

# 大画面ディスプレイ上でのウェブブラウジングにおける 効率的タスク切り替え支援手法

劉 瀟巍† 田島 敬史††

† 京都大学工学部情報学科

†† 京都大学情報学研究科

E-mail: †liugi@dl.kuis.kyoto-u.ac.jp, ††tajima@i.kyoto-u.ac.jp

**あらまし** 近年、インターネットの普及と情報の爆発的な増加により、計算機ユーザはほとんどの情報をウェブ上で検索、閲覧、処理、交換するようになってきている。その結果、ユーザはウェブブラウザで多数のタスクを同時に遂行するようになり、ウェブブラウザがタスク管理において重要な役割を担うようになってきている。しかし、従来のウェブブラウザは多数のタスクからタスクを簡単に発見、選択するための機構を提供していない。一方、大画面ディスプレイが普及しつつあるが、大画面上の余分なスペースはタスク管理で有効に利用されていない。本研究は大画面ディスプレイ上でのウェブブラウジングにおける効率的なタスク切替え支援手法を提案する。まず、開かれているウェブページのサムネイルを大画面ディスプレイ上の周辺スペースに常時表示する。さらに、各ページにあるロゴと特徴的な画像を自動抽出し、サムネイル上に重ねて表示する。また切り替えるタスクのサムネイルを探す負荷を低減するために、切替え履歴を基に各ページへの切り替える確率を予測し、確率が高いページのサムネイルをサイズまたは透明度で強調する。本研究の提案手法の有効性を定量的実験と定性的実験により検証した。

**キーワード** ユーザインターフェース、ウェブブラウザ、タスク管理、サムネイル

## 1. はじめに

近年のウェブにおける著しい変化により、計算機上での日常作業におけるウェブブラウザの役割が大きく変わってきている。その変化は主に三つある。まず、検索エンジンによるウェブ検索が、知的労働においてますます重要になっており、調査[1]によると、大半のユーザは毎日何回も検索エンジンを使用している。情報図書館やペーパーレスオフィスなどの発展に伴い、今後ますます多くの情報がウェブ上で検索できるようになっていくと予想される。次に、ユーザはウェブ上の情報を閲覧するだけでなく、ウェブ上で情報の処理を行うようになってきている。ウェブはその誕生時には単純な閲覧むけのアーキテクチャであったが、近年AJAXやRIAなどの技術によりウェブ上でも多彩なアプリケーションを構築できるようになり、Emailや文書処理などの様々な伝統的なアプリケーションがウェブ上に移りつつある。これらのアプリケーションのウェブ上へのシフトは、ユーザにとっては、データの集中管理によりどこでも同じ環境が再現されデータ損失の心配もないという利点がある。また、従来ネイティブなアプリケーションが支配していたビジネス市場でもクラウドコンピューティング(ユーザが様々なデータ処理機能をネットワーク(通常はインターネット)経由でサービスとして利用する新しいコンピュータの利用形態)の発展により、ウェブベースのアプリケーションが大きな役割を果たすことが期待されている。最後に、ウェブはユーザが情報を交換するプラットフォームにもなっている。近年、多数の一般ユーザによる情報の生成と共有というWeb2.0の考え方にしたがって、ウェブ上でコミュニケーションとコラボレーション

を支援するサービスが多数、登場している。特に、ウェブメールをはじめ、ブログやソーシャルネットワークサービスやマイクロブログなどのサービスは既に広く利用されるようになってきている。このようなサービスは日常のコミュニケーションだけではなく、今後ビジネスと教育においても大きい役割を果たすことが見込まれている。

これらの変化により、ユーザはウェブブラウザ上で同時に多数のタスクを遂行するようになってきている[1]。また大画面ディスプレイを使用するユーザはそうでないユーザより多数のウィンドウを開く傾向が観察されている[2]。しかし、従来のタブブラウザではタブの数がある程度多くなると、タブが切り替えにくくなる。その原因は主に三つある。まず、タブが小さくなると、タイトルの最初の部分だけが表示され区別できなくなる場合がよくある。また、タブが多くなると、長いタブのリストを走査して切り替えるべきタブを探すのが難しくなる。三番目に、タブが小さくなるとポインティングデバイスでクリックしにくくなる。その他にも、タブブラウザにマルチバーチャルワークスペースのようなタスク管理機構がないため、ユーザのタスク管理の負担が高いなどの問題がある。

一方、大画面ディスプレイが安価になり普及しつつあるが、ウェブページを大画面上で表示すると左右に大きな余白が現れることが多く、これらのスペースがウェブ閲覧において有効に利用されていない。そこで本研究では、このスペースを利用して効率的なタスク管理を支援することを目指す。まず、小さいタブに代わって、開かれている全ページのサムネイルを常時、このスペースに表示する(図1)。これは今までのタスク管理システムとは大きく異なるタスクの表示方法だが、これは、人々



図1 本システム



図2 サムネイル



図3 拡張サムネイル

が現在のタスクに関連する文書を机の上に並べ、スペースが足りない場合は、GUIシステムでの最小化のように、あまり使われない文書を重ねていたのと同様の表示であり、人間にとって自然なものである。また、人間の高い視覚的認知能力を生かしてサムネイル(図2)をより認識しやすいように、ウェブページにあるロゴとそのページのもっとも特徴的な画像を自動的に抽出し、サムネイルに重ねて表示する(図3)。

しかし、同時に開くページ数が増えると、ユーザが多数のサムネイルを走査しなければならない問題がやはり生じる。そこで、本研究ではユーザの切替え履歴に基づいて、次に各タブに切り替える確率を予測し、確率が高いタブのサムネイルを強調する。強調方法としては、確率が高いサムネイルをより大きく表示する動的サイズモードと、確率が高いタブのサムネイルをより不透明にする動的透明度モードの二つのオプションを用意する。このような強調表示によってユーザの注意を絞ることで走査するタブの実勢数が減ることが期待できる。

これらの手法の有効性を確認するために二つのユーザ実験を行った。実験1では、タスク数が7個の場合と15個の場合について、タブ、サムネイル、拡張サムネイルの各方式でタスク切替えにかかる時間を計測する。実験2では、被験者に本システムのプロトタイプを使って簡単なマルチタスクを遂行してもらった後、半構造化インタビューを行う。実験1の結果より、拡張サムネイルの優位性を確認した。また、人間の空間記憶のタスク切替えにおける役割についても分析した。実験2では、本システムのユーザにとって満足な点と不満な点が示された。

以下、本論文の構成を簡単に説明する。まず、研究の背景と、なぜウェブブラウザでの効率的なタスク管理が重要かについて述べる。次に関連研究をまとめる。その後、従来のタスク切替え機構の問題点を分析し、本研究ではどのように解決するかを説明する。続いて、拡張サムネイルと切替え確率の計算につい

て詳しく述べる。次に、インターフェースと実装における問題と工夫を説明し、続いて、実験とその結果を示す。最後に実験結果の考察を行い、結論を述べる。

## 2. 関連研究

マルチタスクにおけるタスク切替えの支援の必要性とその心理学的側面についてはこれまでに多数の研究がある。Miyata と Norman はユーザ活動の心理学的側面とウィンドウマネジャーにおけるマルチタスク支援の必要性について論じている [3]。また、Cioni はユーザ活動の階層構造を提唱し、複数の行動を同時に行うための支援機構の必要性を指摘している [4]。

一方、González と Mark は Working Sphere を提唱し、Working Sphere の保存と復旧の重要性について論じた [5]。タスク間切替えの困難性については [6] でも検証されている。Czerwinski は 11 名の Microsoft Windows の熟練ユーザに、現実の業務中のタスク切替えの履歴を記録してもらい、日常タスクにおけるタスク切替えと割り込みの様子を調べた。その結果、割り込まれた後にタスクへ戻る切替えが特に難しいことがわかった。

Rooms [7] は画面上の表示面積不足を緩和するためにマルチ仮想ワークスペースを導入する研究の先駆である。Rooms では仮想ワークスペースは部屋のメタファーで表現され、ユーザは手でタスクを仮想ワークスペースに割り当てることができ、タスクのコピーや移動も可能である。また、「window thrashing」という言葉を用いて、表示面積不足の問題をメモリ管理におけるワーキングセットモデル [8] と関連付けて捉えていた。

GroupBar [2] は Microsoft Windows のタスクバー上でユーザがドラッグ・アンド・ドロップでタスクをグループ化できるようにする研究で、プレビューや自動レイアウトなどの機能も提供されており、実験により、通常のタスクバーと比べてユーザのマルチタスクの効率を向上することが確認されている。

Ishak らはタスク管理における Content-Aware Layout の概念を提案した [9]。Ishak らのシステムでは、ユーザがテキストエディタ中のキーワードを選択してキーボードショートカットキーを押すと、選択されたキーワードを含むウィンドウだけが、キーワードの位置を中心に合わせたレイアウトで表示される。これはタスク管理におけるコンテンツアウェアネスの応用例だが、テキストエディタでしか使用できないという制限がある。

市販オペレーティングシステムでも様々なタスク管理の工夫がなされている。Microsoft Windows Vista では、3D でタスクのサムネイルを並べるタスク切替え機能が導入され、Mac OSX ではタスクのサムネイルをタイル状に並べて一覧表示する機能とマルチ仮想ワークスペースの機能が提供されている。

しかし、これまでの研究やシステムでは、いくつかの重要な問題点がまだ十分に考慮されていない。まず、タスク切替えボタンの視認性を向上させる工夫がないため、タスクが多くなると選択が困難になる。二番目の問題はワークスペース管理のためのユーザ負荷の増加である。そのため、タスクがある程度以上多くなるとユーザが負担を負い切れなくなり、ワークスペースでの管理をあきらめてしまう。三番目に、表示面積が限られていることに着目する研究は多いが、表示面積が十分にある場

合に、いかにタスク管理のために有効利用するかに関する研究は少ない。四番目の問題は、ユーザの切替え履歴と内容の関連性のタスク管理における潜在力があまり重視されていない。

### 3. 問題分析

本章では問題の分析を行うが、具体的な説明に入る前に、まず以下の二つの用語について定義しておく。

**タスク** タスク管理の最小単位。本研究ではウェブブラウザを想定しているので、ブラウザで開かれた各ページがタスク。

**ワーキングセット** ユーザが一つの目標のために開いているタスクの集合。[4]で定義された activity に近い概念である。

また、既存のタスク管理システムにおいて、タスクを切り替えるためのボタンの表現としては、大きくわけて三種類ある。

**タイトル(またはタイトルに加え小さなアイコン)** Microsoft Windows のタスクバーやタブブラウザのタブバーなどで採用。

**サムネイル** Mac OS X の Exposé と Windows Vista の Aero で採用。

**アイコン** Mac OS X のドックと Windows 7 のタスクバーは大きいアイコンだけを表示。

以下、本章では、ユーザのタスク切替えの効率低下の原因となっているいくつかの問題について順に説明していく。

#### 3.1 視認性の問題

上述の、タスク切り替えの三種類のボタンについてそれぞれ視認性の問題を分析する。

**タイトル(またはタイトルに加え小アイコン)** タスク数が増えると、通常、タイトルが一部しか表示できなくなる。その場合、もし違うタスクの表示される部分タイトルが、各タイトルの特異部分ではなく共通部分になると、その部分タイトルではタスクを区別できなくなり、ユーザはマウスをタイトル上に移動してツールチップを見たり、それらのタブの順番を覚えたりしなければならない。特にウェブページのタイトルのユーザビリティはばらつきが大きく、このような問題が頻発する。

**サムネイル** サムネイルの場合も、ウィンドウ中の閲覧部がテキストのみの場合などは、よく似たサムネイルになり区別しにくくなる。また、サムネイルはウィンドウ全体を均一に縮小するため、重要部分とそうでない部分が同じように見にくくなる。

**アイコン** アイコンの場合も同じアイコンを持つタスクが区別できなくなる。Mac OS X のアプリケーションウィンドウ切替え機能では、そのような場合、二回の操作が必要になる。

視認性の問題を解決するために、本研究ではサムネイルをより高機能にする。具体的には、ウェブページの構造を利用して、ページ内にあるロゴと特徴画像を抽出し、これらをサムネイル上に重ねて表示することで、視認性の向上が期待できる。

#### 3.2 走査の問題

タスク数が多くなると、タスク切替えの際に多数のアイコンやサムネイルを走査しなければならない。一般に人間は空間上の配置を記憶する能力に優れているため、多くのアイコンやサムネイルの位置を覚えることもある程度は可能だが、この記憶を維持する労力はユーザの負担となる。また、このような記憶はタスクの遂行に伴い徐々に消えてしまう危険がある。この走

査の問題はマルチ仮想ワークスペースである程度解消できるが、一方、ワークスペースの管理が新たな負担となってしまう。

そこで、本研究では次に各タスクに切り替える確率を計算し、確率が高いタスクのサムネイルを強調することでユーザが走査するサムネイルの実勢数を減らすことを考える。タスクの強調方法としては、二種類の方法について検討を行った。

**サイズ** 確率が高いタスクほどサムネイルを大きくする。

**透明度** 確率が高いタスクのサムネイルを不透明にして、確率が低いタスクのサムネイルを透明にする。

#### 3.3 関連するタスクの短期記憶への復旧の問題

現在のユーザ、特にビジネスユーザは、タスク執行において割込みが頻繁に起こり、割込みと割込みの間のタスクへの集中時間が非常に短いということが観察されている [5], [6]。ユーザが割込み後に当初のタスクに戻る場合、そのタスクに関わる情報を再度短期記憶にロードする必要がある。本研究では自動的に相互に関連するタスクの集合を計算することで、ユーザに負担をかけずに、スムーズなタスク復旧を支援する。

#### 3.4 切替えボタンの問題

マウス等のポインティングデバイスでオブジェクトを選択するのにかかる時間を予測するモデルとして、Fitts Law が知られている [10]。二次元での Fitts' Law として、本研究では次式によって定義される SMALLER-OF 方法 [11] を採用する:

$$MT = a + b \log_2(A/\min(W, H) + 1)$$

ここで、各変数は以下の値を表す。

$MT$ : 所定オブジェクトに移動するのにかかる時間

$A$ : ポインタからオブジェクトの中心への距離

$W, H$ : オブジェクトの幅と高さ

$a, b$ : ポインティングデバイスによって決まる定数

小さなタブやタスクアイコンを用いるシステムでは、Fitts' Law によりタスク選択に時間がかかる。この問題は、タスク数が増えるとタブ等が小さくなるシステムでは特に著しくなる。本研究は大画面上の余ったスペースに大きなサムネイルを表示することで、この問題を解決する。

#### 3.5 ウェブページ自体の情報

ウェブ閲覧に焦点を置く場合、ウェブ自体の特徴にも注目する必要がある。まず、ウェブページ間にはリンクが存在する。また、従来の多くのアプリケーションが専用データ型式を用いるのに対し、ウェブページは標準的型式で記述されており、ページ中のテキストや画像などの抽出、利用が容易である。

### 4. サムネイル

この章ではサムネイルの利点と欠点についてさらに分析し、本研究で提案する拡張サムネイルについて説明する。

#### 4.1 サムネイルの利点

人間はテキストよりビジュアルな情報の方が、覚えたり識別したりする能力が圧倒的に高いことが知られている。そのため、サムネイルはこれまで広くユーザインターフェースで有用な道



図4 サムネイル



図5 テキスト部のサムネイル



図6 拡張サムネイル



図7 ログと特徴画像による拡張サムネイル

具として使用されてきた。例えば、Mac OS X の Exposé はサムネイルを並べて一覧表示することでタスク切替えを支援する。

サムネイルに関する研究についてもこれまでに多くのものがある。Cockburn らはサムネイルの文書閲覧における有効性を検証している [12]。また、Woodruff らはウェブ検索結果のサムネイルを生成し、これにテキスト概要を重ねて表示することで、検索過程を支援する方式を提案している [13]。

また、サムネイルはタスク管理システムにおいて、選択しやすいメリットもある。Fitts Law によって、細長いタブよりサムネイルの方がより選択しやすいことが期待できる。

#### 4.2 サムネイルの欠点

状況によってはサムネイルにも欠点がある。例えば、論文のような一ページ目がテキストのみのページはサムネイルで区別できなくなる。あるいは、Wikipedia 等のページでは、ページ冒頭に特徴的な写真等があり後半はテキスト中心になる場合が多い(図4)ため、ユーザがテキスト部分を閲覧しているとサムネイルが識別しにくくなる(図5)。このように、サムネイルは必ずしも常に有用ではない。その原因は二つある。まず、サムネイルは通常、ユーザがその時に閲覧している部分から作られる。しかし、最もユーザの印象に残っている部分は、必ずしも現在閲覧中の部分とは限らない。また、サムネイルはページ全体を均等に縮小するため、重要な部分も重要でない部分も同じ比率で縮小してしまう。

#### 4.3 拡張サムネイルによる解決

サムネイルの利点は損なわずに、その欠点を解消するため、本研究では、各ページの特徴的な画像を常時サムネイルの上に重ねて大きく表示する方法を提案する(図6)。これにより、ページの特徴的な部分がユーザの閲覧部分によらず常に、かつ、大きく表示されることになり、上述の二つの問題が解決される。

しかし、同じ特徴画像を持つページも存在する。例えば、iPod を買おうとする人は同じ iPod の画像を特徴画像として持つ Amazon のページと楽天のページを同時に開くようなことがよくある。このような場合、ページ中にある各サイトのロゴを表示すると各サイトを区別しやすくなる。一方、ロゴだけを表示するのでは、同じニュースサイトで複数のニュースを閲覧する場合等に識別しにくくなってしまふ。そこで本研究では、ロゴと特徴画像を両方重ねることで、ある一つのものについて色々なサイトのページを開く場合と、ある一つのサイトで色々なページを開く場合の双方に対応する。これにより、ウェブ閲

覧における大抵のページの組合せに対応できる。また、ページの閲覧部分がわかりにくくなることを防ぐため、本研究では特徴画像をサムネイルの角に配置する(図7)。

##### 4.3.1 ログと特徴画像の抽出

本研究では、ウェブページ中の要素の下記の属性に関するヒューリスティックを用いて、ページ中のロゴと特徴画像を抽出する。判定に用いる重みや閾値等は観察と実験により決定しているが、今後、多数のページの正解データを作成し、機械学習を用いて、より精度が高い判定方法を開発する予定である。

**位置** 要素の左上角のページ左上角からのオフセット、画面サイズによってページ表示のマージンが大きく変わる場合があるので、マージンの部分は含めないものとする。

**サイズ** 要素の面積。

**アスペクト** 要素の幅と縦の比率。

**タグタイプ** 要素のタグの種類。

**特徴文字列** 要素の CSS クラス名や a タグの href 属性や img タグの src 属性を連結した文字列。

ページ中の各要素に関して、下記の方法でヒューリスティック値を計算し、値が閾値以上かつ最大となるの要素を選択する。

##### • ログ抽出

**場所** 左上または中央上部にある要素を優先

**サイズ** 500 以上、かつ 40000 以下

**アスペクト** 1 以上、かつ 8 以下

**タグタイプ** img タグと a タグを優先

**特徴文字列** ad, ads, banner を含まない要素と logo を含む要素を優先

##### • 特徴画像抽出

**場所** ページ上部から 80 ピクセル以上離れていて、かつページの上部にある。

**サイズ** 8000 以上、かつ 1000000 以下

**アスペクト** 1/6 以上、かつ 6 以下

**タグタイプ** img タグが優先

**特徴文字列** ad または banner を含まない要素

##### 4.3.2 抽出精度データ

Alexa.com が公開しているトラフィックベースのグローバルのウェブサイトアクセスランキングの上位 100 個を使用して上記抽出手法の精度を測定した結果を表1に示す。ただし、上位 100 件中の不適切なサイトと重複するサイト (google.com と

表 1 ロゴと特徴画像の抽出精度

	ロゴ	特徴画像
総数	62	62
除外数	0	22
精度	83%	76%

google.jp など) は除外し、また、日本で普段よく使われるサイトからいくつか典型的なものを加えた。各サイトから一つまたは二つ代表的なページを選びテスト集合としている。正解は人間が手動で判定した。その際、正解となる特徴画像を持たなかったために除外したページ数を「除外数」の行に示す。

## 5. 切替えの予測

本研究ではマルチ仮想ワークスペースのような手動タスク管理の代わりに、切り替える確率が高いと予測されるタスクのサムネイルを強調することによる、自動的なタスク管理を実現する。本章では切替え確率の予測について説明し、具体的な表示については 6. 章で説明する。

まず、いくつかの記号を定義する。ユーザの過去のタスク切替え履歴を  $s(a_1, a_2, 1), s(a_2, a_3, 2), \dots$  のような記法で記述する。  $s(a, b, t)$  はユーザが  $t$  回目の切替えでタスクを  $a$  から  $b$  へ切り替えたことを示す。また、履歴中に現れる  $s(a, b, t)$  と  $s(b, a, t)$  の総数を  $sc(a, b)$  で表す。本研究では、タスク間の関連の強さを用いて予測を行うため、 $a$  から  $b$  への切替えと  $b$  から  $a$  への切替えを区別しない。タスク間の関係に方向性がある場合もありうるが、操作の間違いなどで逆の順番で戻る操作も頻繁に行われるため、方向を区別する利点は小さいと考え、ここでは方向性を無視することで計算を単純化する。

また、ユーザが想定するワーキングセットを正確に予測することは困難なため、ここではワーキングセットをその要素と非要素が二値的に決定される集合として定義する代わりに、タスク間の関連度によって暗黙的に曖昧なワーキングセットを考え、後の強調表示も二値的ではなく連続的に行う。そのため、一つのタスクが複数のワーキングセットに属する場合もある。

### 5.1 切替え確率の計算

総タスク数が  $n$  である時、各タスクに識別子として  $1 \dots n$  を割り当て、履歴中でのタスク  $i$  から  $j$  に切り替える確率  $p_{ij}$  を以下で近似することにする。

$$p_{ij} = p_{ji} = \frac{sc(i, j)}{|pre(i) \cup post(i) \cup pre(j) \cup post(j)|}$$

ここで、 $pre(i) = \{s(j, i, t)\}$ 、 $post(i) = \{s(i, j, t)\}$  である。

次に、マルコフチェーンモデルを利用して予測切替え確率を求める。  $p_{ij}$  を  $ij$  要素とする  $n \times n$  行列を  $p$  とし、最終的な予測切替え確率の行列  $pr$  を

$$pr = p * (1 - \epsilon) + p * p * \epsilon$$

で求める。ここで  $\epsilon$  は履歴の中では二回の切替えで遷移した先に一回で遷移する確率である。  $p * p$  の項を加えるのは、履歴がまだ十分収集できていないために関連するタスク間に切替え履歴が存在しない場合でも、二回の切替えでの遷移先まで含めて

考えることにより、タスク間の関連をより広く補足するためである。また、検索結果ページやポータルページのようなハブとなるページから複数のリンク先を開くような場合、そのハブから開かれたページ同士は関連があるものの、互いに直接切り替えたことがない場合が多いが、そのような場合も、 $p * p$  を加えることでこのようなページ間の関連を補足できる。

### 5.2 タスク進化

ユーザがタスクを遂行する際、タスク内容が変化していくことがよくある。例えば、一つの複雑なタスクが処理しやすいように複数のサブタスクに分割されたり、逆に、サブタスクが進んで全体的な複雑さが低下すると一つのタスクに再度、合併されたりする。また、タスクの当初の目標がタスクの進行にともしない変わっていくこともある。

一方、[7] で示されていたように、多数のタスク間での切替えの問題は、メモリ管理におけるページングの問題に類似している。そこで、この連想に従って、タスク変化の問題の解決策として Denning のワーキングセットモデル [8] を考慮してみた。Denning らの研究では固定サイズのスライドウィンドウを使用してワーキングセットを計算している。しかし、この方法をそのままタスク管理に適用すると、ユーザが長い時間をかけて一つのワーキングセットを遂行する場合、そのワーキングセット内のタスク切替えがスライドウィンドウを埋め尽くし、他のワーキングセット内の切替え履歴が全部消えてしまう。この問題を解消するため、全切替え履歴にスライドウィンドウを適応するのではなく、各タスクペアの間の切替え履歴に適応する。すなわち、各タスクペア  $(a, b)$  における  $\{s(a, b, t)\}$  の集合にスライドウィンドウを適応する。

### 5.3 リンク効果

ウェブページは相互にリンクによって繋がられていて、ユーザはほとんどの場合リンクを通して、他のページへと遷移する [1]。そこで、リンクによる暗黙の情報を利用するために、本研究ではリンクで新しく開かれたページを特別に扱う。具体的な扱い方を説明する前に、まずいくつかのウェブブラウジングによくある切替えパターンを分析する。Weinreich の研究 [1] によると、ユーザはいくつかの共通の目的でページを新しいタブやウィンドウで開く。例えば、ユーザは検索結果を確認する時に、素早く戻れるために新しいウィンドウでその結果を開く。この場合、新しいタブで開かれた検索結果が望ましいにせよ、そうでないにせよ、ユーザは他の結果を見るためか検索結果のページを閉じるために、検索結果のページに戻る。また、ユーザが時間を節約するために、一つのハブページにある興味を持つページを全部新しいタブで開く場合があるが、この場合も、ユーザはハブページに戻る。また、そもそもリンクが新しいタブで開かれるようにページ中で定義されていることもよくある。よって、本研究では A タブにあるリンクから B タブを開いた場合には A と B が同じワーキングセットに属すると仮定する。

### 5.4 少数ページ間の往復動作

また、二三個のタスクの間で頻繁に切り替えたい状況がよくある。このような状況を支援するため、本研究では直前二個のタスクの予測確率にボーナスファクターを与える。



図 8 動的サイズモード



図 9 動的透明度モード

## 6. ユーザーインターフェース

前章で述べた方法で各タブに切り替える確率を計算し、確率が高いタブのサムネイルを強調する。本研究では二つの強調モードを用意した。

**動的サイズモード** 動的サイズモードでは確率が高いタブのサムネイルが大きく表示され、確率が低いタブのサムネイルが小さく表示される(図8)。これには二つの利点がある。まず、確率が高いタスクのサムネイルを大きく表示することでユーザの注意を絞る、余分なスキャンを避けることができる。さらに、大きく表示することでマウスで選択しやすくなる。

しかし、サムネイルのサイズが変化すると表示の位置も変わってしまい、ユーザの空間的記憶に残っているサムネイルの位置が無効になってしまう。この問題を緩和するため、サムネイルのサイズが変わる際にアニメーションを用いた。これにより、ユーザはタスクがどこに移動したかを感知しやすくなる。

**動的透明度モード** 動的透明度モードでは確率が高いタスクのサムネイルがより不透明に表示され、確率が低いタスクのサムネイルがより透明に表示される(図9)。動的サイズモードと同じように、不透明と透明の違いでユーザの注意をより確率が高いタスクのサムネイルに絞ることが期待できる。

## 7. 実験

本研究では提案するシステムのプロトタイプを Firefox の拡張として実装し、提案システムを評価するために、二つの実験を行った。実験1は定量的に本システムにおける各要素の影響を計る。実験2はユーザに簡単なマルチタスクを遂行してもらい、

半構造化インタビューを行う。実験1と実験2はともに普段タブブラウザを使い慣れた8人の被験者によって行った。実験環境はCPUが2.53GHzのMacbook Proと解像度1920×1200の24インチ大画面ディスプレイとロジクールのマウスからなっている。実験は全てFirefox中で行うので、WindowsユーザがMacに不慣れなことによる問題はないと考える。サムネイルを表示する領域のサイズは全ユーザで統一した。

### 7.1 実験 1

実験1ではタスク数が7の場合と15の場合について、タブ方式、均一サイズサムネイル方式、均一サイズ拡張サムネイル方式の三方式でのタスク切替えにかかる時間を計測した。3×2=6通りの各実験項目毎に、被験者に50回のタスク切替えを行ってもらい、次の項目を始める前に約一分間の休憩を取る。切り替えるべきタスクはランダムに選ばれ、ページ中央にそのタイトルを表示し、ユーザが正しく選択できると次へ進む。ユーザが間違った選択をした場合は、メッセージを表示し、再度、選択させる。いずれの場合も、次の選択を始める前に、一旦、マウスカーソルを画面中央に移動させた。

タスク数7は人間の短期記憶の平均容量であり、ユーザがタブまたはサムネイルの位置をすぐに全て覚えられる場合に、サムネイルと拡張サムネイルが有効かを調べるものである。一方、15はユーザが全ての位置を覚えられない場合に対応する。

特定のページを覚えてしまう影響を避けるために、7タスクと15タスクは異なるページセットを用意した。また、拡張サムネイルの効果を測るため、ロゴと特徴画像が正しく抽出されたページを選んだ。さらに、Googleの検索結果のページは頻繁に利用される重要なページなので、異なるキーワードの検索結果ページを両方のセットに含めた。また、英語力の個人差の影響を避けるために、全てのページを普段日本人がよく見るサイトから選択した。実験の順番の影響を避けるために、ユーザごとに6つの実験項目の順番をシャッフルし、また、タブの場所を覚えてしまうことを防ぐために、なるべく7タスクと15タスクを交替で行うようにシャッフルした。実験を始める前には、各ページのタイトルからそのページを思い出せるように被験者に二つのページセットに目を通してもらった。

実験結果から8秒以上または0.8秒以下かかったものは除外した。サムネイルや拡張サムネイルにより、ユーザが切り替えたいタスクを探す走査時間が短縮されているかを確認するため、切替えにかかった時間から、[11]による以下の式を用いて推定したマウス移動時間を差し引いたものを走査時間として比較した。

$$MT = 230 + 166 \log_2(A/\min(W, H) + 1)$$

**タブ vs サムネイル** 図10と表2で示すように、タブのかわりにサムネイルを使用することでタスク切替えの効率が向上している。t検定により両者の間に有意な差があることを確認した(7タブ:t=7.225, p=1.183e-12, 15タブ:t=2.8493, p=0.004496)。走査時間(図11と表2)を見ると、7タブでの走査時間はタブがサムネイルより短く、拡張サムネイルより長い。15タブではタブの走査時間がサムネイルより長いことがわかる。これは、普段タブを使い慣れているユーザにとっては、

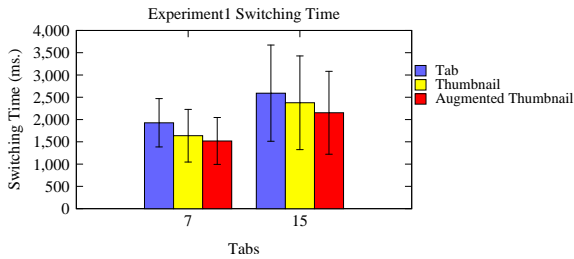


図 10 切替え時間

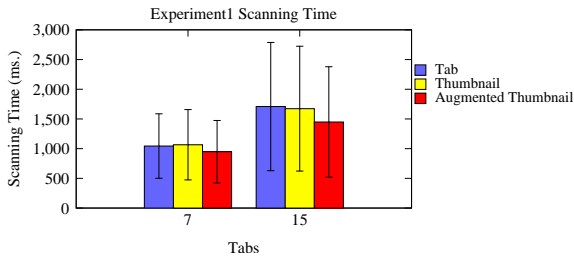


図 11 走査時間

表 2 実験結果 1 ( $\mu$ =平均,  $\sigma$ =標準偏差)

	切替え時間				走査時間			
	7 タブ		15 タブ		7 タブ		15 タブ	
	$\mu$	$\sigma$	$\mu$	$\sigma$	$\mu$	$\sigma$	$\mu$	$\sigma$
タブ	1928	542	2592	1079	1044	542	1708	1079
サムネイル	1637	591	2376	1051	1065	591	1673	1051
拡張サムネイル	1520	525	2152	929	948	525	1448	929

タブ数が少ない時にはタブとサムネイルの探しやすさが変わらないためと思われる。

**サムネイル vs 拡張サムネイル** 図 10 と表 2 で示すように、拡張サムネイルはサムネイルよりタスク切替えが早くなる。t 検定により、両者の間に有意な差があることを確認した (7 タブ:t=2.9629, p=0.00314 15 タブ:t=3.1849, p=0.001505)。7 タブの場合、被験者がほぼ場所を覚えられるので、拡張サムネイルとサムネイルでは差が出ないことが予想されたが、事件結果では差が見られた。15 タブの場合、タブ数が多いので拡張サムネイルの優位性がより強く現れており、サムネイルは拡張サムネイルより走査時間が 15% 長い。

また、注目すべき点として、走査時間がタスク数にしたがって線形的に増えていない。この理由としては人間の空間記憶の効果が予想される。今後、毎回サムネイルの場所が変わる場合の走査時間についても調査する予定である。

## 7.2 実験 2

実験 2 では被験者に本システムのプロトタイプを使用してマルチタスクを遂行してもらう。切替え確率の予測精度の測定と、ワーキングセット間の切替えの使い勝手を調査するために、四つのワーキングセットを用意した。

- Macbook と Macbook Air の仕様を比較しながら、それぞれに対する口コミを閲覧する
- オバマ大統領の経歴や紹介ページから情報収集する
- 京都大学の有名人四名のプロフィールを閲覧し経歴上の相違点を調べる
- 「オリンピック 2008」を Google で検索し、上位結果を閲覧する

最初の三つのワーキングセットはあらかじめ開いておいたウェブページを使ってもらう。最後のワーキングセットはユーザーに自由に新しいページを開いてもらう。実世界の割込み多発によるワーキングセット間の切替えをシミュレートするために、被験者に 2 分毎に「割り込み」をして他のワーキングセットへの切替えを指示し、全ワーキングセットを三周したら終了である。

動的サイズモードと動的透明度モードの比較を行うために、一回目はウォームアップとしていずれかのモードで遂行し、その後の二回はそれぞれ動的サイズモードと動的透明度モードを使用する。順番の影響を取り除くため、使用するモードの順番はシャッフルする。実験終了後、被験者にアンケートを記入してもらい、最後に本システムに対する満足点、不満点、要望に関するインタビューを行う。

ユーザーアンケートは「そう思う」「少しそう思う」「どちらとも言えない」「あまり思わない」「思わない」という五段階のリカート尺度を採用している。これらをそれぞれ 5 から 1 に換算した結果を表 3 に示す。この結果を見ると、タスク切替えにおいて本システムはタブより好まれており、予測による強調表示に関しては、サイズの変化が透明度の変化よりユーザーの注意を引いている。質問 4 と質問 5 から動的透明度モードはあまり有効ではないことがわかる。また、質問 6 によりユーザーは動的サイズモードで起こる位置の変化を嫌っていることがわかる。

サムネイル部分の結果から、本システムで提案するタスク管理でのサムネイルの使用とサムネイルの改良との有効性が確認できる。まず質問 7 の結果から、タスク管理においてサムネイルの優位性がわかる。質問 8 で、拡張サムネイルが非常に好まれていることがわかる。質問 7 と質問 8 はともに点数が高く、かつ標準偏差も全質問の中で最も小さい。質問 9 の結果から、拡張サムネイルが従来のサムネイルの見易さを阻害していないこともわかった。好感度の部分では、ユーザーの満足度がある程度達成されていることがわかった。質問 17 では、半分以上の被験者が大画面ディスプレイを使用する場合は、このシステムを使うと答えており、再度、本システムの優位性が確認された。

アンケートの後のインタビューの結果を以下にまとめる。被験者 1, 2 はサムネイルのサイズ変化によって起こる位置ずれに不満を持っていたのに対し実験者 4, 7 はこのような位置ずれが気にならない。実験者 5, 7 は動的透明度モードで切替え確率が低いタブのサムネイルが透明になると選びにくくなることを指摘している。実験者 4, 8 はサムネイルが左右両辺に表示されると視線が行き来して疲れると指摘し、実験者 1, 2 はサイズや透明度の変化の意味がわからず、システムが行っている予測のことをあまり意識していなかった。

## 8. 考察と結論

本研究では、まず拡張サムネイルを用いて効率的なタスク切替えを支援する手法を提案した。実験 1 より、拡張サムネイルはサムネイルより探しやすいことが示されている。特に、タブ数が増えることにしたがって、拡張サムネイルの効果が大きくなる。拡張サムネイルはウェブページにある特異な部分、すなわちロゴと特徴画像を大きく表示することで、ユーザーのビジュ

表3 ユーザアンケート結果 ( $\mu$ =平均,  $\sigma$ =標準偏差)

切替え	$\mu$	$\sigma$
1. このシステムがタブよりタスクの切替えが早くなる	4.50	0.71
2. 動的サイズモードで私が次に切り替えるタブのサムネイルを大きくしてくれる	4.00	0.50
3. サムネイルが大きくなることで探しやすくなる	4.63	0.89
4. 動的透明度モードで私が次に切り替えるタブのサムネイルをより不透明にしてくれる	3.67	0.78
5. サムネイルが不透明になることで探しやすくなる	3.33	0.91
6. 切り替える際にサムネイルのサイズと場所が変わるのがいやだ	2.13	0.94
サムネイル	$\mu$	$\sigma$
7. サムネイルはタブより探しやすい	4.63	0.48
8. ロゴと特徴画像をサムネイルに重ねて表示すると区別しやすく、かつ探しやすくなる	4.63	0.48
9. ロゴと特徴画像をサムネイルに重ねて表示すると見にくくなる	4.00	0.79
好感度	$\mu$	$\sigma$
16. 私はこのシステムに満足だ	4.25	0.61
17. 大きいディスプレイが持ったらこのシステムを使う	4.38	0.82

アル感知能力を最大に発揮させることを目指しているが、実験2のアンケートでは、拡張サムネイルは非常に高い評価を得た。また、インタビューにおいて、商品や人物などの特徴画像が明示的なものであるウェブページでは拡張サムネイルは非常にわかりやすいという意見を得た。実験1では実験者に連続的にタスクを切り替えていたため、タブまたはサムネイルの位置はほぼ短期記憶に保存されるが、実際のウェブブラウジングではユーザの作業に伴って、タブまたはサムネイルの位置は徐々に忘れられてしまうので、拡張サムネイルはより大きな効果を期待できる。

また、両辺に常時サムネイルを表示する手法は、特にワイド型の大画面ディスプレイの表示領域を有効利用できる利点があるが、サムネイルを探す際に視線が左右に行ったり来たりすることで効率が下がることがわかった。

また、本研究では、マルチ仮想ワークスペースのようなタスク管理システムにおいて必要とされる手動でのワーキングセット管理を自動的にを行うために、履歴によるタスク切替え確率の予測手法を開発した。しかし、実験2においては、多くのユーザが、確率によりサイズや透明度を変化させる強調表示を、明示的に意識していなかった。また、実際のウェブブラウジングにおいて、検索結果などのページは一回しか切り替えられないことがよくあるので、履歴による予測はそのような状況に適切ではない。今後は、内容の関連性と履歴の双方を利用する予測手法について開発、検討する予定である。

予測される確率でサムネイルを強調する方法としては、動的透明度と動的サイズの二通りを用意したが、動的透明度は実験2のアンケートにより、あまり好まれていないことが示されている。また、インタビューでは、確率が低いタブのサムネイルが透明過ぎて選びにくいことも指摘された。動的サイズモードの主な問題はサイズ変化によって起こる位置変化である。実験2のアンケートにも反映されているように、空間記憶はタスク管理において重要な役割を果たして、サイズの変化による位置のずれは切替え効率を悪化させる。一方、空間記憶の効用にも関わらず、7タブでも拡張サムネイルはタブより走査時間

が短い結果が得られたことは興味深い。この結果はユーザが空間記憶からタブまたはサムネイルの大体な位置を取得したあとに、確認のプロセスも行っていて、この確認プロセスにおいて、拡張サムネイルがタブより効率的であるのではないかと推測される。

今後の課題としては二つの方向を考えている。一つは、タスク切り替え支援手法の改良である。本研究で提案した手法に関しては、ユーザから、サムネイルが画面上の左右両辺に離れて表示される点とサムネイルの強調によって表示位置がずれる点が特に被験者に問題として指摘されており、今後、ユーザの空間的記憶をうまく利用できるサムネイルの新しい配置手法を提案したいと考えている。もう一つは、拡張サムネイルの様々な可能性である。今回の実験でタスク切り替えにおいて拡張サムネイルの有効性を確認できたので、今後拡張サムネイルを検索結果やデスクトップサーチなどに応用する可能性を検討していきたいと考えている。

**謝辞** 本研究の実験にご協力頂いた方に深甚の謝意を表す。本研究は科研費(特定領域研究「情報爆発」, 18049041)の助成を受けたものである。

## 文 献

- [1] H. Weinreich, H. Obendorf, E. Herder and M. Mayer: "Not quite the average: An empirical study of web use", ACM Trans. Web, **2**, 1, pp. 1-31 (2008).
- [2] G. Smith, P. Baudisch, G. Robertson, M. Czerwinski, B. Meyers and D. Andrews: "Groupbar: The taskbar evolved", Proc. OZCHI 2003, pp. 34-43 (2003).
- [3] Y. Miyata and D. Norman: "Psychological issues in support of multiple activities", chapter 13, pp. 265-284 (1986).
- [4] L. Cioni: "User interfaces and multiple activities: The user perspective" (1994).
- [5] V. M. González and G. Mark: "'constant, constant, multi-tasking craziness': managing multiple working spheres", CHI '04: Proc. of the SIGCHI conference on Human factors in computing systems, ACM, pp. 113-120 (2004).
- [6] M. Czerwinski, E. Horvitz and S. Wilhite: "A diary study of task switching and interruptions", CHI '04: Proc. of the SIGCHI conference on Human factors in computing systems, ACM, pp. 175-182 (2004).
- [7] J. D. Austin Henderson and S. Card: "Rooms: the use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface", ACM Trans. Graph., **5**, 3, pp. 211-243 (1986).
- [8] P. J. Denning: "The working set model for program behaviour", Commun. ACM, **11**, 5, pp. 323-333 (1968).
- [9] E. W. Ishak and S. Feiner: "Content-aware layout", CHI '07: CHI '07 extended abstracts on Human factors in computing systems, ACM, pp. 2459-2464 (2007).
- [10] P. Fitts: "The information capacity of the human motor system in controlling the amplitude of movement", Journal of Experimental Psychology, **47**, pp. 381-391 (1954).
- [11] I. S. MacKenzie and W. Buxton: "Extending fitts' law to two-dimensional tasks", CHI '92: Proc. of the SIGCHI Conf., ACM, pp. 219-226 (1992).
- [12] A. Cockburn, C. Gutwin and J. Alexander: "Faster document navigation with space-filling thumbnails", CHI '06: Proc. of the SIGCHI Conf., ACM, pp. 1-10 (2006).
- [13] A. Woodruff, A. Faulring, R. Rosenholtz, J. Morrisson and P. Pirolli: "Using thumbnails to search the web", CHI '01: Proc. of the SIGCHI Conf. on Human factors in computing systems, ACM, pp. 198-205 (2001).