness of the book, etc. as incidental information. In many cases, users read a text while paging it. The area of one page and the number of characters are fixed. Each page is physically consecutive. The following effects are given.

- A1. A whole text can be experienced and can be grasped.
- A2. The specific part of the text can be inspected and can be quickly opened and it is possible to read in detail.
- A3. Comparing sentences is easy.
- A4. The three above-mentioned points are possible at the same time.

1.3. Problems displaying electronic text

One of the advantages with electronic text is that it can be written in one long text file as long as the memory permits. However, many existing displaying systems have the stress

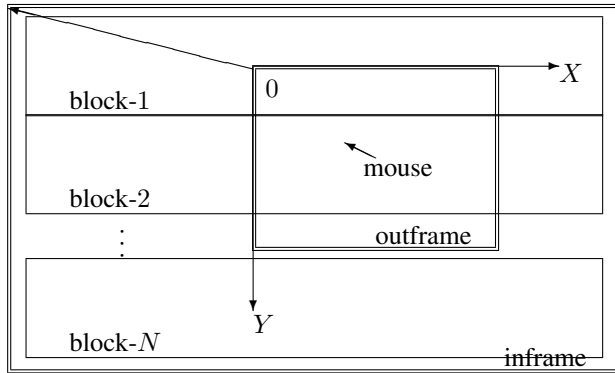


Figure 5. Outline of Coordinate System

to depend on the transmission rate and the window size. So, one text is divided into many short text files and many link buttons are used. Moreover, the capacity of text is different in each file. The following problems are caused.

- B1. A whole text cannot be experienced and cannot be grasped.
- B2. The specific part of the text to be inspected cannot be quickly opened.
- B3. Comparing sentences is not easy.
- B4. Unlike items A1–A3 of the previous paragraph doing them simultaneously with electronic text is difficult.

With these problems, it can be stated that electronic text on a web browser lost the advantages of paper media shown in the previous paragraph.

1.4. Solutions

It can be guessed that a comfortable text display in a web browser can be achieved if an electronic text has the features of items A1–A4 as same as paper media.

To achieve item A1, a bird's-eye view of a whole text is displayed by integrating a whole text and by putting them in one space. Then, a position of the desired text can be grasped. To achieve item A2, displaying text from index to body or annotation data in a seamless way can be achieved by switching display/not-display in each layered structure of the sentence and by switching expansion/reduction of fontsize. Then, table of contents and many frame windows do not need to prepare, and a target position is immediately searchable. Moreover, a detailed view in its parts can be grasped too. To achieve item A3, link buttons are not used by moving the integrated text in a seamless way for omnidirectional positions. The three above-mentioned solutions allows the achievement of item A4.

In this research, we will introduce “MaSSText” of our prototype system for displaying text that has the operational feeling like a map system displayed on a web browser [7]. Moreover, additional useful functions with seamless pages and the effectiveness of our research are shown.

```
<?xml version="1.0" encoding="utf-8"?>
<html>
  <head>
    <width>inframe base width  $w_0$ </width>
  </head>
  <body>
    <h1>title of text</h1>           ...layer0
    <h2>title of section</h2>       ...layer1
    <h3>title of subsection</h3>    ...layer2
    <h4>title of subsubsection</h4>  ...layer3
    <p>
      paragraph (text)           ...layer4
      <q>figures</q>              ...layer5
      <cite>footnotes etc.</cite>   ...layer6
    </p>
    <table>equations (tables)</table> ...layer4
    <dl>items</dl>                ...layer4
  </body>
</html>
```

Figure 6. XML Form

2. Details of MaSSText

2.1. Design policy

We constructed a new system of displaying text like a map system that meets these demands. The map system has overall view and detailed drawing, latitude and longitude information, and reduced scale information. However, a text does not have this information. Therefore, we have to define this information for displaying text as follows.

First, the title and contents of the text are assumed to be equivalent to the overall view in the map system. Texts are also assumed to be equivalent to the detailed drawing. Second, the coordinate system on a given browser of each hierarchy of the sentence structure is assumed to be equivalent to the longitude and latitude in a map system. Third, the concept of the reduced scale does not exist in the text. Therefore, a different fontsize of each hierarchy of the sentence structure will be used. As using these replacements, displaying text seamlessly is achieved.

2.2. Setting of coordinate system

“MaSSText” has been tested using Internet Explorer 6, 7, 8 and Firefox 3 on Windows 2000, xp and Vista. Ajax programming is used for seamlessly displaying text. This coordinate system is defined as a coordinate system on the web browser. **Figure 5** shows the outframe that is the displayable area and the inframe that is put on the text area. The outframe is fixed at the client area of the web browser. The inframe is given by distance measured from top-left position of the outframe. The inframe has block elements of N pieces by dividing text in each layered structure of the sentence. The contents of the inframe that are the outside of the outframe is set to be non-display. Absolute coordinate of the web browser is used for comparing the mouse position and block positions of the inframe.

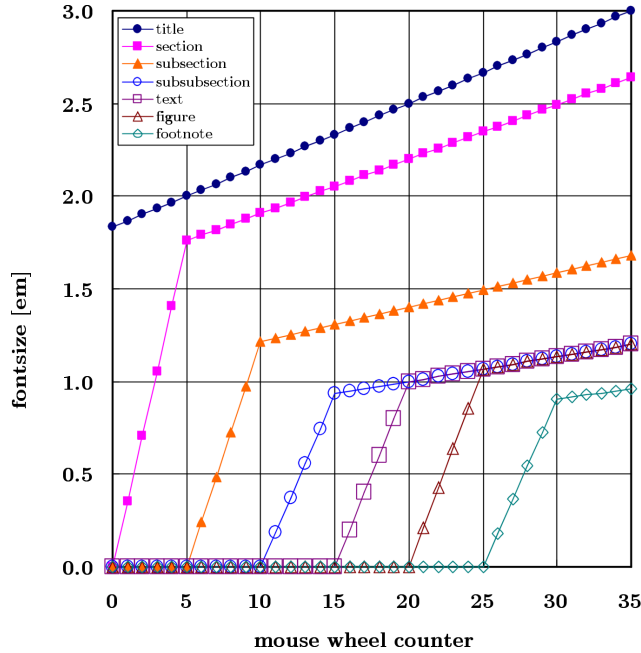


Figure 7. Fontsize Curve

2.3. XML form

XML form is used to input texts for “MaSSText”. This XML is almost the same as HTML form because HTML expresses the layered structure of the sentence well. Moreover, it is convenient for the confirmation of the prototype system. **Figure 6** shows the XML form. The route tag is set as `<html>`. The initialization is set in `<head>` and the texts are set in `<body>`. The tags in `<body>` have the layer numbers in each layered structure of the sentence. HTML tags which are `<h1>`, `<h2>`, ..., `<p>`, `<table>`, `<dl>` and so on can be described almost as it is in `<body>`. However, `<q>` and `<cite>` are used for the figure frame and the footnote, respectively as a different meaning. Moreover, inframe base width is set in `<width>`.

2.4. Fontsize curve

In this system, expansion/reduction of the text area is achieved by changing the fontsize. Fontsize is changed by using up/down mouse wheel count. The unit of fontsize is [em] that is a relative unit. Therefore, the user can use the menu for changing fontsize on the web browser. Even if the fontsize is changed, the magnification of fontsize between layers must be kept. This means the pauses of layered structure of the sentence is maintained. To give the information of expansion/reduction to the user, it is desirable to change font sizes of all layers always. Increase and decrease of the fontsize of a specific layer to which display and non-display are switched should be able to be achieved by spinning the mouse wheel a few times.

Fontsize curve to achieve these demands is defined. The magnification when the layer number is l , and the magnification of $l = 0$ is 1.0 is given as **Equation (1)**.

$$M(l) = \left(\frac{25}{25}, \frac{22}{25}, \frac{14}{25}, \frac{10}{25}, \frac{10}{25}, \frac{10}{25}, \frac{8}{25} \right) \quad (1)$$

A linear function is used for the curve of top-layer that is always displayed. The inclination of this function is $d = 1/30$, the maximum layer number is l_{max} , the number of times for display/not-display is N_r , the maximum mouse wheel count is $C = l_{max}N_r$. Then, **Equations (2a)–(3c)*** represent the fontsize curve $F(l, n)$. **Figure 7** shows the fontsize curve.

if $l = 0$

$$F(0, 0) = 1.83333 \quad (2a)$$

$$F(0, n) = F(0, 0) + dn; \quad (1 \leq n \leq C) \quad (2b)$$

if $l \geq 1$

$$F(l, n) = F(0, n)M(l); \quad (N_r l \leq n \leq C) \quad (3a)$$

$$F(l, n) = F(l, N_r l) \frac{N_r - (N_r l - n)}{N_r}; \quad (N_r l - N_r \leq n \leq N_r l - 1) \quad (3b)$$

$$F(l, n) = 0; \quad (0 \leq n \leq N_r l - N_r - 1) \quad (3c)$$

2.5. Inframe width

If the text area is fixed and the fontsize is only changed, the user cannot understand the feeling of expansion/reduction. Therefore, the inframe width should be changed. The inframe width w_{ne} is given by **Equation (4)**.

$$w_{ne} = w_0 + \frac{n}{2} \quad (4)$$

where w_0 is the inframe base width, n is the mouse wheel count. That is, w_{ne} is changed little by little by changing fontsize**. This parameter w_{ne} gives the user the feeling of expansion/reduction of the text space, though the movement of w_{ne} is very little.

2.6. Changing coordinate of block elements

The mouse position should be set at the center of expansion/reduction position. If using image data, it can be achieved easily. That is, the mouse position only has to be set at the position which has the same rate between the before image size and the after image size (**Figure 8(a)**).

However, if using text data with display/not-display and expansion/reduction, the system should calculate the coordinates of the block elements and the inframe according to the

*. Initial value $F(0, 0)$ was calculated so that the fontsize when non-display of the text layer ($l = 4$) began might become 1.0[em].

**.. w_{ne} can be adjusted for the environment of user because w_0 can be set by using MaSSText’s menu after XML input.

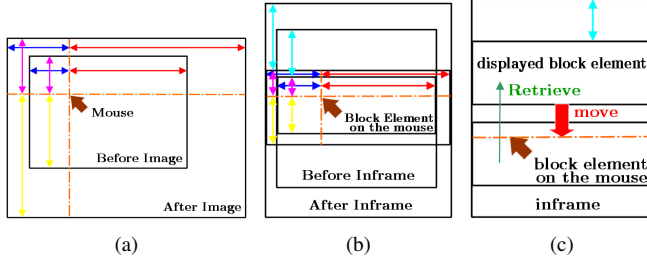


Figure 8. Expansion/Reduction Sequences

following sequence (**Figure 8(b)**). First, the block element on the mouse is retrieved. The rate of this block element is calculated like an image data as shown in **Figure 8(a)**. Second, the relation between the block element on the mouse and the inframe position is calculated.

The coordinate of the mouse is (x_m, y_m) , the coordinate of left-top position of the block element on the mouse before changing the inframe is (x_{oe}, y_{oe}) , the width before changing the inframe is w_{oe} , the height before changing the inframe is h_{oe} , the coordinate of left-top position of the block element after changing the inframe is (x_{ne}, y_{ne}) , the width after changing the inframe is w_{ne} , the height after changing the inframe is h_{ne} , the coordinate of left-top position of the outframe is (x_o, y_o) , the coordinate of left-top position of the inframe after changing the inframe is (x_i, y_i) , then, the magnification rate (r_x, r_y) is represented by **Equation (5)**.

$$r_x = \frac{|x_m - x_{oe}|}{w_{oe}}, \quad r_y = \frac{|y_m - y_{oe}|}{h_{oe}} \quad (5)$$

The distance between the coordinate of left-top of the block element on the mouse after changing the inframe and the coordinate of left-top position of the inframe is represented by **Equation (6)**.

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \begin{bmatrix} x_{ne} \\ y_{ne} \end{bmatrix} - \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad (6)$$

Finally, the coordinate of left-top position of the inframe after changing the inframe (x_{ni}, y_{ni}) is represented by **Equation (7)**.

$$\begin{bmatrix} x_{ni} \\ y_{ni} \end{bmatrix} = \begin{bmatrix} x_m \\ y_m \end{bmatrix} - \begin{bmatrix} w_{ne} r_x \\ h_{ne} r_y \end{bmatrix} - \begin{bmatrix} x_d \\ y_d \end{bmatrix} - \begin{bmatrix} x_o \\ y_o \end{bmatrix} \quad (7)$$

If the block element on the mouse will be not displayed after reduction, the other block element is moved to the mouse position as follows (**Figure 8(c)**). First, the displayed block element is retrieved to the upper direction of inframe. Second, the bottom position of found block element is moved to the mouse height position. Third, the other block elements are moved according to move the found block element. That is, set $r_y = 1$ and calculate y_{ni} of **Equation (7)**. Then, seamless expansion/reduction is achieved and the position of block element on the mouse is not lost.

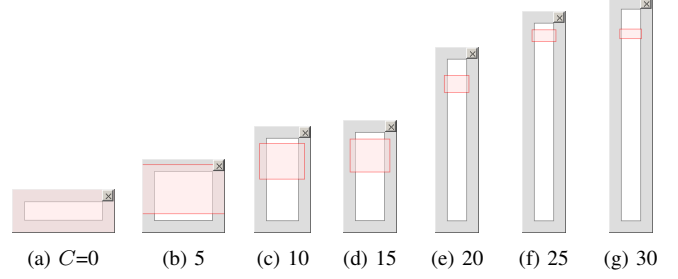


Figure 9. Omni-directional Indicator

2.7. Omni-directional Indicator

In this system, a text in the inframe can be moved to the omni-directional position like a map system. Then, the indicator should be displayed as shown by **Figures 9(a)–(g)**. An indicator on a map system shows only a partial map of the upper layer. On the other hand, the indicator displaying text should show the overview of the text because a bird's-eye view in its totality should be displayed for legibility. Both width w_{ii} and height h_{ii} of the indicator are calculated by using a logarithm with the mouse wheel counts that are denoted at the bottoms of these figures and these maximum sizes are limited until the half size of outframe.

$$w_{ii} = \log \left(\frac{w_i}{h_i} + 1 \right) w_{i0}, \quad h_{ii} = \log \left(\frac{h_i}{w_i} + 1 \right) h_{i0} \quad (8)$$

$$25 < w_{ii} < w_{i0}, \quad h_{ii} < \frac{h_o}{2} \quad (9)$$

where, w_{i0} denotes base width, h_{i0} denotes base height. Moreover, h_o denotes outframe height, w_i denotes inframe width, h_i denotes inframe height.

3. Simulation results

3.1. Changing Fontsizes and Layers

Figures 10(a)–(i) show the captured images of our prototype system “MaSSText”. First, the title of the text is only displayed at the starting software as shown in **Figure 10(a)**. Second, the contents which are section names appear by spinning the mouse wheel once as shown in **Figure 10(b)**. Third, the spaces between section names expand slowly by continuously spinning the mouse wheel several times as shown in **Figure 10(c)**. Fourth, the subsection names appear in the expanded spaces as shown in **Figure 10(d)**. Fifth, the spaces between subsection names grow further by continuously spinning mouse wheel as shown in **Figure 10(e)**. Sixth, the subsection names appear in the expanded space as shown in **Figure 10(f)**. Seventh, the spaces between subsection names grow further as shown in **Figure 10(g)**. Eighth, the texts appear slowly in the expanded spaces by continuously spinning the mouse wheel as shown in

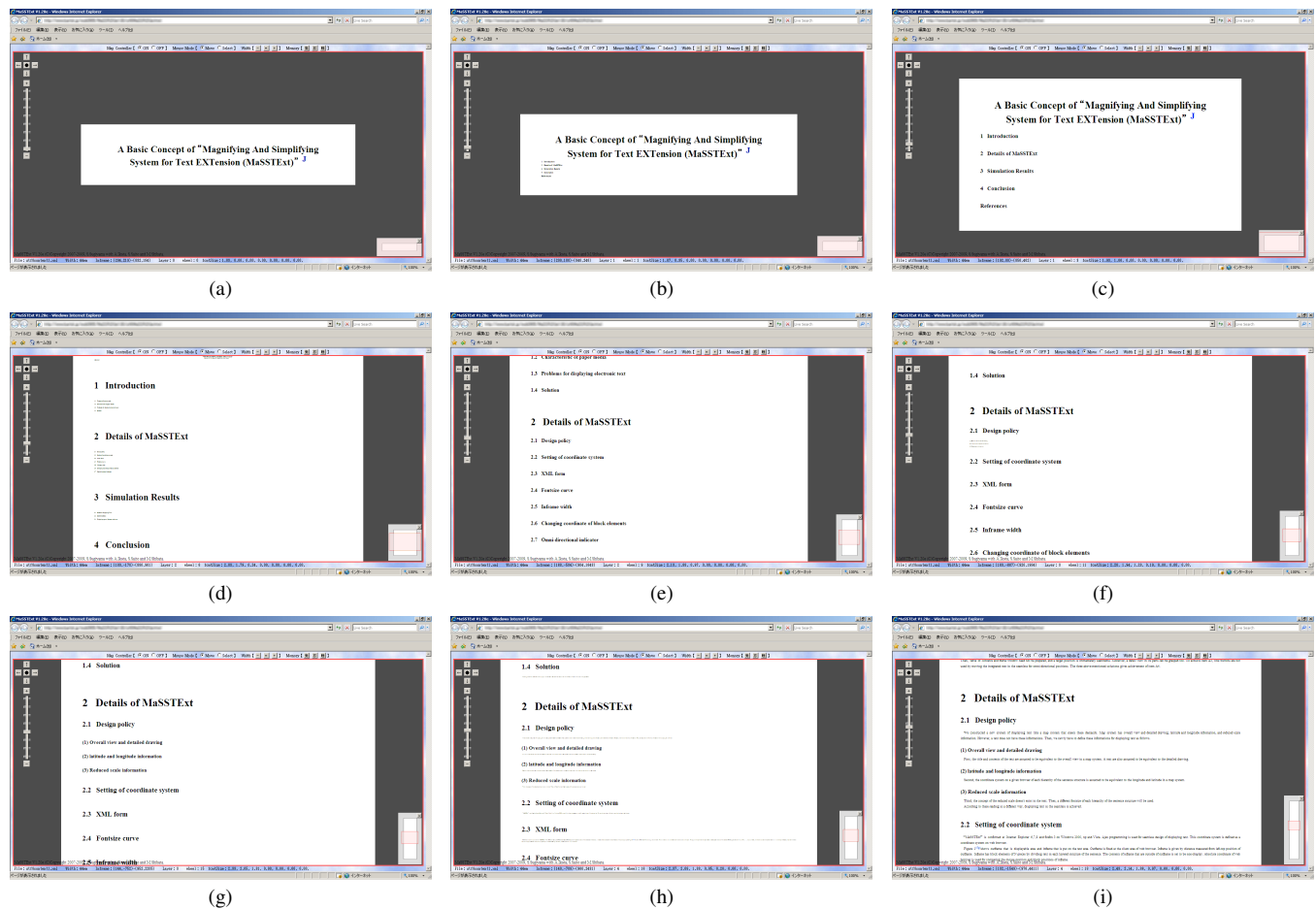


Figure 10. Captured Images of “MaSSText” for Changing Fontsizes and Layers

Figure 10(h). Finally, the user can read the texts of just good for read size as shown in Figure 10(i). Note that up from (a) to (i) and down from (i) to (a) are seamlessly changed. In addition, the large size of the captured image of “MaSSText” is shown in Figure 16.

3.2. Displaying some data in some expanded spaces

These changes of expansion/reduction are seamless by only using the up/down control of a mouse wheel. The white rectangular space which is the inframe with the text can be moved to omni-directional positions by mouse drag control. “MaSSText” does not have the other frame window for table of contents. However, it can be seen by using down control of the mouse wheel. If the user wants to read the text of another section, the user only have to move the position of the mouse arrow right above the section name, and turn the mouse wheel up. Then, the present mouse position’s text appears in the expanded spaces, and it is possible to read the text that the user wants to read quickly without link buttons.

Expanded spaces between texts can be increased to the limitation of the PC memories. Therefore, some data can be

displayed in the spaces. For example, Figure 11(a) shows the “figure” on the space between the texts with the figure’s explanation. The (fig) mark on the text denote the reference of the existing figure which is not display to the user. The text which is near the (fig) mark is divided into two blocks to the top and bottom by using an expansion command. Then, the figure which is not-displayed can be displayed in the expanded space between texts.

Similarly, Figure 11(b) shows the “footnote” on the space between the texts with an explanation. “Reference” of the text is also shown instead of “footnote”. In general, footnote/reference cannot be located at the optimal positions on a web browser by using HTML. However, they can be located at optimal positions in this system by using this concept. Figure 11(c) shows that both original text and translation text of foreign language can be located side by side easily. <cite> in the XML is used for these changes.

3.3. Useful functions

The map-controls on the top-left of the outframe can be used instead of a mouse control. The horizontal segments on

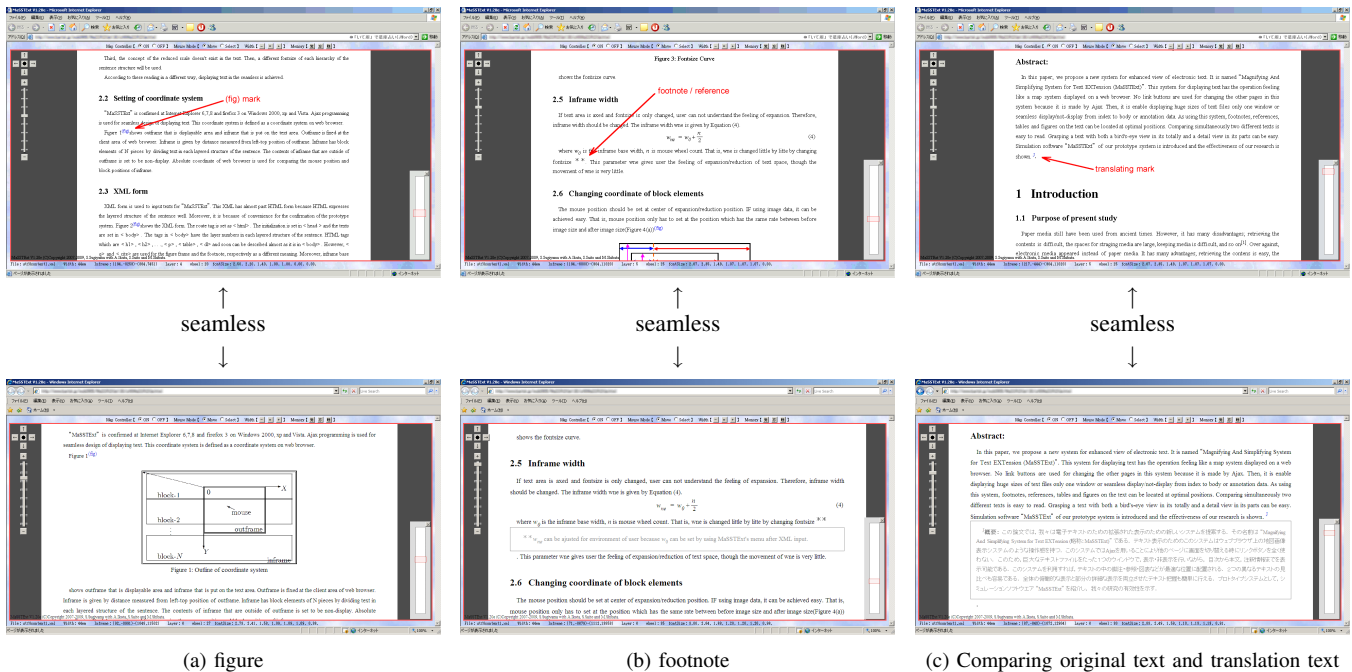


Figure 11. Captured images of “MaSSText” for displaying some data



Figure 12. Map Controller

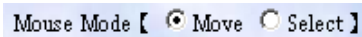


Figure 13. Mouse Mode



Figure 14. Width



Figure 15. Memory

the map-controls denotes the changing layers points. Using the map-controls, the user can reach the maximum layer, minimum layer and any one layer immediately. Display/not-display of the map-controls can be selected by the switching radio buttons as shown in **Figure 12**. The $\uparrow, \downarrow, \leftarrow, \rightarrow$ keys of the keyboard can be used instead of the mouse control for moving text position.

The switching radio buttons of “moving mode” and “selecting mode” as shown in **Figure 13** can be used for changing mouse operations. If the user choose “selecting mode”, copying and pasting texts can be utilized. That is, “MaSSText” can use the standard function of an operating system. If using image data instead of text data, this convenience cannot be given.

The switching radio buttons of “Width” can be set the inframe width. **Figure 14(a)** shows the button for reducing the width. **Figure 14(b)** shows the button for setting the initial width. **Figure 14(c)** shows the button for expanding the width.

If using no link button, text display positions cannot be memorized. That is, the user cannot know the URL at the any place of the text. So, “memory mode” is installed for memorizing the information which are the fontsize, the layer number, the inframe position and so on. Then, the user can recall the information at the next time to use “MaSSText” by

pushing the memory button once. **Figure 15(a)** shows the button for memorizing the current position. **Figure 15(b)** shows the button for deleting the memorized position. **Figure 15(c)** shows the button for recalling the memorized position. Note that these buttons are “KANJI” which show the means directly in Japanese, and this is made by using a cookie system on a web browser.

4. Verification

In general, a text has the structure of the section, subsection, subsubsection, and so on. HTML format is originally made for providing the document structure to display on a web browser. So, it is convenient to write text using HTML. Moreover, web browsers are able to retrieve and/or to classify the text easily. However, existing web browsers did not display the table of contents, the whole text, the detailed text, and so on with automatic classification even when writing HTML with strict rules. As a result, users must construct showing rules themselves for displaying the whole text and the detailed text simultaneously.

Many users do not prefer this task. After all, texts can be shown without strict rules of HTML tags. So, many users disregard the rules and prefer the free fontsize and/or free layout like a word possessor. We believe that this is a reason

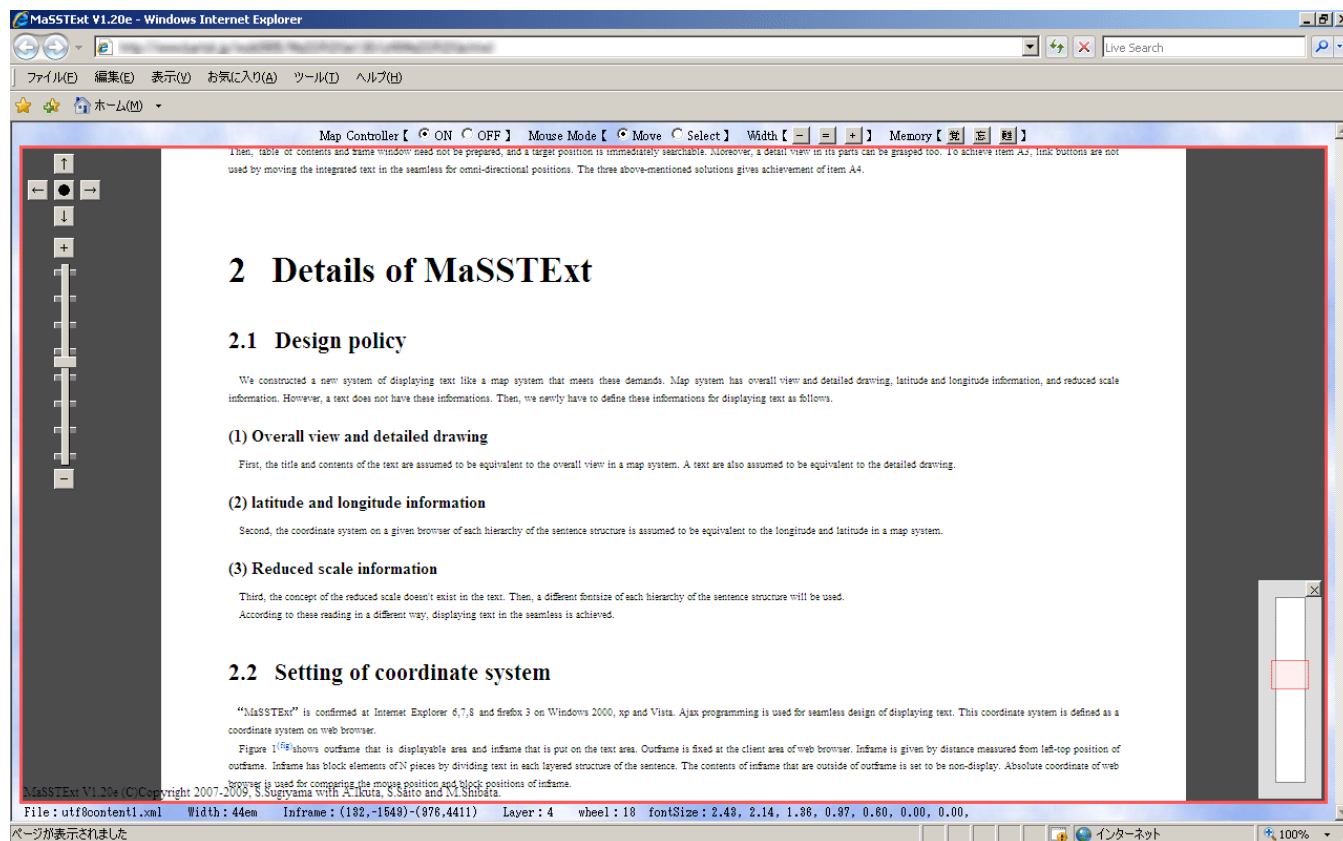


Figure 16. The large size of the captured image of MaSSText

of preferring the style sheet rules. However, there are many difficult viewing sites on the Internet.

On the other hand, using MaSSText, seamless text viewing can be constructed by using only strict rules of sentence structure like strict rules of HTML. Of course, one huge text has no link button in this system but the user can reach all locations of the text immediately by using mouse operations. Therefore, existing unclear rules of HTML can be resolved. Why can MaSSText do so? Because MaSSText has not only the expansion/reduction command but also the display/not-display command according to the text layer structure and can use the variable length text that is a feature of electronic text. So, MaSSText can give us the legibility of text.

5. Conclusion

In this study, the concept of MaSSText as a new legible displaying method of electronic text was described. Electronic text that was previously difficult to read, has now been made clearly legible by using MaSSText. We are currently in the process of creating a format that is user friendly, and can be used by a wide range of people from many fields. Some solutions of the input format and many applications by using this concept will be discussed in the future.

References

- [1] K. Ohno. Digital content types based on information media symmetry. In *IPSJ SIG Notes*, 2004(36), pages 39–46, 2004.
- [2] K. Ohno. A study on compound document technology. In *IPSJ SIG Notes [In Japanese]*, 2005(72), pages 41–48, 2005.
- [3] The universal digital library - million book collection. Carnegie Mellon University. <http://www.ulib.org/>.
- [4] Google book search. <http://books.google.co.jp/>.
- [5] S. Sugiyama, S. Saito, A. Ikuta, and M. Shibata. Kojiki knowledge assistant database system for academic usage : A study concerning display form of genealogy historical materials. In *IPSJ SIG Notes [In Japanese]*, 2007(78), pages 47–54, 2007.
- [6] S. Sugiyama, M. Shibata, A. Ikuta, S. Saito, and S. Miyashita. A study for displaying book with seamless pages : Basic concept for magnifying and simplifying system for enhanced view of text. In *IPSJ SIG Notes [In Japanese]*, 2007(100), pages 37–44, 2007.
- [7] S. Sugiyama, M. Shibata, A. Ikuta, S. Saito, and S. Miyashita. A basic concept of displaying electronic text with seamless pages –magnifying and simplifying system for text extension (masstext)–. In *Proc. of IEEE int. Conf. on 2009 World Congress on Nature and Biologically Inspired Computing (NaBIC 2009), CISIM2009-98 SS3-5*, pages 953–958, 2009.

Authors Biographies

Seiji Sugiyama has been an assistant professor at the College of Information Science and Engineering, Ritsumeikan University, Japan, since 2009. He received his B. Eng., his M. Eng. and his Ph.D. Eng. from Ritsumeikan University, in 1992, 1997 and 2000, respectively. He was a part-time lecturer at both Ritsumeikan University, during 2001–2004 and 2007, and Otani University, Japan, in 2001 and 2002, respectively. He was a lecturer at Otani University, during 2003–2006. And he was a research associate at Ritsumeikan University, in 2008. He was awarded a research prize “Yamashita-kinen” for a paper on MaSSTExt from the Information Processing Society of Japan, in 2009. His research interests include interface designing and robotics.

Atsushi Ikuta is currently a part-time lecturer at Otani University and Ryukoku University, Japan. He received his B.A. and M.A. degrees at Kyoto Prefectural University, Japan, in 1997 and 1999, respectively. His research interests include historical science and interface designing.

Miyuki Shibata is currently an Associate Professor at the Department of Humane Informatics, Faculty of Letters, Otani University, Japan. She received her B.A. at Kyoto Notre Dame University, Japan, in 1987. Her research interests include interface designing and intellectual property law.

Susumu Saito is currently a part-time lecturer at Otani University, Japan. He received B.A and M.S. degrees in Kyoto University, in 1997 and 1999, respectively. His research interests include agricultural science, database systems, and geographic information systems.

Seiki Miyashita is currently a professor at the Department of Humane Informatics, Faculty of Letters, Otani University, Japan. He received his B.A. at University of Kanazawa, Japan, in 1973 and his M.A. at Otani University, Japan, in 1976. He served as a Director of Otani University Library during 1994–1996, a Director of Otani University Shin Buddhist Comprehensive Research Institute during 1998–2000, a Chair of Department of Humane Informatics during 2000–2002, and as a Dean of Faculty during 2004–2006. His research interests include Buddhist studies and editing systems for humanities.