

# ブラウザ上のジェスチャによる DOM要素選択方法の提案

海老澤 雄太<sup>1</sup> 丸山 一貴<sup>2</sup> 寺田 実<sup>3</sup>

**概要**：本研究では、ウェブブラウジングにおいて、ジェスチャ入力を用いて DOM 要素を選択する方法を提案する。DOM 要素の選択は、印刷範囲の設定やウェブデザインの解析、ページの閲覧時のカスタマイズなどにおいて必要となる操作である。従来手法として、マウスドラッグを利用するものやマウスカーソルの位置からそこにある要素を選択する手法があるが、タッチデバイスでは使い辛く、要素を上手く選択できない場合がある。そこで、本手法ではジェスチャを利用することで、より柔軟で簡潔な DOM 要素の選択とタッチデバイスへの親和性の向上を目指す。ブラウザの拡張機能としてシステムを実装し、入力されたジェスチャの位置や大きさから対象とする DOM 要素を推定する。ジェスチャと DOM 要素との共通領域のそれぞれに対する面積比の積を用いて重み付けを行うことで、入力されたジェスチャに見合った要素の選択が可能となる。

**キーワード**：ブラウザ、ジェスチャ、DOM、選択、拡張機能

## 1. はじめに

ウェブブラウジングは、すでに日常生活の中に溶けこんでおり、日々ブラウザを通じて情報検索やオンラインショッピングなどの様々な活動が行われている。

このとき、ユーザが行う操作は基本的にリンクのクリックやフォームへの入力といった簡単なものである。しかし、近年ではブラウザに表示された内容から、それに対応する HTML の内容を指定するための操作をする場合がある。

例えば、ウェブページ中のあるテキストを選択して、コピーする操作が挙げられる。表示され

た文字列から必要な部分を選択することで、そこに対応する HTML のテキストを対象としてコピーや検索を行う。また、ウェブページを部分的に印刷したいという場合にも、表示された内容から印刷対象となる部分を選択する必要がある。さらに、開発においてはウェブデザインの参考のため、ウェブページの一部がどのようなタグで記述されどのような CSS によるデザインが指定されているのかを調べるために、表示された内容から対応する HTML の要素を選択するという用途も増えてきた。

### 1.1 既存手法とその問題点

広く使われている選択操作に、マウスドラッグによる選択が挙げられる。PC など、マウスを用いるデバイスでは一般的な操作であり、特に文字列の選択に関しては有効な手法である。

しかし、文字列の選択に特化している面が少な

<sup>1</sup> 電気通信大学大学院 情報理工学専攻 情報・通信工学専攻

<sup>2</sup> 明星大学 情報学部 情報学科

<sup>3</sup> 電気通信大学 情報理工学部 情報・通信工学科

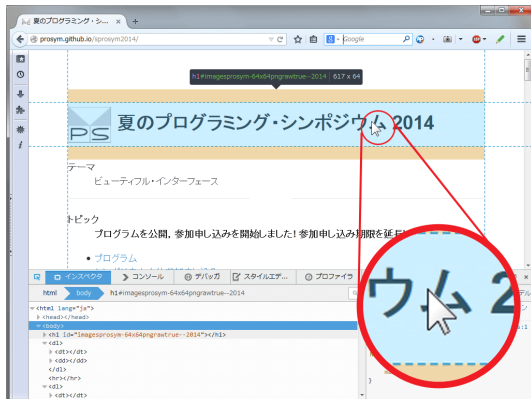


図1 Firefoxの開発者ツールにおける採用例. マウスカールが指す座標にある, 最も前面の要素が選択される.

からずあり, そのため先述の印刷範囲の設定や, デザイン調査のための要素の選択に関してはあまり適した操作ではないと言える.

HTMLの要素の選択については, マウスカールの指す1座標から要素を選択する方法がある. この操作は, 各ウェブブラウザが搭載する開発者向けのツールや, ウェブページでのデザイン解析を行うツール [2] などで用いられている. この手法を採用しているFirefoxの開発者ツールの使用例を図1に示す. この図では, ページのタイトルにあたる文字部分にマウスカールがあるので, 結果としてヘッダー要素 (h1) が選択され, 青い矩形のオーバーレイや寸法線などが表示されている. この手法では, ユーザがどの要素を選択しようとしているのかをリアルタイムで確認することができ, 巨大な要素に関してもクリック1つで選択することができる.

しかし, この方法はマウスが指す1座標から選択対象を決定するため, ユーザの想定と異なる要素が選択される場合がある. 特に, ウェブページ中の表の1行分を選択しようとしても, 一度のマウスクリックで選択できない場合や, 可能だとしても繊細なマウス操作が要求されることがある.

近年ではタッチパネルを搭載したスマートフォンやタブレットなどのデバイスが広く浸透しているが, 上記の2つの方法はこれらのデバイスにとって適していない. マウスドラッグによる方法は, それに相当する機能がないか困難である場合が多

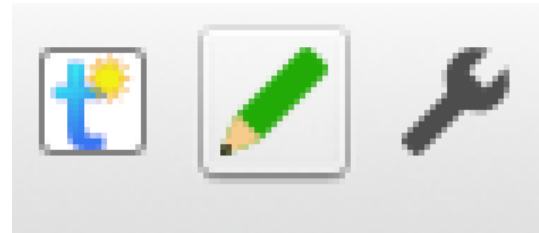


図2 提案システムの起動ボタン

い. また, 指定した座標の1点から調べる方法も, タッチした瞬間に何かしらの要素が選択されてしまうため, 目的の要素を指定するために何回も操作を繰り返す必要が生じる可能性がある.

## 1.2 本研究の目的

そこで, 本研究では前述の手法に代わり, ウェブページ中の要素を選択するための方法としてジェスチャを利用する方法を提案する. 以下では, この手法を実現するためのシステムとその実装について述べる.

## 2. システム概要

提案システムは, ウェブページ閲覧中にユーザが明示的に起動することで, 選択機能を提供する. 具体的には, ブラウザ上部に追加されるボタン (図2) をクリックすることで, 現在閲覧中のページにおいてシステムを起動する.

ユーザがシステムを起動させると, システムはウェブページ上にジェスチャを入力することを可能にする. このとき, ユーザがマウスもしくはタッチパネルを用いてジェスチャを入力することにより, システムが入力ジェスチャを解析して要素を1つ選択する (図3).

システムが選択するのは, 画面上に表示される, HTMLのタグ単位の要素を1つ選択する. つまり, ドラッグを用いる方法のように, 画面上の文字列の一部を選択するものではない.

## 3. 実装

### 3.1 概要

提案システムは, ブラウザの拡張機能として実装する. 専用ブラウザではなく, 拡張機能として

## 夏のプログラミング・シンポジウム「ビューティフル・インターフェイス」2014.8.24

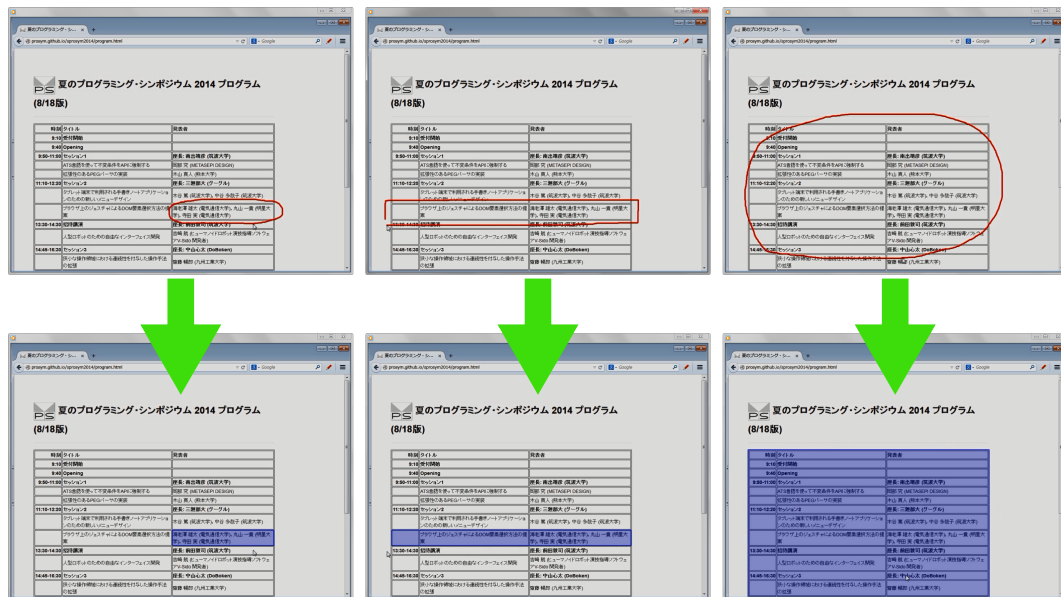


図3 ジェスチャによる選択。上側の赤線が入力ジェスチャ、下側の青い矩形が選択された要素を示す。ジェスチャの位置と大きさに応じて適切な要素が選択される。

実装した理由としては、ユーザが普段から使用しているブラウザを用いて広く利用できることを可能にするためである。

対象とするブラウザは Google Chrome<sup>\*1</sup> および Firefox<sup>\*2</sup> とした。

### 3.2 選択のための準備

ユーザが明示的な操作により、システムを起動すると、システムはジェスチャによる選択を可能にするための操作を行う。これは、現在閲覧中のページにスクリプトを挿入することで行う。スクリプトでは、ユーザがジェスチャを入力するための環境構築と、選択対象となりうる要素のデータを収集する処理を行う。

ユーザがジェスチャを入力できるようにするために、スクリプトはウェブページの最も前面に透明な要素を追加し、そこでのマウスイベントおよびタッチイベントを検知する。ユーザがジェスチャを入力し始めると、その軌跡を描画し、入力終了とともに入力ジェスチャの解析を行う。

\*1 <https://www.google.co.jp/chrome/browser/>

\*2 <https://www.mozilla.org/ja/firefox/new/>

選択対象となる要素は DOM ツリーを再帰的に辿り、body 要素の子孫ノードである可視状態の要素について、その位置を大きさなどを記録する。ジェスチャを解析し、要素を選択するときに、ここで記録したデータを参照する。

### 3.3 ジェスチャの認識

システムは、ユーザから入力されたジェスチャを解析し、ジェスチャの形状が円形や矩形などの閉じた図形であると判定した際に選択するようにした。

この、ジェスチャの形状解析に、\$I Recognizer[3][5] を利用した。このライブラリは、高速かつ正確にジェスチャの形状を解析することが可能であり、また利用が容易であるために使用することを決めた。

### 3.4 選択対象の推定

システムは、入力されたジェスチャの位置および大きさから、ユーザが意図したと思われる要素を1つ推定し、それを選択する。

選択対象となる要素は、要素を内包する最小の



図4 選択対象の列挙。リスト部分（横縞の緑の領域）とジェスチャ（縦縞の青の領域）は重なっている（十字の橙の領域）ので、選択対象となる。

矩形（このような領域を以下バウンディングボックスと呼称する）が、ジェスチャのバウンディングボックスと重なりあうものを対象とした（図4）。

通常、このような選択手法においては、入力された閉曲線に内包されるものを対象とする場合が多い。しかし、ウェブページにおける要素は、見た目と実際の領域の大きさに差がある場合がある。そのため、ユーザは内包するように入力したつもりだが、実際には内包されていないという事象が発生しやすいと考えられる。そこで、このようなユーザへの不利益を回避するため、対象範囲が広がるもののバウンディングボックスを使用することにした。

選択対象となる候補は、全て次に示す式に従ってスコアを計算し、その値が最も大きくなった要素を選択するようにした（図5）。

$$score = \frac{CR}{GR} \cdot \frac{CR}{ER}$$

$GR$  = ジェスチャの領域の面積

$ER$  = 要素の領域の面積

$CR$  = 共通領域の面積

ジェスチャと要素、それぞれの面積に対する共通領域の面積の割合の積を用いることにより、入力されたジェスチャとの中心距離に近い大きさの似た要素を選択することが可能である。

また、ブラウザの画面に収まらないような巨大な要素に関しては、要素の領域を画面内に表示された部分に限定してスコアを計算することにした。

図5 スコアの計算（図4の場合）

このような対処により、巨大な要素についてもジェスチャ操作で一度に選択できることを可能にした。

#### 4. 考察

提案手法には、次に述べるような利点があると考えられる。

まず、選択にジェスチャを利用しているため、タブレットやスマートフォンなどのタッチパネルを搭載したデバイスにおける親和性が挙げられる。特に、マウスカーソルなどが指す1点の座標から要素を選択する方法と比較した場合、提案手法の方がどの要素を選択したいかを明確に示することができる分有利であると思われる。

また、入力されたジェスチャの位置と大きさを考慮して選択を行うため、より見た目に近い形で要素を選択することができる。例えば、Google Chromeでは印刷範囲の設定のため、テキスト選択と同様のマウスドラッグによる操作でそれを行うことができる。しかし、ユーザが想定しているような範囲を設定するために、細かなマウス操作や選択操作のやり直しが必要になる場合がある。これは、マウス座標から決定されるキャレット位置を元に選択を行っているため、ユーザの認識と実際に選択される領域との間に誤差が生じるためである。そのため、ドラッグ開始時の位置が原因で大きな範囲が選ばれてしまうなどの選択ミスが発生してしまう。図1で示したマウスカーソルの指す座標から要素を選択する方法も、位置のみを基準に選択を行っているために同様の現象が起こる場合がある。それらと比較すると、提案手法では位置ではなく大きさも考慮して選択を行うため、このような事態の発生を軽減させることができる。

しかし、ジェスチャを使うことによって、次のような欠点もある。

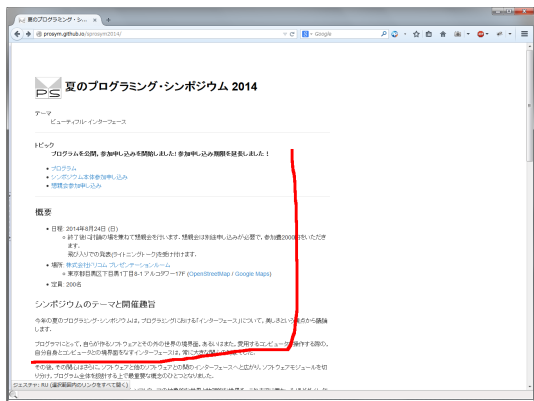


図 6 FireGestures

1つ目は、選択操作により時間を要してしまうことだ。マウスクリックやドラッグと異なり、明らかにジェスチャの入力には時間がかかる。また、ジェスチャの解析結果によっては入力のやり直しなども考えられるので、他の手法と比べて時間が必要となる。

また、ジェスチャを解析してから対象を推定して選択するため、実際に要素が選択されるまで、何が選択されるかをユーザは知ることができない。

## 5. 関連研究

Firefox のアドオン、FireGestures<sup>\*3</sup> (図 6) をはじめとする拡張機能は、ブラウザにおけるジェスチャ操作を提供するシステムとして一般的なものである。これらの拡張機能は、ユーザが入力したジェスチャに応じて、ページの遷移や更新などのブラウザが提供する機能呼び出すものである。

PDF ビューワとして有名な Adobe Reader<sup>\*4</sup> には、マーキーズームという機能がある (図 7)。これは、ユーザが画面内に矩形領域を設定することで、その領域が表示領域内で最大に表示されるよう画面を拡大する機能である。

寺田らは、閉曲線を利用して、拡大・縮小を行う手法を提案している [4]。閉曲線の中心が画面の中心となるようにし、閉曲線の大きさから拡大率を決定する。また、閉曲線の向きを元に、拡大操

<sup>\*3</sup> <https://addons.mozilla.org/ja/firefox/addon/firegestures/>

<sup>\*4</sup> <https://get.adobe.com/jp/reader/>



図 7 マーキーズーム 左:範囲の設定, 右:拡大後の画面



図 8 SUUMO アプリにおける物件検索の様子

作か縮小操作であるかを決定する。

不動産情報サイト「SUUMO<sup>\*5</sup>」がスマートフォン向けに提供するアプリには、地図上でユーザが閉曲線を入力することにより、その閉曲線内にある物件情報を表示する機能がある。図 8 はその機能を用いて物件を検索した様子を示す

Aitken らの特許では、ジェスチャを用いたウェブページデザインに係る特許において、投げ縄状の

<sup>\*5</sup> <http://suumo.jp/>

ジェスチャによって部品を選択する操作が挙げられている [1]. 本研究においては、投げ縄状のジェスチャに寄らない点、ウェブページ作成システムではなく閲覧中の一般のウェブページにおいて要素を選択するための手法である点において、この特許との差異があると言える。

## 6. まとめ

本研究では、ウェブページの要素を選択するための新たな手法として、ジェスチャを用いる方法を提案した。

入力に時間がかかるなど、いくつかの問題点があるものの、位置や大きさを考慮した柔軟が選択ができると考えられる。また、タッチデバイスなどにおいても有効に使うことができると思われる。

## 7. 今後の課題

現在、推定アルゴリズムの関係もあり、1つの要素しか選択することができない。そこで、新たなアルゴリズムも使用することにより、複数の要素を一度に選択可能にしていく。複数要素の選択を可能にすることで、現時点では行うことのできない表の列の選択など、より入力ジェスチャの見た目に応じた選択が可能になる。

また、より複雑な形状のジェスチャへの対応により、ジェスチャを用いる強みをより強化していきたい。

さらに、今は HTML のタグ単位という粒度で選択を行っているが、これを文節などのより細かい単位に対して行えるようにすることで、タッチデバイスにおける、文字列選択操作として活用できる可能性がある。

最後に、この選択方法をブラウザに限らず、大きな要素を持つ要素を選択するための手法として提案したい。特に、地図アプリケーションなどでは、ある特定の地域を選択しそこを拡大表示するために利用できるのではないかと考えられる。

## 参考文献

- [1] D. E. Aitken, C. H. Jackson, R. J. Aitken, J. B. Hoecherl: “Gesture-based web site design”. US Patent App. 12/651,941. Jul. 7, 2011.
- [2] Kerry Shih-Ping Chang, Brad A. Myers: “WebCrystal: understanding and reusing examples in web authoring”. CHI '12, pp. 3205-3214.
- [3] Yang Li: “Protractor: a fast and accurate gesture recognizer”. CHI '10, pp. 2169-2172.
- [4] 寺田実, 武井英人: “対称性に着目した手書きズーム方式の提案”. WISS 2005.
- [5] Jacob O. Wobbrock, Andrew D. Wilson, Yang Li: “Gestures without libraries, toolkits or training: a \$1 recognizer for user interface prototypes”. UIST '07, pp. 159-168.

## 質疑・応答

八木原 (キヤノンソフトウェア) ターゲットユーザをどのように想定しているか。また、HTML の知識のない人が上手く利用できるのか。

海老澤 個人的にはすべての人に使ってもらいたい。主な用途としては、HTML の知識を持つ人を前提としたものを挙げた。

八木原 表の列などは (HTML の構造として 1 つのタグで囲まれていないため) 縦にまとめて選択できない、と初心者になってしまうだろうが、そのあたりをどのように考えているか。

海老澤 今後、1 つのタグで囲まれていない複数の要素の同時選択を可能にすることで対応したい。

---

平石 (京大) 囲む操作は面倒だと思うので、操作の始点と終点から決定される矩形を用いた選択が良いと思う。ユーザがジェスチャを使う必要あるほどに複雑な形状で選択したいと思うのか。

海老澤 複雑な形状の例としては、あるレベルのリスト項目のみを選択するという操作が考えられる。

---

横山 (東大) デモで画面外に広がる大きな要素も選択できたのはなぜか。

海老澤 重み計算において、計算に用いる要素の大きさを画面内に限定して計算している。

横山 要素の大きさを画面内に限定することによって、悪影響はないのか。

海老澤 悪影響については未確認。むしろ、2カラムデザインのサイトにおいて、1カラム部分だけを選択したいとき、このようにすることでもうまく選択することが可能になる。

横山 面積によるスコアリングで、ユーザの意図をもう少し反映できないか。

---

河内谷 (日本 IBM) 表が巨大だと、画面全体を囲んでも表全体ではなく、ある1行のみの選択にならないだろうか。対象が巨大な場合のみに有効となるようなルールを作るのはどうか。ところで、複数の要素を選ぶときに、現行のスコア計算の自然な拡張でうまくいくのか。

海老澤 現行の方法のままでは、恐らくうまくいかない。単一選択と複数選択とで、スコア計算の方法を変更する必要があると思う。

---

大島 (東大) 選択にはジェスチャではなく矩形を用いる方がよいと思う。マウスで図形を書くのは難しいし、スコアリングに用いているバウンディングボックスはそもそも矩形であるのだから、矩形で選択するのが自然だ。

---

平石 リアルタイムに入力中のジェスチャから決定される選択対象を表示させた場合の反応性はどうか。

海老澤 そのような操作を検証していないので不明である。