

Webページのレイアウト調整システムの提案 A Proposal of a Layout Adjustment System for Web Pages

若松 祐樹† 早川 智一† 疋田 輝雄†
Yuki Wakamatsu Tomokazu Hayakawa Teruo Hikita

1. はじめに

近年、Web サイトの数が増加している。Netcraft 社の報告[4]によれば、2014年5月時点でのWeb サイト数は約9億7,526万で、前月比で1,600万以上の増加とのことである。

Web サイト数の増加は、Web サイト間のアクセス競争に拍車をかけることにつながった。これは、Web サイトのアクセス数を増やすことで情報提供の機会が増えるため、Web サイト作成者がアクセス数を重要視するからである。

一般に、作成したWeb サイトへのアクセス数を増やすためには、他のWeb サイトとの差別化が必要である。ここで、差別化が可能である要素として、(1) Web サイト上の情報の質、(2) Web サイト上の情報の量、(3) Web サイトのデザインやレイアウトがあると考えられる。ここで扱う(3)には、一般的に決まったモデルは存在せず、作成者が自由に施すことが可能である。

Web サイトのデザインは、利用者にWeb サイトの印象を残すことができる点などで、差別化に大きな役割を果たす。実際に、Web サイトのデザインが評価されて「グッドデザイン賞[11]」を受賞したことにより、大きな宣伝効果を得た企業が存在する[10]。

しかし印象的なWeb サイトを作成したとしても、それが必ずしもWeb サイトの利用者にとって使いやすいものであるとは限らない。一般に、利用者が使いやすいと感じるWeb サイトの特徴の1つに、Web サイトのデザインが各利用者の好みに合っているというものがある。もし、Web サイトのデザインが好みに合わず、Web サイトが使いにくいものだとした場合、あらかじめ決められたデザインに従って利用者はWeb サイトを閲覧しなければならない。

本研究の目的は、Web サイトのデザインを各利用者の好みに変更することでWeb サイト利用者の利便性を高めることである。また、ウェブプログラミングの知識がない利用者にも使えるように、HTML (HyperText Markup Language) や XHTML (Extensible HyperText Markup Language)、CSS (Cascading Style Sheets) などの知識を必要とせずに、Web ページのレイアウトをウェブブラウザ上で各利用者が変更できるシステムを提案する。

本論文の構成は以下のとおりである。2節では関連するシステムや論文について言及する。3節では提案手法について説明する。4節では提案システムの設計と実装について詳説する。5節では提案システムの評価結果について報告する。6節ではまとめと今後の展望について述べる。

2. 関連研究

2.1. Stylebot

Stylebot [2] は、閲覧中のWeb ページのスタイルを変更できるウェブブラウザ Google Chrome の拡張機能である。Stylebot は、一部項目の値を GUI で編集できる Basic モードと、スタイルシートを記述して編集を行う Advance モードの2つを利用することで、Web ページのスタイルの変更を実現する。Basic モードは、テキストや背景、HTML 要素の枠、レイアウトに関するスタイルの編集を可能にする。ボタン操作でスタイルの編集を行うので、CSS についての知識を必要とせずに利用できる。ただし、レイアウトについて編集できることは、(1) HTML 要素の表示・非表示、

(2) HTML 要素のマージンとパディング値の変更の2つだけである。Advance モードは、HTML 要素に対して直接 CSS を記述することで、あらゆるスタイルの編集を可能にする。Web ページのレイアウトも自由に変更可能であるので、利用者の好みに応じたレイアウトを施すことができる。ただし、このモードを利用するためには、CSS についての知識が必要である。

提案システムと Stylebot とは、Web ページのデザインを変更できる点で類似性があるが、レイアウトの変更方法や使えるブラウザが異なる。Stylebot は、Google Chrome の拡張機能であるため、ウェブブラウザが Google Chrome でなければ使用することができない。また、利用者の好みに応じたレイアウトの変更を行うためには、CSS についての知識が必要である。一方、提案システムは、特定のウェブブラウザに依存せず、マウス操作でのレイアウトの変更を可能にすることで、CSS の知識を必要としないレイアウトの変更を実現する。

2.2. 関連論文

西ら [13] は、ウェブブラウザ上で Web ページをリアルタイムに編集可能にするシステムを提案している。土井ら [12] は、このシステムを拡張し、ブックマークレットを用いることで、ウェブブラウザ上で Web ページの編集を可能にする手法を提案している。これらのシステムの目的は、Web ページの容易な編集を実現することで情報の発信の支援を行うことである。これらのシステムは、Web ページの編集を行った際に HTML ファイルを書き換えて保存し、その編集結果をすべての Web ページ閲覧者のウェブブラウザ上に反映させる。また、HTML ファイルを書き換えるシステムの性質上、すべての Web ページ閲覧者に対して編集権限を与えるのはリスクが高いことから、編集権限の管理と Web サイト内の管理を行えるようにしている。

彼らのシステムと提案システムとは、ウェブブラウザ上でレイアウトの変更が行える点で類似性はあるが、

† 明治大学理工学研究科

School of Science and Technology, Meiji University

HTML ファイルの扱い方やシステムの適用範囲が異なる。彼らの手法では、書き換えた HTML ファイルを Web サーバ上に保存する必要があるが、提案システムではその必要がない。それにより、他の Web ページ利用者とレイアウトの変更に関して干渉することがなく、閲覧ごとに Web ページの編集権限を与える必要もない。また、彼らのシステムは、自分で作成した Web ページしかレイアウトの変更ができないが、提案システムでは、閲覧できる Web ページであればレイアウトの変更が行える。

3. 提案手法

3.1. 概要

我々は、Web ページを閲覧するとき、その Web ページのレイアウトを利用者ごとに変更できるようにするため、

- (1) Web サーバから受け取る HTML ファイルをウェブブラウザ上で編集し、
- (2) 編集後のレイアウト情報をウェブブラウザのローカルに保存する

という手法を提案する。(1)により、必要なソフトやプラグインなどをシステム利用者がインストールする手間を省くことができる。さらに、(2)により、Web ページのレイアウトを他の利用者と共有しないので、同一の Web ページに対して利用者ごとに異なるレイアウトを施すことができる。なおレイアウトの変更は各ユーザに対して一時的であるため、HTML ファイルのコピーライトに関して問題は生じないと考える。

3.2. 提案システムの実行情例

提案システムを適用してレイアウトを変更した Web ページの例を 2 つ、図 1 と図 2 に示す。Yahoo! JAPAN [9] と MSN Japan [3] を例に用いた理由は 5 節で説明する。図 1

(b) と図 2 (b) では、利用者のローカル環境の画像を背景に適用している。さらに図 1 (b) では、ニュース等を表示する中央部の要素の位置を変更し、広告やメール等を表示する右側の要素を削除している。図 2 (b) では、HTML 要素の位置の変更はしていないが、多くの HTML 要素を削除している。これは、図 2 (a) と図 2 (b) とのシークバーの長さが異なることより確認できる。

また、図 1 で示した Web ページに対応する body 要素以下の HTML ドキュメントを図 3 に示す。図 3 (a) と図 3 (b) とでは、id が topixsbox と wcup である 2 つの HTML 要素の順序が入れ替わっている。また、図 3 (a) では id が sub の HTML 要素が存在しているが、図 3 (b) ではその HTML 要素が存在しない。そのため、図 1 (b) の Web ページ上にも id が sub の HTML 要素が表示されていない。そして、図 3 (a) では body 要素に属性が何も付与されていないが、図 3 (b) では style 属性が付与されている。この style 属性では、Web ページの背景を設定できる background プロパティにローカル環境の画像ファイルを指定している。これにより、図 1 (b) のように背景を利用者好みの画像に変更できる。

3.3. システムの適用方法

提案システムの適用方法に関する概要図を図 4 に示す。これは、システム適用のための独自プロキシを介して Web ページを閲覧するときはレイアウトの変更ができるが、プロキシを介さずに Web ページを閲覧するときはレイアウト



(a) レイアウト変更前の Web ページ



(b) レイアウト変更後の Web ページ

図 1 提案システムによる Yahoo! JAPAN のレイアウトの変更例



(a) レイアウト変更前の Web ページ



(b) レイアウト変更後の Web ページ

図 2 提案システムによる MSN Japan のレイアウトの変更例

トの変更ができないことを表している。提案システムは、プロキシサーバ内でシステムの適用 (4.4 節) を自動的に行うことで透過的に利用できる。

<pre> <body> <div id="wrapper"> ~省略~ <div id="contents" class="clearfix"> <div id="toptxt"> ~省略~ </div> <div id="navi"> ~省略~ </div> <hr class="separate"> <div id="division"> <div id="main"> <div id="topicsbox"> ~省略~ </div> ~省略~ <div id="wcup"> ~省略~ </div> ~省略~ </div> <hr class="separate"> <div id="sub"> ~省略~ </div> </div> ::after </div> ~省略~ </div> </body> </pre>	<pre> <body style="background: ~省略~"> <div id="wrapper"> ~省略~ <div id="contents" class="clearfix"> <div id="toptxt"> ~省略~ </div> <div id="navi"> ~省略~ </div> <hr class="separate"> <div id="division"> <div id="main"> <div id="wcup"> ~省略~ </div> ~省略~ <div id="topicsbox"> ~省略~ </div> ~省略~ </div> <hr class="separate"> <div id="sub"> ~省略~ </div> </div> ::after </div> ~省略~ </div> </body> </pre>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

(a) レイアウト変更前

(b) レイアウト変更後

図3 Yahoo! JAPANのbody要素のHTMLドキュメント

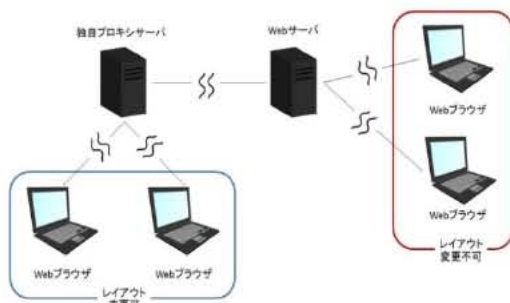


図4 独自プロキシを介した提案システムの適用

3.4. 実現する機能

提案システムは、閲覧中のWebページに対して

- (1) HTML要素の位置の変更,
- (2) HTML要素の削除,
- (3) 削除した要素の復元,
- (4) ローカル環境の画像を背景画像に適用,
- (5) レイアウト変更後のドキュメントのHTML形式でのローカル環境への保存,
- (6) ウェブブラウザへの変更したレイアウト情報の保存,
- (7) レイアウトの初期化

を可能にする。

3.5. レイアウト変更のためのウィンドウ

提案システムは、通常のブラウジングが可能な「ノーマルウィンドウ」とレイアウトの変更が可能な「カスタマイズウィンドウ」の2つのウィンドウを使用してWebページのレイアウトを変更する。これらのウィンドウは、独自に作成したコンテキストメニューで切り替えができる。コンテキストメニューは、ノーマルウィンドウとカスタマイズウィンドウで別々に存在する。それぞれのコンテキストメニューを図5に示す。各メニューの詳細は4.5節で述べる。

3.6. 動作環境

提案システムは、複数のウェブブラウザ上での動作を可能にする「クロスブラウザ」を実現する。提案システムが適用可能なウェブブラウザは、Internet Explorer, Mozilla

```

Start customization of this Web page
Download this document as HTML file
Initialize the setting

```

(a) ノーマルウィンドウのコンテキストメニュー

```

Move / Delete elements
Change background image
Recover deleted elements
Save the settings and exit customization

```

(b) カスタマイズウィンドウのコンテキストメニュー

図5 レイアウト変更のためのウィンドウ

Firefox, Google Chrome, Safari, Operaの5つである。

また、提案システムは、HTML4.01かXHTML1.1で記述されたWebページに対して適用ができるようにした。これは、HTML4.01とXHTML1.1がW3C勧告されているマークアップ言語であり、ほとんどのWebページがこれらで実装されているからである。また、HTML5[6]で記述されたWebページにも適用(4.3節)できる。

4. 提案システムの設計と実装

4.1. 提案システムの動作

図6に提案システムのシーケンス図を示す。この図が示す動作の流れは次のとおりである：

- (0) Webページの閲覧を利用者が要求をする。
- (1) ウェブブラウザがHTTP要求を発行する。
- (2) ウェブブラウザからのHTTP要求をプロキシサーバは受け取り、WebサーバにHTTP要求を転送する。
- (3) WebサーバからのHTTP応答をプロキシサーバは受け取る。
- (4) プロキシサーバは、HTTP応答をウェブブラウザに転送する前に、HTTP応答で受け取ったHTML文書に対してHTML要素のスク립トタグを挿入し、変更したHTML文書をHTTP応答としてウェブブラウザに転送する。
- (5) ウェブブラウザは、プロキシサーバからHTTP応答を受け取る。
- (6) ウェブブラウザは、変更後のレイアウトの情報がウェブブラウザ上に保存されている場合、それに基づいてHTMLファイルの書き換えを行う。
- (7) ウェブブラウザは、書き換え後のHTMLファイルに従ってWebページを利用者に表示する。
- (8) Webページのレイアウトを利用者が変更する。
- (9) ウェブブラウザは、変更後のレイアウトをウェブブラウザに保存し、HTMLファイルを書き換える。

なお(10)以降の処理は(1)以降の処理にそれぞれ相当し、Webページ利用者がレイアウトの変更を行うたびに繰り返し行われる。

4.2. DOMによるHTML要素の操作

提案手法では、Webページのレイアウトの変更を実現する際に、HTMLドキュメントの要素に対して処理を行うためDOM(Document Object Model)を使用する。DOMは、HTMLやXML(Extensible Markup Language)で記述された各要素へ動的にアクセスするためのAPI(Application

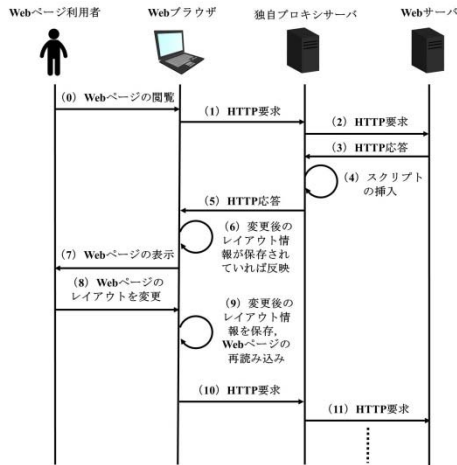


図 6 提案システムのシーケンス図

Programming Interface) であり、ドキュメントの構造的な表現を提供し、ドキュメントの内容や構造の変更を可能にする。

4.3. 提案システムの使用技術

我々は、提案システムを構築するプログラミング言語として JavaScript を採用した。この理由に、

- (1) 主要なウェブブラウザのほとんどで JavaScript がサポートされているため、導入が容易であること、
- (2) クライアント側で Web ページに対する処理ができるため、利用者ごとに異なる動作を実現できること

がある。

また、提案システムは、HTML5 準拠のウェブブラウザ上で動作する。これは、HTML5 から追加・標準化された API を使用することで、システムの機能の一部を実現しているからである。提案システムでは、Drag and Drop、Web Storage、File API の 3 つの API を採用した。

4.4. プロキシサーバ

提案システムを適用するためのプロキシサーバを表 1 のソフトウェアを用いて実装した。Privoxy [5] は、HTML 文書の内容を Perl の正規表現で変更できる Web プロキシソフトである。提案システムでは、</head>文字列を<script> ~ </script></head>文字列に置換することでスクリプトの挿入を実現している。

4.5. 機能の実装

提案システムによる Web ページのレイアウトの変更は、「ノーマルウィンドウ」と「カスタマイズウィンドウ」の 2 つのウィンドウを使用することで実現する。提案システムの動作のフロー図を図 7 に示す。

4.5.1. ノーマルウィンドウ

ノーマルウィンドウは、Web ページにアクセスしたときに最初に表示するウィンドウである。このウィンドウでは、ブラウザのローカルストレージに保存されているレイアウト変更後の情報に基づいて、Web ページ読み込み時にレイアウトの変更を自動的に行う。また、独自のコンテキストメニューを開いて、図 5 (a) に示す 3 つのメニューを実行することができる。それぞれのメニューの実装を以下に述べる。

表 1 独自プロキシサーバの実装に使用したソフトウェア

ソフトウェア	バージョン	用途
Privoxy	3.0.21-3	Web フィルタ
CentOS	6.5	OS

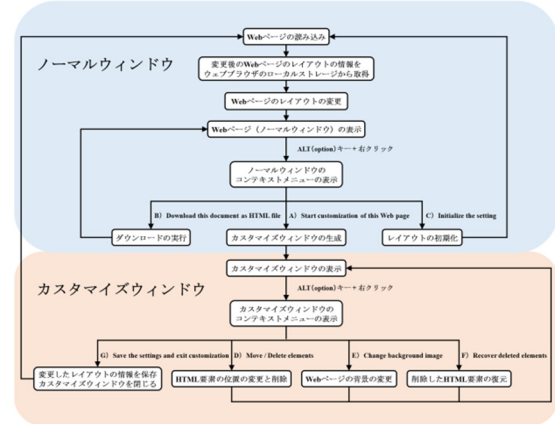


図 7 提案システムの動作のフロー図

a) Start customization of this Web page

window.open () メソッドを用いて新たにカスタマイズウィンドウを生成し表示する。

b) Download this document as HTML file

html タグの innerHTML を取得して encodeURIComponent 関数の引数として渡すことでダウンロードの実現をする。

c) Initialize the setting

ウェブブラウザのローカルストレージに保存しているレイアウトの変更情報を削除する。

4.5.2. カスタマイズウィンドウ

カスタマイズウィンドウは、ノーマルウィンドウが生成し呼び出す独自のウィンドウである。このウィンドウで表示する Web ページのドキュメント構造は、ノーマルウィンドウのものと同様であるが、以下の 4 つの点で異なる。

- (1) a タグでの Web ページへのリンクが存在しない。
- (2) オリジンサーバとの通信を無効にしている。
- (3) もともと Web ページに適用されていたスクリプトを無効にしている。
- (4) レイアウトの変更が可能である。

(1), (2), (3) を実現した理由は、レイアウトの変更を行う際の不都合を取り除くためである。レイアウトの変更中に他の Web ページへ遷移してしまうと変更内容が保存できないため、(1), (2) の処理が必要である。また (3) は、提案システムと他のスクリプトとの干渉を最小限に抑えるために行う。(4) は、コンテキストメニューを用いて行い、図 5 (b) に示すメニューを実行することで実現する、それぞれのメニューの実装を以下に述べる。

d) Move / Delete elements

操作したい HTML 要素には、getElementById () メソッドでアクセスする。この関数を使用する性質上、操作対象となる HTML 要素には id 属性が必要不可欠である。位置

の変更をするときは、対象となる HTML 要素に `ondragstart`・`ondragover` イベントと値が `true` の `draggable` 属性を付与する。また、操作対象となる HTML 要素の親要素には `ondrop` イベントを付与する。削除をするときは、削除したい HTML 要素に `ondblclick` イベントを付与する。

e) Change background image

`dataTransfer` オブジェクトを使うことで、ドラッグしたローカル環境の画像ファイルのデータを保持し、それを `ondrop`・`ondragover` イベントを付与した `body` 要素にドロップして渡すことで、背景にその画像ファイルを適用する。

f) Recover deleted elements

削除した HTML 要素を別のウィンドウ上に表示させ、その要素に `draggable` 属性、`ondragstart`・`ondragover` イベントを付与する。復元したい HTML 要素の親要素に `ondrop` イベントを付与することで、ウィンドウ間のドラッグ&ドロップによる HTML 要素の移動を可能にする。

g) Save the settings and exit customization

ウェブブラウザのローカルストレージに、ルート要素から順に取得した HTML 要素の id 値の配列、削除した HTML 要素の id 値の配列、背景に設定した画像ファイルの Blob URI を保存する。その後、`window.close()` メソッドでカスタマイズウィンドウを閉じる。

5. 評価

提案システムの有用性を評価するために、提案システムのユーザビリティの測定を行った。Whitney [7,8] は、効率性 (Efficient)、有効性 (Effective)、魅力 (Engaging)、エラー対応 (Error Tolerant)、学習容易性 (Easy to Learn) の 5 つの E を評価することでユーザビリティを客観的かつ公平に分析することができると主張している。我々は、彼女の主張する 5 つの E を、利用者視点 (効率性、魅力、学習容易性) と開発者視点 (有効性、エラー対応) の 2 つに分け、今回は利用者視点でのユーザビリティの測定を行った (今回の被験者の数は 6 名である)。

効率性の評価では、レイアウト変更前と変更後の Web ページ操作の比較を行った。我々は、評価のために自作の画像描画用のページを用意し、課題として出された単純な絵を被験者が描き終わるまでにかかる時間を測定した。被験者を 3 人ずつに分け、一方はレイアウト変更前の Web ページを、もう一方には、我々があらかじめレイアウトの変更を行うことで使いやすくなったと考える Web ページを操作してもらった。効率性の評価結果の平均値を表 2 に示す。

魅力の評価では、被験者に提案システムを利用してもらった後で、そのシステムについてのアンケート調査を行った。アンケートは、

- (1) 操作性：システムは使いやすいか、
- (2) 学習容易性：システムの使い方は覚えやすい、
- (3) 有用性：システムは役に立つか、
- (4) 推薦度：他者へシステムを推薦したいか、
- (5) 期待度：実際に自分がシステムを使いたいのか

の 5 つの項目に対して 5 段階評価で回答してもらった。魅力の評価結果の平均値を表 3 に示す。

学習容易性の評価では、利用者が提案システムの使い方を覚えるまでの時間を測定した。我々は、被験者が

- (1) 提案システムの説明書を読みながら実際にシステムを動かして使い方を学習し、
- (2) 最後に、レイアウトの変更に関する課題を説明書なしで解く

という一連の動作を完了するまでの時間を評価値とした。学習容易性の評価結果の平均値を表 4 に示す。

また、提案システムによる、Web ページのレイアウト変更の有効性の評価を行った。評価には、3 種類の Web ページと 5 種類のウェブブラウザ (表 5) を用いた。今回、レイアウトの変更を行った Web ページは、Yahoo! JAPAN, Infoseek [14], MSN Japan の 3 つのサイトのトップページである。これらの Web サイトはすべてポータルサイトであり、多くの情報を提供し利用者も多いため評価の対象とした。(Alexa [1] は、2014 年 6 月 29 日時点で、Yahoo! JAPAN, Infoseek, MSN Japan のアクセス数がそれぞれ日本で第 1 位、第 112 位、第 89 位だと報告している。) 有効性の評価結果を表 6, 表 7, 表 8 に示す。評価結果に使用した記号「○」は、正しく処理が行えたことを示し、「△」は、処理は行えるが一部不具合があったことを示し、「×」は、正しく処理が行えなかったことを示す。一部不具合というのは、変更後のレイアウトが反映する場合と反映しない場合があるというものであった。これは、Web ページを読み込む時に本スクリプトが正常に読み込まれないことがあるのが原因だと考える。また正しい処理ができなかった原因は、ブラウザごとの挙動の違いを本スクリプトが吸収しきれなかったのだと考える。

評価の結果、Web ページのレイアウトを変更することで、Web ページを利用する際の利用者の作業効率が 23% 上がることがわかった。また、提案システムの魅力の評価する 5 つの項目すべてで平均 3 以上の評価を得た。そして、提案システムの使い方を覚えるまで、平均 20 分かからな

表 2 提案システムの効率性の評価結果の平均値

	レイアウト変更前 所要時間	レイアウト変更後 所要時間
平均	7 分 45 秒	5 分 57 秒

表 3 提案システムの魅力の評価結果の平均値

	操作性	学習 容易性	有用性	推薦度	期待度
平均	3.67	3.33	4.67	4	3.5

表 4 提案システムの学習容易性の評価結果の平均値

	所要時間
平均	16 分 26 秒

表 5 提案システムの有効性の評価に用いたウェブブラウザ

ウェブブラウザ	バージョン
Internet Explorer	11.0.9600.17126
Mozilla Firefox	30.0
Google Chrome	35.0.1916.153m
Safari	7.0.4
Opera	22.0.1471.70

表6 Yahoo! JAPANでの提案システムの有効性の評価

	移動	削除	復元	背景	DL
IE	○	△	○	△	×
Firefox	○	△	○	△	○
Chrome	○	△	○	△	○
Safari	○	△	○	△	×
Opera	○	△	○	△	○

表7 infoseekでの提案システムの有効性の評価

	移動	削除	復元	背景	DL
IE	○	○	○	△	×
Firefox	○	○	○	○	○
Chrome	○	○	○	○	○
Safari	○	○	○	○	×
Opera	○	○	○	○	○

表8 MSN JAPANでの提案システムの有効性の評価

	移動	削除	復元	背景	DL
IE	×	×	×	×	×
Firefox	○	○	○	○	○
Chrome	○	○	○	○	○
Safari	○	○	○	○	×
Opera	×	×	×	×	○

いことがわかった。

しかし一方で、システムによるレイアウト変更の完全な適用およびクロスブラウザの実現が完全ではない例があることが判明した。

6. おわりに

本論文では、Web ページ利用者の利便性を高めるために、Web ページのレイアウトをブラウザ上で変更できるシステムを提案した。評価の結果は、提案システムを使うことで Web ページ利用者の作業効率を向上できることを示した。この結果から、我々は、提案システムのユーザビリティは実用に足るものであり、提案システムが Web ページ利用者の利便性を高めるために有用だという結論を得た。

今後の課題としては、追加の評価やより完成度の高いクロスブラウザの実現を行う予定である。追加の評価としては、開発者視点（有効性、エラー対応）の調査を行う予定である。有効性の調査では、提案システムがどのような Web ページにどの程度適用できるかを評価する。エラー対応の調査では、提案システムを導入することで起こり得るエラーに関して評価する必要がある。また、現段階では、使用するウェブブラウザによって提案システムの挙動が異なることがあった。これを改善し、より完成度の高いクロスブラウザの実現することで、Web ページ利用者の利便性のさらなる向上を目指す。

今後の展望として、我々は、提案システムの機能性・利便性の向上を検討している。例として、レイアウトの変更だけでなく、Web ページのスタイルも変更できるようにすることを考えている。これにより、利用者の好みにより

一層応じた Web ページの編集ができるため、Web ページ利用者の利便性がさらに向上すると我々は考える。

参考文献

- [1] Alexa Internet, Inc.: Alexa - Actionable Analytics for the Web, Alexa Internet, Inc. (online), available from <http://www.alexa.com/> (accessed 2014-06-29).
- [2] Google, Inc.: Chrome ウェブストア - Stylebot, Google, Inc. (オンライン), 入手先 <https://chrome.google.com/webstore/detail/stylebot/oiaejidbmkiecgbjefioejpgmdaleoha> (参照 2014-05-27).
- [3] Microsoft, Co.: MSN Japan, Microsoft, Co. (online), available from <http://jp.msn.com/> (accessed 2014-06-29).
- [4] NETCRAFT, Ltd.: May 2014 Web Server Survey | Netcraft, NETCRAFT, Ltd. (online), available from <http://news.netcraft.com/archives/2014/05/07/may-2014-web-server-survey.html> (accessed 2014-05-27).
- [5] Privoxy Developers: Privoxy - Home Page, Privoxy Developers (online), available from <http://www.privoxy.org/> (accessed 2014-06-04).
- [6] W3C: HTML5, W3C (online), available from <http://www.w3.org/TR/html5/> (accessed 2014-06-04).
- [7] Whitney Quesenbery: Balancing the 5Es, Cutter IT Journal, February 2004, pp.4-11 (2004).
- [8] Whitney Quesenbery: What Does Usability Mean: Looking Beyond 'Ease of Use' - Whitney Interactive Design, WQ usability, available from <http://www.wqusability.com/articles/more-than-ease-of-use.html> (accessed 2014-06-08).
- [9] Yahoo Japan, Co.: Yahoo! JAPAN, Yahoo Japan, Co. (online), available from <http://www.yahoo.co.jp/> (accessed 2014-06-22).
- [10] 井出一誠: あの Web サービスも受賞! 2013 年グッドデザイン賞を取った Web サービスまとめ | Find Job! Startup, 株式会社ミクシィ・リクルートメント (オンライン), 入手先 <http://www.find-job.net/startup/good-design-award-2013> (参照 2014-06-29).
- [11] 公益財団法人日本デザイン振興会: Good Design Award, 公益財団法人日本デザイン振興会 (オンライン), 入手先 <http://www.g-mark.org/> (参照 2014-06-22).
- [12] 土井達也, 白松俊, 大園忠親, 新谷虎松: ブックマークレットを用いた既存 Web ページのリアルタイム編集機構, 第 74 回全国大会講演論文集, 2012 (1), pp.503-505 (2012).
- [13] 西健太郎, 新谷虎松, 松尾徳朗, 田代慎治, 伊藤孝行: 既存 Web ブラウザを利用したオンライン編集可能な Web ページの実現, 電気学会論文誌 (部門誌) C, vol.125, no.4, pp.660-665 (2005).
- [14] 楽天株式会社: 楽天が運営するポータルサイト: 【インフォシーク】Infoseek, 楽天株式会社 (オンライン), 入手先 <http://www.infoseek.co.jp/> (参照 2014-06-22).