

指紋認識に基づくユーザインタフェース

杉浦 淳 古関 義幸
 NEC C&Cメディア研究所
 {sugiura, koseki}@ccm.cl.nec.co.jp

本論文では、指紋認識技術に基づくユーザインタフェース手法「指紋UI」を提案する。指紋は個人特有の特徴パターンを持つため、主に個人識別の手段として利用されてきた。本UIは、異なる人間の指紋の差異だけでなく、単一の人間の10指の指紋に差異があることを利用したものである。指紋UIでは、ユーザが入力デバイスを操作した(触った)指の種類を指紋によって識別し、その指に割り当てられているデータやコマンドに対応する処理を行なう。単一の入力デバイスであっても、操作する指の種類によりシステムに異なる動作を行わせることが可能であるため、あたかも自分の指にデータオブジェクトやコマンドを保持しているかのような操作感が実現される。

A User Interface Using Fingerprint Recognition

Atsushi Sugiura Yoshiyuki Koseki
 C&C Media Research Laboratories, NEC Corporation

This paper describes a new user interface, called a *fingerprint user interface* (FUI), which employs fingerprint recognition. While the unique feature patterns of fingerprints have mainly been used for personal identification, the FUI is unique in that it uses not only the differences among fingerprint patterns of different persons, but also the differences among the ten fingers of a single person. In the FUI, the system identifies the finger that has operated (touched) an input device through pattern matching of fingerprints and it performs the task assigned to the identified finger. Since users are able to specify different tasks by using different fingers, they feel as if data objects and commands were actually held on their fingers.

1 はじめに

コンピュータ、家電機器、産業用制御システムなどの多くは、操作パネル上に配置されたボタンやタッチスクリーンなどの入力デバイスを指で操作するように設計されている。しかし、指が入力デバイスを触る機会は非常に多いにもかかわらず、指と入力デバイスの間では限られたインタラクションしか行われていない。例えば、複数のボタンから必要なものを選んで指で押下するボタンインタフェースを考える。この場合、指はボタンに割り当てられたコマンドを実行するためのトリガを与えているに過ぎない。

本論文では、指紋認識技術を用いて指-コンピュータ間のより高度なインタラクションを実現する「指紋ユーザインタフェース」(以下、指紋UI)を提案する。本UIでは、システムはユーザが入力デバイスを操作した(触った)指の種類を指紋によって識別し、その指に割り当てられているデータやコマンドに対応する処理を行なう。

これにより、ユーザはあたかも自分の指にコマンドやデータを保持しているかのような操作感を得ることができる。例えば、単一の入力デバイスであっても、そのデバイスを操作する指の種類によりシステムに異なるコマンドを実行させることが可能と

なるため、あたかも自分の指からシステムへコマンドを転送しているかのような感覚が得られる。また、各指に動的にデータを割り当てれば、入力デバイスを触った指ごとに異なるデータをエディタなどにペーストすることができ、指を仮想的なデータ記憶媒体のように利用することができる。

指紋UIではユーザがボタンなどの入力デバイスを触った際に指紋画像を取得するが、そのためには指紋スキャナを入力デバイスの表面に装着しなければならない。しかし、現時点では指紋スキャナの形状やサイズに制限があるため、PDAや携帯電話などの小型機器に組み込むことはできない。このような現状を踏まえ、筆者らは、まずノートブックコンピュータに光学式指紋スキャナを外付けした試作システムを構築し、いくつかのアプリケーション例をシミュレートすることにより、指紋インタフェースの有効性を検証した。

以下では、まず関連研究との比較を行い、次に、指紋インタフェースの概要、実現方法、および利用形態について説明する。その後で、今回の実験システムの詳細および作成したアプリケーションについて述べ、最後に、指紋UIの有効性などについて検討を行う。

2 関連研究

指紋は個人特有の特徴パターンを持つため、従来から個人識別や認証の手段として利用されてきた。これまでも指紋認識技術を用いたセキュリティシステム、入出力管理システムなどが数多く開発されている [4] [5] [7] [10]。指紋 UI は指紋認識技術の一つの応用であるが、指紋認識をユーザインタフェースに積極的に利用した初めての試みであると言える。また、指紋 UI と従来の個人識別システムでは、指紋の利用方法および利用目的が大きく異なる。指紋 UI は、異なる個人間の指紋の差異だけでなく単一の人間の 10 指の指紋の差異を利用しており、それによりコンピュータへの入力にバリエーションを持たせたり指に仮想的にデータを保持できるようにすることを目的としている。

ユーザインタフェース手法としての観点からは、指紋 UI はマルチモーダルインタフェース[6]として位置づけられる。マルチモーダル UI は、キーボードやマウスなどの入力手段以外のモダリティ(modality: 情報伝達のチャネル)を利用するヒューマン-コンピュータインタラクション手法である。特に、音声[9]、ジェスチャ[1]、顔表情[2]、視線[3]といったモダリティを使った UI の研究開発が盛んに行われている。指紋 UI は、新しいモダリティとして「指の種類」を導入したものである。

指紋 UI に最も類似したユーザインタフェースとしては Pick-and-Drop[8]があげられる。Pick-and-Drop は、ペンを用いて複数の端末間でデータオブジェクトのコピーや移動を行う直接操作(direct manipulation)インタフェースである。Pick-and-Drop では、まず、ペンが一方の端末のタッチスクリーンに触った際に、ペン ID とペンがポイントしたオブジェクトをバインドする。次に、ペンが別の端末に触った際にペン ID を確認し、そのペン ID にバインドされているオブジェクトをネットワーク経由で取得しコピーする。これにより、実際には記憶媒体を持たないペンに仮想的にオブジェクトがコピーされたかのような操作感が実現される。

指紋 UI は、ユーザの指に仮想的にオブジェクトやコマンドを保持するという点で Pick-and-Drop と類似している。また Pick-and-Drop と同様にネットワークを利用することにより、指紋 UI においても異なる端末間でオブジェクトをコピーすることが可能である。しかしながら、いくつかの点で指紋 UI に優位性がある。

まず、指紋 UI では指を ID 付きのペンと同様に扱うことができるため、Pick-and-Drop のように ID を認識可能な特殊なペンを持たなくとも、指だけで

データオブジェクトのコピーを行える。また、Pick-and-Drop では各ペンに固有の ID を与えてそれらを管理する必要があるが、指紋 UI の場合は、各個人の各指の指紋パターンの一意性がもともと保証されているためそのような作業も不要である。さらに、指紋 UI では、指ごとに異なるオブジェクトを保持することができるため、複数(最高 10 個)のオブジェクトを同時に扱える。Pick-and-Drop でも、複数のペンを持ち替えて使えば複数オブジェクトを同時に扱うことができるが、これは現実的な方法ではない。また指紋 UI では、指に(オブジェクトではなく)コマンドを割り当てておけば、指の種類を変えることにより異なるコマンドを指定することができるが、このような操作は Pick-and-Drop では実現できない。

3 指紋ユーザインタフェース

3.1 概要

指紋 UI は、ユーザが入力デバイスを操作した指の種類によって、異なる処理をシステムに行わせることが可能なユーザインタフェースである。一個人であっても各指の指紋パターンは異なるため、指紋を識別することにより指の種類を特定することが可能である。

指紋 UI の構成を図 1 に示す。指紋 UI では、まず、ユーザが入力デバイスを操作した際に、入力デバイス表面に装着された指紋スキャナから指紋画像を取得する。次に、取得した指紋画像と Finger ID Table に登録されている指紋パターンとの間でマッチングを行ない、マッチした指紋パターンがあればそのパターンに割り当てられているコマンドまたはデータを操作対象のシステムに通知する。

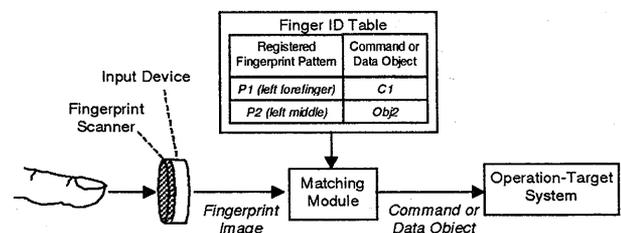


図1: 指紋UIの構成

このように指の種類を区別することにより、指-コンピュータ間の全く新しいインタラクションを実現できる。指紋 UI の特徴の一つは、ユーザが自分の指に仮想的にコマンドやデータを保持して、それらをシステムに転送するような操作感が得られることである。

例えば、3 種類のアプリケーション(AP)をボタ

ン押下により起動させる場合を考える。指の種類を認識しないのであれば、3種類のコマンドを指定するために3個の異なるボタンが必要である(図2a)。この場合、一つのボタンには、例えば「Webブラウザを起動(APの種類+起動)」という機能が割り当てられていることになる。これに対し指紋UIでは、1個のボタンを3種類の指で操作することが可能である(図2b)。そのため、ボタンには「指の種類で指示されたAPを起動」という機能しかなく、ユーザから見ればシステムが本来持っていた機能の一部が自分の指に付加されたかのような操作感になる。

また、図3のように、指紋スキャナを触った指の種類により、異なるデータがコンピュータ上のテキストエディタにコピーされるというアプリケーションを考える。この場合、指の上に保持されたデータを指からシステムに流し込むような操作感が実現される。すなわち、指を(仮想的な)記憶媒体とみなして利用することが可能となる。

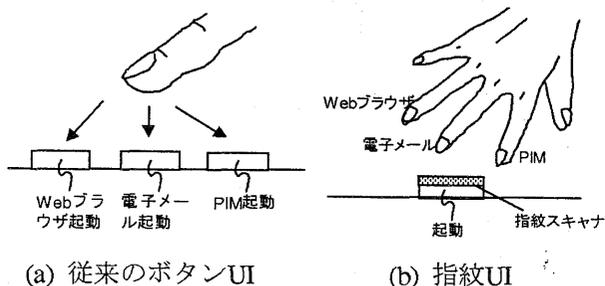


図2: ユーザインタフェースの概念的な比較。

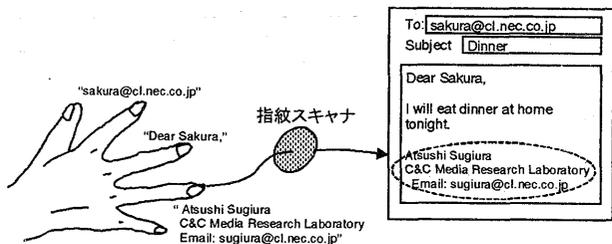


図3: 仮想記憶媒体としての指紋UIの利用。

3.2 指紋UIの実現

図1に示したように指紋UIは、指紋スキャナ、指紋照合モジュール、指紋パターンとコマンドまたはデータオブジェクトを管理するFinger ID Tableを基本的な構成要素としている。これらに加え、指に割り当てられたコマンドやデータの可視化、コマンドおよびデータの仮想的な動きのアニメーションといった処理が、指紋UIの利用を容易にするために必要となる。

3.2.1 指紋スキャナの組み込み

指紋UIの実現にあたって最も重要なのは、いかにして指紋スキャナを入力デバイスの一部として組み込むかということである。ユーザが入力デバイスに触った時に指紋パターンを取得するためには、指紋スキャナを入力デバイスの表面に装着する必要がある。そのため、指紋スキャナのサイズ、形状によりその適用範囲が限定される。

従来、個人識別/認証システムにおいてはプリズムを利用した光学式スキャナ[11]が広く用いられてきた。光学式スキャナは光源となるLEDとプリズムの対面に置かれたCCDカメラにより構成されるが、LEDから指、CCDまでの反射光路を確保するため、小型化には限界がある。現在の平均的な光学式スキャナの厚さは3-4cmであるため、ある程度大型の機器に埋め込んでタッチボタンとして利用することは可能であるが、携帯機器の押しボタンなどの表面に貼り付けて利用することはできない。筆者らの実験システムでは光学式スキャナをタッチボタンとして利用している(後述)。

しかし、近年、半導体センサによるスキャナ[5][7]が開発され急速に小型化が進んでいる。Veridicom社の半導体指紋スキャナは、指が接触した時の静電容量の差を求めることにより指紋画像を取得する。厚さが約3mmで強度も十分なため、ボタンの表面に張りつけて利用することが可能であると思われる。今後、より小型化が進み、指紋UIが適用可能な範囲も増えると期待される。

指紋UIで指紋スキャナを利用する際の問題点の一つは、オブジェクトの直接操作ができないことである。直接操作のためには、指紋画像のスキャニングが可能なタッチスクリーンが必要となる。仮にそのようなタッチスクリーンが開発されたとすれば、指でポイントしたオブジェクトにメッセージ(コマンド)を直接送ったり、ポインティングしたオブジェクトをスクリーンから直接指へ拾い上げる(コピーする)ことが可能となる。

3.2.2 指紋照合

指紋照合には、個人識別で用いられてきた指紋認識手法をそのまま用いることができる。主な認識手法としては、指紋隆線の端点や分岐点の位置関係を利用するマニューシャ方式[4]、指紋画像の濃淡パターンを利用する画像マッチング方式[11]の二種類の手法がある。

これら二つの方式はともに指紋UIで利用可能

¹ 本稿では、実際に押下することにより動作するボタンを押しボタン、触っただけで動作するボタンをタッチボタンと呼ぶ。

であるが、認識精度と認識時間においてトレードオフがある。一般的に、マニユーシャ方式の方が認識精度は高いが、より長い認識時間を必要とする。認識時間が長いのは、指紋画像から指紋端点や分岐点などの特徴点を抽出する前処理が必要なためである。

今回筆者らが構築した実験システムでは、認識精度を重視してマニユーシャ方式を採用した。システムからのレスポンスが遅くなるが、それよりも誤認識によりシステムが意図しない動作をする方が問題が大きいと考えたためである。

3.2.3 指紋登録およびコマンド割当て

指紋 UI では、ユーザが利用する指の指紋とそれに対応するコマンドをあらかじめ登録しておく必要がある。指紋登録は指紋スキャナに指を置けば可能であるため、登録のために特別なハードウェアは不要である。ただし、一連の操作の中で必要な指紋情報を獲得でき、特別な登録作業を必要としない場合もある。これに関しては、*Finger Storage (Finger Memo)* の節で説明する。

3.2.4 コマンドの可視化

指紋 UI での一つの問題は、指にバインドされているコマンドやデータがビジュアルに見えないことである。GUI (Graphical User Interface) では、ボタンやメニューにコマンド名が表示されているため、それさえ見えれば確実に操作できる。しかし、指紋 UI では、指に割り当てられているコマンドを記憶しておかなければならない。これに加え、指紋パターンのマッチングでの誤認識が問題となる。誤認識が起これば、ユーザが意図しないコマンドが実行されてしまうことになる。これらの問題に対処するために、バインディングを忘れた場合にそれを確認したり、指紋の誤認識の場合に操作を途中で中断できる仕組みが必要となる。

指紋 UI では、指が指紋センサに接触した際にその指に割り当てられているコマンドまたはデータをスクリーンに表示し、ユーザが内容を確認できるようにしている (図 4)。プッシュボタンの表面に指紋スキャナを付けた場合は、指がスキャナに接触した時点で(押下する前に)コマンドを表示する。ユーザは自分が意図した通りのものであればそのままボタンを押すだけでよい。また、タッチボタンとして利用する場合には、指紋スキャナに一定時間以上触っていた場合のみコマンドを実際に行う。すなわち、指紋スキャナに触って表示されたコマンドが意図したものでなければ、その場でスキャナから指を離せば実際には実行されない。意図通りのものであれば、コマンドが実行されるまでスキャナに指を置いておけばよい。

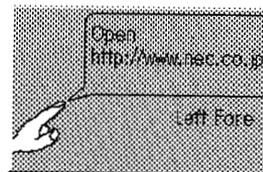


図4: コマンドおよびデータの表示ダイアログ

3.2.5 コマンド・データの仮想的な流れを示すアニメーション

指紋 UI では、指に割り当てられたタスクを実際に行う際に、指(スクリーン上で指紋スキャナに最も近い場所)からアプリケーションへのコマンド・データの仮想的な流れをアニメーションで表示する (図 5)。これにより、ユーザは指紋 UI の概念を容易に理解できるものと期待される。

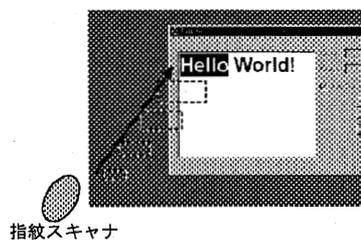


図5: アニメーション

3.3 指紋UIの利用形態

指紋 UI のコンセプト自体はシンプルであるものの、様々な場面で有効である。その利用形態は、指にコマンドを割り当てて利用する *Finger Command* と、指を仮想的な記憶媒体として利用する *Finger Storage* に分けられる。以下、例を用いて説明する。

3.3.1 Finger Commandとしての利用

● モードレス操作

指紋 UI により、コマンドモードの切り替えなしで様々な操作を行なうことが可能となる。例えば、Windows のような GUI 環境では、マウスクリックで単一オブジェクト選択、Ctrl キー+マウスクリックで複数選択というようにキーの押下によってオブジェクト選択モードの切り替えを行っている。これに対し指紋 UI では、マウスボタンの表面に指紋スキャナを装着し、さらに各指にモードを割り当てれば、モードレス操作が可能である。すなわち、人差指でマウスクリックした場合は単一オブジェクト選択、中指では複数選択といった指定できる。従来のように Ctrl キーの押下とマウスクリックを両手で行う必要はなく、指の種類を変えるだけで片手での操作が可能となる。

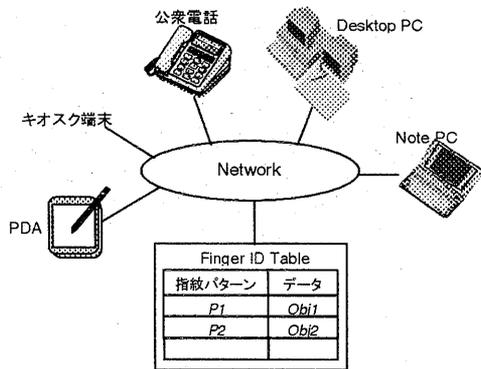


図6：ネットワーク環境での指紋UIの利用

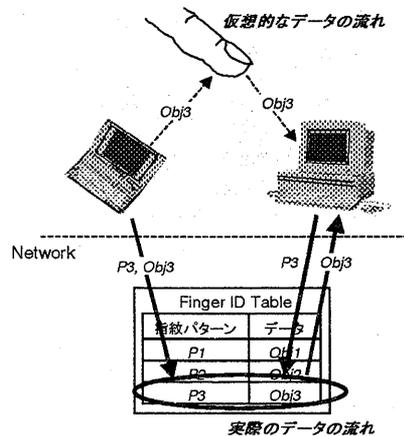


図7：Finger Memoの概念図

● 操作面を見ないでの操作

指紋 UI の特徴の一つは、ボタンなどの入力デバイスを直接見なくても操作できることである。例えば、鞆の中にあるヘッドホンステレオを操作する場合を考える。この場合、操作面にある 5-6 個のボタンの中から手探りで目的のボタンを探すのは非常に困難である。それに対し指紋 UI では、一つのボタンを見つけてそのボタンを異なる指で押し分ければよい。指に割り当てたコマンドさえ記憶しておけば操作はかなり容易になる。同様に、暗闇での操作にも有効である。また、視覚障害者のためのユーザーインターフェースとしても利用可能と考えている。

● 他人にわからないように操作

指紋 UI の利用方法として興味深いのは、他人にわからないようにコマンドを区別できるということである。例えば、人差し指でボタンを押せば扉が開き、中指で警察へ通報するという機能を持ったキャッシュレジスタを作成できる。強盗にレジを開けるように強要された場合でも、指の種類を変えてボタンを押せば強盗にわからないように通報できる。

● ボタン数最少化

指紋 UI では、一つのボタンで複数のコマンドを指定可能であるが、逆に言えば、必要なボタンの個数を減らせるということである。携帯端末のようにボタンを配置できるスペースが限られている小型機器の UI として有望であると考えている。

3.3.2 Finger Storageとしての利用

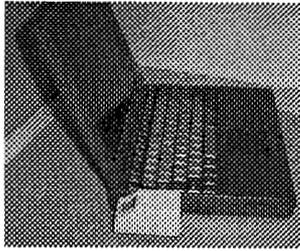
指紋 UI では指をあたかも記憶媒体のように使用できる。例えば、中指で <http://www.aaa.co.jp/> を表示、薬指で <http://www.bbb.co.jp/> を表示というように、各指に Web ページの URL を割り当てておけば、指を Web ブラウザのブックマークとして使うことが可能である (第 4.2.2 節参照)。また、図 3 のように、電

子メールでよく使う文字列を指に割り当てておきメール作成時に必要に応じてコピーすれば、携帯端末のように文字入力が困難な状況では有効である。また、指をカットバッファとして利用すれば、複数のカットバッファを同時に利用できることになる。

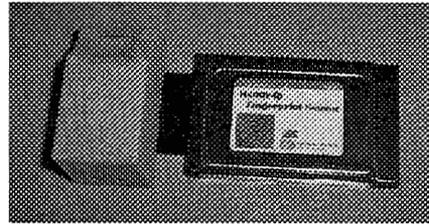
Finger Storage において特筆すべきは、異なる端末間でのデータコピーを、指を介して行えることである。これは、指紋パターンとデータの対応を管理する Finger ID Table をネットワーク上に配置し、異なる端末から共有することにより実現される(図 6)。

具体的には、図 7 に示めすように、最初に片方の端末の指紋スキャナを指で触った際に、そこで得られた指紋パターンとデータオブジェクトをネットワーク上にある Finger ID Table に登録する。次に、指が別の端末を触った際に Finger ID Table の指紋パターンとマッチングを行い、その指に対応するデータオブジェクトをネットワーク経由で取得する。第 2 節で述べたように、指紋 UI では Pick-and-Drop[8] のように特別なペンは不要である。本論文では、Pick-and-Drop と区別するために、指紋 UI のこの利用形態を Finger Memo と呼ぶ。

各個人の指は独自の指紋パターンを有しており一意性が保証されているため、Finger Memo は個人用の機器だけでなく、不特定多数のユーザが利用する公衆端末にも適用可能である。例えば、家のデスクトップで受け取った電子メールに連絡先の電話番号が書いてあれば、それを指に保持して外出し公衆電話でその電話番号に電話する、といったことができる。逆に、キオスク端末 (駅前情報端末、図書館の図書検索端末など) で有用な情報を見つけたらそれを指に保持して持ち帰り、家のデスクトップや PDA に保存することも可能である。メモやペンを携帯せずとも必要なデータを保持して扱うことができるため、非常に有用な利用形態であると思われる。



(a)



(b)

図8： 実験システム外観および指紋スキャナ

4 実験システム

指紋 UI の実現性および有効性を確認するために、実験システムを構築した。しかしながら、現時点で筆者らが所有している指紋スキャナは光学式スキャナだけであるため、図 8a のようにノートブックコンピュータに光学式指紋スキャナを外付けした環境のみでシステム構築を行なった。このような制限があるために実装可能なアプリケーションは限定される。今回は、文書スクロール、Finger Bookmark、Finger Memo の 3 個のアプリケーションを作成した。

4.1 実装

今回使用した光学式スキャナは、NEC 製のものである。図 8b に示すように PCMCIA カードに結合されており、PCMCIA スロットに挿入して使用する。カードを除いたスキャナ部分のサイズは、幅 3.3cm × 奥行き 6.2cm × 高さ 3.1cm である。PCMCIA カードと一体化しているために、指紋スキャナをプッシュボタンと組合せて利用することはできず、タッチボタンとしてのみ利用可能である。すなわち、指をスキャナに置くことにより指紋画像が取得され、そのイベントがシステム本体に通知されることになる。また、本指紋スキャナはその構造上 1 個の PC に 1 個しか装着できない。

指紋識別方式としては、指紋パターンの特徴点を利用したマニューシャ方式[4]を採用した。指紋スキャナで得られた指紋画像の特徴点と予め登録した指紋の特徴点を比較してマッチングスコアを計算し、そのスコアが指定された閾値を超えた場合に指が特定される。

実験システムは、OS が Windows95、CPU が 133MHz Pentium の B5 版サイズ Note PC 上に実装されている。指紋スキャンおよび特徴点抽出にかかる時間は約 1.6 秒であり、1 指当たりの指紋照合時間(マッチングのみ)は平均約 0.06 秒である。登録された指紋が n 個の場合、指が特定されるまで順に照合していくため、平均で $1.6 + 0.06 * (n + 1) / 2$ 秒かかることになる。

4.2 例題アプリケーション

4.2.1 文書スクロール

最初のアプリケーションは、指紋 UI を Finger Command として利用したものである。本 AP では、指紋スキャナを触る指の種類によりスクロールの方向および幅を変更する。具体的には、左手人差し指、中指、薬指、小指で、それぞれ下方向に 1 行スクロール、ページスクロール、上方向に 1 行スクロール、ページスクロールを行う。指の種類とコマンドとの対応はあらかじめ設定されているため、ユーザは自分の指を登録するだけで利用可能である。

4.2.2 Web ブラウジングのための Finger Bookmark

Finger Bookmark は、指に Web ページの URL (Uniform Resource Locator) を保持したブックマークである。あらかじめ各指に URL を登録しておき、指が指紋スキャナを触った際に対応する Web ページを Web ブラウザに表示する。そのため、ブラウザのブックマークを開いてその中から目的のページを選択する必要はない。ブラウザが起動されていないければ、ブラウザの起動も合わせて行う。

Finger Bookmark は、指に URL を保持するという Finger Storage としてだけでなく、Web ページを開くためのショートカットという Finger Command としての側面も合わせ持っていると言える。

4.2.3 Finger Memo: 複数端末間でのデータ交換

Finger Memo は、指を Finger Storage として利用したものである。指に仮想的にデータを拾い上げて、異なる端末間でのデータのやり取りを行う。今回の実装では、指紋スキャナを装着した Note PC を 2 台使用して実現した。これら 2 台のコンピュータはネットワーク上に置かれた共通の Finger ID Table を参照して動作する。

Pick-and-Drop とは異なり、ペンを用いなくても 2 台の端末間でデータの転送が行えるが、データの直接操作は行えない。Pick-and-Drop では、タッチス

クリーン上のデータオブジェクトをペンで直接 Pick し、Drop することが可能である。指紋 UI で同様の直接操作を行うためには、ディスプレイ自体に指紋スキャン機能を付加し、ディスプレイを指で触った際に指紋画像を取得する必要があるが、現時点では実現できない。本実験システムではマウスと組合せて操作を行う。具体的には、データをマウスで選択し、その後でスキャナを触るとその指に選択データが仮想的に指にコピーされ、もう一方のマシン上でデータのコピー先をマウスで指定した後にスキャナを触るとデータがペーストされる。

5 議論、検討

5.1 Finger Storageとしての有効性

現時点での実験システムの構成であっても、Finger Memo および Finger Bookmark では、あたかも指にデータが保持されたかのような感触を体感可能であった。Finger Memo では、ディスプレイから指へデータオブジェクトを直接コピーできないという短所はあるものの、ユーザは特別なツール(ペンなど)を用いずとも異なる端末間でのデータの移動が容易に行うことができ、その有効性は明らかである。

また、3.3.2 節で述べたように、不特定多数の人が利用する公衆端末にも指紋 UI は適用可能である。今後、街中に置かれた電話や電子掲示板などあらゆる機器がネットワークで結合されるようになれば、様々な場面で活用できると期待される。

Finger Memo をインターネットなどの公衆ネットワーク上で利用する場合には、指紋データの盗用の危険性がある。人間の指紋はパスワードのように変更することはできないため、指紋データの暗号化などの対策は不可欠である。しかしながら、Finger Memo の場合、指紋データが盗まれ暗号が解読されたとしても、それが誰の指紋データであるかを割り出すことはできないため、比較的危険は少ないと考えている。

5.2 Finger Commandとしての有効性

今回は、Note PC + Windows 95 デスクトップ環境上に例題 AP を構築したが、残念ながらこのような環境では Finger Command としての有効性はあまり発揮できない。すなわち、Finger Bookmark を Windows95 のデスクトップ上に置かれた URL へのショートカットと比較しても、大きな優位性は確認できない。また、文書スクロールの場合も、キーボードの“↑”、“↓”、“PageUp”、“PageDown”キーを押し分ける場合と比較して優位性はない。デスクトップやキーボードのように十分な数のアイコンやボタンを配置でき

る場合には、一つのボタンに一つの機能が割り当てられている方がむしろ自然だからである。

指紋 UI を Finger Command として利用する場合は、アイコンやボタンを多く配置できないスマートフォン携帯端末での操作、操作面を見ることができない暗闇での操作、鞆の中にある機器の操作、といった特殊な環境で有効性が発揮されると期待される。例えば、Web ブラウジングを行う場合、ショートカットアイコンを配置するスペースが無いスマートフォン端末では、Web ブラウザを起動し、ブラウザのブックマークを開き、その中から目的のページを探して指定しなければならない。電車の中で立ちながら端末を操作するといった不安定な状況を想定すると、かなり負担のかかる操作である。これに対し、指紋 UI を用いた Finger Bookmark の場合、端末にブラウザ用のボタンを一つだけ付ければ、一度の操作で目的のページをブラウズすることが可能となる。また、筆者らは鞆の中にある機器の操作に関して予備実験を行ったが、操作面が見えない状況で複数のボタンの中から目的の一つのボタンを探り当てるよりは、一つのボタンを異なる指で押下する方が操作は容易であった。

一つのボタンで複数のコマンド指定を行う方法として、マウスボタンのダブルクリックがある。しかしながら、シングルクリックとダブルクリックでは、2 個のコマンドを区別できるにすぎない。3 個以上のコマンドを区別するためには、トリプルクリックが必要となるが、マウスボタンを短い間隔で繰り返してクリックするのは困難な操作である。指紋 UI では指の種類を変更してボタンをクリックすれば、1 アクションで操作が可能であり、最高 10 個のコマンドまで指定可能である。

5.3 操作性

例題アプリケーションの評価を通じて得られた別の知見は、指紋 UI は入力スピードが要求される操作には適さないということである。一つのボタンを複数の指で押すためには、手の平の位置を移動しなければならない。ボタンの位置に対して、手の平の位置を一定に保つことができずホームポジションが決まらないため、入力速度の向上は見込めない。

また、手の平とボタンの相対位置が固定されている状態で指だけを左右に動かすのは非常に困難であるため、携帯機器を片手で持ちつつその手で操作するという状況では指紋 UI の利用は難しい。すなわち、両手で操作するか、または、固定された装置を片手で操作する必要がある。

さらに、指紋スキャナへの指の押し付け方にも気を配って操作する必要がある。現状では、指紋認

識に十分な数の指紋特徴点が獲得できるように、指の腹の広い面積を指紋スキャナに押し付けなければならない。しかしながら、実際の利用場面では、指紋スキャナを触る角度や、指紋スキャナのサイズにより、指紋画像の一部しか得られないことも考えられる。入力デバイスとして気軽に用いるためには、指紋画像の断片からでも正確に指の種類を識別できる指紋認識方式が必要となる。

6 結論および今後の課題

本論文では、指紋認識技術を用いたユーザインタフェース手法である指紋 UI を提案した。指紋 UI により、指を仮想的な記憶媒体として利用したり、指に特別な機能を付加することが可能であり、従来にない操作感、利便性が実現される。

しかしながら、筆者らが構築した実験システムは、Note PC と光学式スキャナの組み合わせであり、現時点で構築可能なアプリケーションやその利用環境は非常に限定されている。今後、半導体センサを用いることにより、携帯機器のボタンやマウスボタンに指紋 UI を適用し、様々な利用環境での有効性を確認していく必要がある。

また、指紋 UI の実用化のためには、指紋認識スピードの向上が必要となる。現在、1 回の認識に最低 1.7 秒程度(指紋画像スキャン、特徴点抽出、および 1 個の指紋のマッチング)必要であり、操作をしてからのレスポンスが悪い。第 4.2.3 節の Finger Memo のように指を記憶媒体として利用する場合はそれほど苦痛ではないものの、第 4.2.1 節の文書スクロールのように短い間隔で操作を繰り返す必要がある場合は非常にいらついてしまう。現在、特徴抽出やマッチング処理はソフトウェアで実現しているが、これらの処理をハードウェアで実現することにより、レスポンスが高速化できると考えられる。

これに加え、第 5.3 節で述べたように、十分な数の指紋特徴点が得られない場合でも指の識別ができるように、マッチングアルゴリズムを改良する必要がある。指紋 UI の操作性を向上させるためには、指の腹の一部しかスキャナに触れず、指紋画像の断片しか取得できなかった場合でも指の種類を正しく認識できることが要求される。

本論文では、一個人の 10 指の指紋の違いを用いた UI の利用方法に焦点をあてたが、元来の個人識別としての指紋認識技術の利用も重要である。単一の機器を複数の人で利用する場合、指紋はパーソナライゼーション(Personalization)の有効な手段であり、今後はその利用方法を検討していく。

謝辞

実験システムの構築にあたってご支援下さった NEC C&C メディア研究所パタン情報 TG 内田主任、黄氏に感謝いたします。また、様々な有用な助言を下さった NEC C&C メディア研究所マルチメディアミドルウェア TG 神場主任研究員、坂上主任、小池主任、島村氏、坂口氏に感謝いたします。

参考文献

- [1] Bolt, R.A., "Put-That-There: Voice and Gesture at the Graphics Interface", Computer Graphics, Vol. 14, No. 3, pp. 262-270, 1980.
- [2] Dohi H. and Ishizuka M. "Visual Software Agent: A Realistic Face-to-Face Style Interface connected with WWW/Netscape", Proceedings of IJCAI'97 Workshop on Intelligent Multimodal systems, pp.17-22, 1997.
- [3] Frey L. A., Preston White Jr. K., and Hutchinson T. E., "Eye-Gaze Word Processing", IEEE Transactions on systems, Man, and Cybernetics, Vol. 20, No. 4, 1990.
- [4] Hoshino Y., Asai K., Kato Y., and Kiji K., "Automatic Reading and Matching for Single Fingerprint Identification", Proceedings of The 65th International Association for Identification Conference, 1980.
- [5] Klett J., "Thermal Imaging Fingerprint Technology", Proceedings of Biometric Consortium 9th Meeting, 1997.
- [6] Maybury M.T., "Intelligent Multimedia Interfaces", The AAAI-Press/MIT-Press, 1993.
- [7] McCalley K., Setlak D., Wilson S., and Schmitt J., "A Direct Fingerprint Reader", Proceedings of CardTech/SecureTech, Vol. 1: Technology, pp.271-279, 1996.
- [8] Rekimoto J., "Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environments", Proceedings of UIST'97, pp.31-39, 1997.
- [9] Sadek, M.D, Ferrieux A., Cozannet A., Bretier P., Panaget F., and Simonin J., "Effective Human-Computer Cooperative Spoken Dialogue", International Conference on Spoken Language Processing, 1996.
- [10] Schneider J., "Improved Image Quality of Live Scan Fingerprint Scanners Using Acoustic Backscatter Measurements", Proceedings of Biometric Consortium 8th Meeting, 1996.
- [11] 松本, 梅崎, 「2次元フーリエ変換に基づく指紋照合法の評価」, 電子情報通信学会技術報告, Vol. PRU93, No. 39, pp.37-44, 1993.