

原著論文

入力領域縮小機能を追加したタブレット向けの親指ジェスチャーによる文字入力方式

秋田 光平¹⁾, 田中 敏光¹⁾, 佐川 雄二¹⁾

¹⁾名城大学理工学部

Character input method by thumb gesture for tablet PC that is added the automatic shrink function of the input area

Kohei AKITA¹⁾, Toshimitsu TANAKA¹⁾, Yuji SAGAWA¹⁾

¹⁾ Faculty of Science and Technology, Meijo University

Abstract: We are developing a character input system by thumb gesture for tablet PC. With the system, an expert can enter about 60 characters per minute (cpm). Beginners can enter about 18 [cpm] in average after only 20 minutes training. The system hides only small area of screen that is for the guide, since the input area is transparent. However, a user can not touch behind of the area. To solve the problem, we added the reduction function for the input area. When the gesture is not done for a certain second, the area shrinks in automatic. The area returns to normal size immediately after the thumb touch the area. To achieve the usability in character input same to the system without the reduction function, the position of the first touch must be on the reduced area. However, the distribution of the positions varies by each person. So that, we developed the method for computing the size and the position of the area from the record of touch points. Experimental results proved that the proposed method can determine the shrunken area that is optimized for each user.

Keywords: tablet, user interface, gesture, character input, both hands holding

キーワード: タブレット, ユーザーインターフェース, ジェスチャー, 文字入力, 両手持ち

1. はじめに

大型のタブレットを手で持って使う場合, 両端を握ると姿勢が安定し, 腕の疲れも少なくなる。ただし, この持ち方では, 画面の左右端を親指でタッチすることしかできなくなるため, Windows[1]や Android[2]にはスクリーンキーボードを左右に分割配置する機能が用意されている。また, qwerty キーボードを縦置きにした手法[3], Tagtype [4]をスクリーンキーボードにした手法[5][6], KALQ 配列の左右分離キーボード[7], 親指の旋回方向と曲げ伸ばし方向にキーを並べた手法[8], 左右の指の上下方向の位置で文字を選ぶ手法[9], などが提案されている。これらの手法は, いずれも画面上にキーを配置し, 指で触れることで文字を指定する。片側のキーの数は, 少ないもので縦 5 個, 多いものでは縦 5×横 6 個となっている。一方, 深津らの研究[10]では, ブラインドタッチだと 3×3 のキー配置でも 19%程度の入力誤りが生じることが報告されている。この結果に従えば, 上記の手法では目で指の位置を確認しながら操作する必要があり, 文字入力の位置によっては, 目疲れの原因となる視点移動の回数が増える。

これに対して, ジェスチャーを使う方法では指先を視認する

2017年5月30日受理。

必要はない。スマートフォン向けではあるが, 青木ら[11]や井川ら[12]の手法ではアイズフリー入力が可能である。ただし, 移動方向が 8 分割されており, 同一方向への大きな移動も使っているため, そのままタブレットに適用することはできない。これは, タブレットの両端を持つと親指の旋回と曲げ伸ばしでジェスチャーを行うことになるため, 指先の到達範囲が狭く, 細かい方向の指定も難しいからである。

そこで我々は, 移動方向を上下左右に限定し, タッチ位置から各方向に 1 ストロークの範囲内でジェスチャーを完結する手法を開発した[13]。詳細は次章で述べるが, この手法には, 入力時に指先を視認する必要がなく, 画面を隠す面積が少ない特徴がある。ただし, ジェスチャーを入力する領域では, 背後に置かれたものを見ることはできるが, タッチすることはできない。このため, 誤操作が増えたり, 思ったように操作できないことでフラストレーションがたまったりする。

本研究では, 先行研究の入力領域の背景に触ることができない問題を解決するため, 入力が止まったら自動的に入力領域を縮小する機能を追加する。この機能に求められる条件は, 先行研究の特徴である「指先を視認する必要がない」長所を消さないように, 「縮小された領域に指先を見ずにタッチできること」である。本研究では, 待機時の領域の位置と大きさを, 利用者ごとの利用履歴から自動的に設定する手法を実装することで, この条件をクリアする。

2. 先行研究

先行研究[13]では、タブレットの両端を持ち、親指で画面に触れてジェスチャーを行なう。利き手で文字を、反対の手で制御コードを入力する。図1は右利き用のインターフェースで、右側の白い矩形の内側が文字の入力領域（ジェスチャーを行う場所）である。左側の黒い矩形は制御コードの入力領域である。タブレットを持つ位置は個人により異なるので、どこを持ってもしっかり入力できるように、画面の左右端のそれぞれで上から下までを入力領域としている。

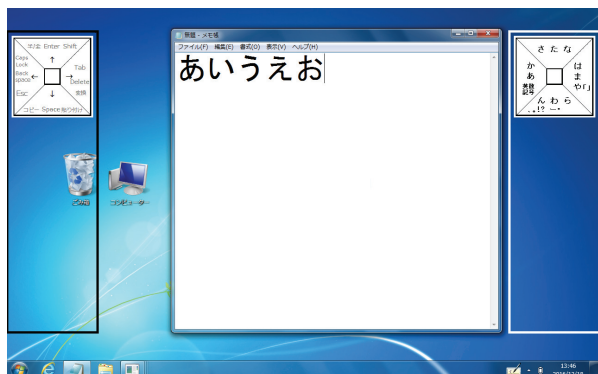


図1 先行研究のインターフェース

文字は平仮名、英数、記号に分けて入力する。平仮名モードでは、50音表の行と段をこの順に指定することで1文字を入力する。指を上下左右のいずれかの方向に一定の長さ以上動かす動作をストロークとし、ストロークを1つまたは複数つないでジェスチャーを定義する。行選択では、「あ行」は「左→右」、「か行」なら「左→上」のように、2ストロークのジェスチャーで清音を指定する。濁音・半濁音・小文字の行には清音の最後のストロークと逆向きのストロークを追加したジェスチャーを割り当てる。例えば「が行」のジェスチャーは「左→上→下」となる。段選択は、フリック入力に倣い、「あ段」をタップで、「い段」～「お段」をそれぞれ「左」「上」「右」「下」のストロークで指定する。

ジェスチャーの判定に指先の座標を使わないので、ボタン等の位置を限定する情報を表示する必要はない。このため、入力領域は背後の表示が見えるように透明化されている。また、座標を使わないので、指先が触れる位置を気にする必要はない。先行研究の実験[13]では熟練者の入力速度は59.4cpmとなっているが、クローズ型HMDを装着し画面を表示することで手元を完全に見えない状態にして行っても、入力速度はほとんど変化しない。このように、慣れれば手元を見なくても問題なく文字を入力することができる。

初心者でも容易に入力システムを使えるようにするために、入力を補助するガイド(図1の左上と右上にある、文字が書かれた白い四角)を表示する。ここには、その時点で、4方向のそれぞれに指を動かしたときに入力できる文字が表示される。

表示は1ストローク入力されるたびに更新されるので、ガイドでジェスチャーの正誤を確認できる。また、ガイドに示される文字を追いかけて指を動かすことで、ジェスチャーを全く覚えていなくてもその文字を入力することができる。

画面に表示されるのは入力ガイドと入力領域の枠線だけなので、画面を隠す割合が低い。図1では画面の約93%が可視であり、仮想キーボードに比べて大幅に視認できる割合が高い。しかし、入力領域(図1の白い矩形と黒の矩形の内側)へのタッチはジェスチャーの入力として処理されるので、入力領域にあるものは、見えていてもタッチできない。つまり、図1のごみ箱にはタッチできない。タッチできる面積で比較すると、仮想キーボードと差が無くなってしまう。

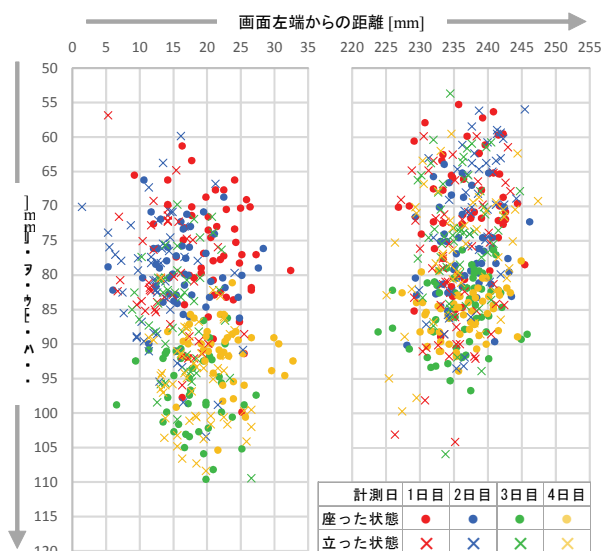
3. 研究の目的

先行研究の欠点は、入力領域の背後にタッチできないため、入力システムの起動中は画面の操作が制限されてしまうことである。この問題を解決するために、入力が止まった時(待機時)に自動的に入力領域を縮小する機能を追加する。ただし、縮小するとタッチ位置が入力領域から外れる可能性が生じる。先行研究の長所は、指先を視認することなく使うことができる点にあるので、待機からの復帰でも指先の視認は不要でなければならない。最初にタッチする位置(ジェスチャーを始める位置)は個人ごとに異なることが予想されるので、縮小された入力領域の位置と大きさは、個人ごとに、適切な値に選ぶ必要がある。そこで本研究では、初めに、タッチ位置の分布を調べる実験を行い、分布の特徴を分析する。次に、その結果に基づいて、使用履歴から個人ごとに待機領域(待機時の入力領域)の位置と大きさを決める方法を提案する。

4. タッチ位置の計測実験

ジェスチャーを始める位置、すなわち最初にタッチする位置の個人差や、日時や姿勢による変動を調べる実験を行った。この実験では、タブレット(Surface3 Pro, 外形:幅292mm×高さ201mm, 画面サイズ:12インチ)の両端を握り、親指で画面をタップする。タブレットを持つ位置は指定せず、被験者が持ちやすいと思う位置を選んでもらう。

画面上のアイコンをタップした後に文字を入力する場面を想定して、画面のランダムな位置にマークを表示し、被験者にはマークをタップした後に、その手で入力領域をタップしてもらう。左右どちらの手でタップするかは被験者に任せる。握ったままでは画面中央には指が届かないので、手を放してマークをタップする場面も生じる。どの指でマークをタップするかも被験者に任せるが、入力領域へのタップは、タブレットの端を握った状態で、親指で行う。この方法で100回のタップ動作を続けて行い、1セットの実験とする。1名の被験者に対して、立った状態と座った状態のそれぞれで4セットずつの実験を行



(a) 左手親指にタップ位置 (b) 右手親指のタップ位置
 図 2 被験者 G のタップ位置の分布

う。各セットは日を変え、延べ 8 日間かけて行う。

この実験を男子大学生 7 名に対して行った。その中で位置のばらつきが大きかった被験者 G の結果を図 2 に示す。図の左側は左手親指の、右側は右手親指のタップ位置を示している。位置は画面の左上を原点、右方向を X 軸の正の向き、下方向を Y 軸の正の向きとして、mm 単位で表示している。また、図 2 の左右端が画面の左右端に対応している。図の●は座った状態での、×は立った状態での計測値である。色は日の違いを表しており、赤、青、緑、黄色がそれぞれ第 1, 2, 3, 4 日目の実験結果を示している。右手と左手の比較では、明らかに上下方向にタップ位置の分布がずれている。この結果は、左右で縮小時の入力領域(以後、待機領域と呼ぶ)の位置と大きさを変える必要があることを示している。

色の分布も偏っているので、日によるタップ位置の変動が認められる。図 3(a)に、被験者 G の右手のタップ位置の X 座標(画面左端からの距離)の分布を示す。このグラフは、計測日ごとに、X 軸方向の 1mm を 1 区間として計測値のヒストグラムを作り、サンプル数で割って相対度数とし、5 区間(前後それぞれ 2 区間)で移動平均を行った結果を、折れ線で表示している。実線が座った状態で、破線が立った状態で計測した結果である。計測した日によってピークの位置も分布の範囲も異なっている。

図 3(b)に、立った状態で計測した 4 日分のデータと座った状態で計測した 4 日分のデータをそれぞれ平均したグラフを示す。2つのグラフの形はよく似ている。X 座標のそれぞれの値で 2つのグラフの差の絶対値を求め、合計すると 0.108 となる。各グラフの相対度数の総計は 1 なので、2つのグラフの差は 10.8%に過ぎない。これは 4 日分のデータを平均することで日々の変動が抑えられた結果だと考えられる。タッチ位置の X 座標は親指の曲げ伸ばしにより変わるが、位置の基準はタブ

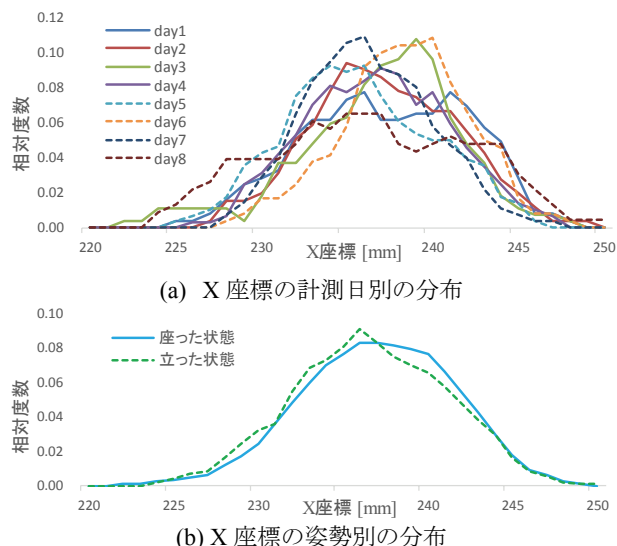
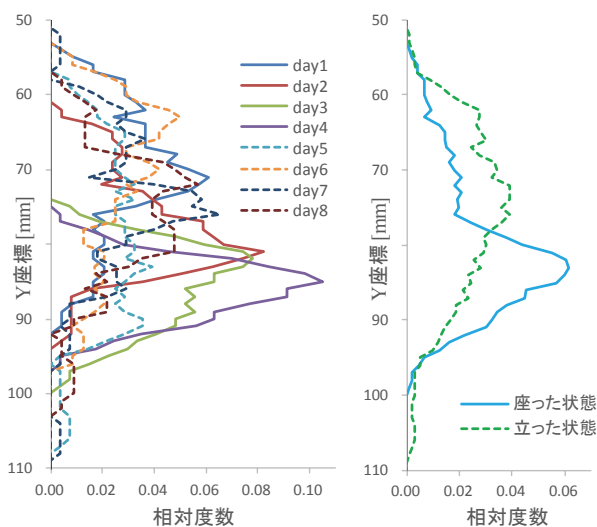


図 3 被験者 G の右手のタップ位置の X 座標の相対度数



(a) Y 座標の計測日別の分布 (b) 姿勢別の分布
 図 4 被験者 G の右手のタップ位置の Y 座標の相対度数

表 1 2つの姿勢の分布が重なる範囲に含まれるサンプル

被験者	A	B	C	D	E	F	G
X座標	97.3%	98.2%	98.0%	95.7%	97.2%	96.1%	98.1%
Y座標	95.7%	95.9%	99.5%	94.5%	92.9%	92.5%	95.2%

レットの右端であり、タップできる範囲は親指の長さで決まるため、それほど大きく変動することは無い。このため、4 日分のデータ(サンプル数は約 200)でほぼ平均化できたと考えられる。

一方、タッチ位置の Y 座標(画面上端からの距離)はタブレットを握る高さで変動するため、ばらつきが大きくなる。図 4(a)に同じ方法で求めた Y 座標の分布を示す。計測日により、分布のピークの位置が上下方向に大きく変動している。図 4(b)は姿勢ごとに平均したグラフだが、X 座標とは異なり、分布が上下にずれている。これは、タブレットを握る高さは日ごとに変わるが、図 4(a)の結果から、同一計測日の分布は範囲が限ら

れるので、4 日分の計測結果では位置のはばらつきが平均化されなかったためと考えられる。

そこで、2 つの分布が重なる範囲を次の手順で求めた。(1) 立った姿勢と座った姿勢のそれぞれで、1mmを1区間とするヒストグラムを作る、(2)両方のサンプルが存在する区画の上限と下限を求める、(3)ばらつきの影響を減らすため、分布の上下端から相対度数の合計が1%以下となる区間を削る。被験者GのY座標では、分布が重なるのはY座標の値が60から94の範囲となるが、この間にサンプルの95.2%が含まれていた。

残りの被験者についても、先に述べた手順で座った状態と立った状態の分布の重なりを求めた。表1に、重なった範囲に含まれるサンプルの割合を示す。X座標での最小値は被験者Dの95.7%で、分布の範囲はほぼ重なっているといえる。Y座標ではX座標より値は低下するが、最低値は被験者Fの92.5%であり、全ての被験者で姿勢によるタップ範囲の大きなずれは認められなかった。この結果から、縮小範囲の推定においては、姿勢を分ける必要は無いと考えた。

図5に被験者A～Fの右手の計測結果を示す。この図では、日や姿勢を区別せずに表示している。被験者間で比べると、タップ位置の分布は大きく異なっている。被験者Cではタップ位置が狭い範囲に集中している、AとBも狭い範囲に集まっているが、Aでは円形に分布しているのに対して、Bではやや縦に伸びている。D～Fでは分布が広がっており、縦方向に伸びている。特にFと図2に示したGの分布では、縦方向が横方向の2倍以上に広がっている。この分布の差異は、個人ごとに待機領域の位置と大きさを決める必要があることを示している。

5. 待機領域の算出方法

図6に、各被験者の8日分の右手のタップ位置を合計した分布を示す。グラフは図3、図4と同じ方法で作成している。X座標(画面の左端からの距離)の相対度数は多くの被験者で正規分布に似た形となっている。Y座標(画面の上端からの距離)では、分布が狭い被験者A,B,Cでは正規分布に似た形となっているが、分布が上下に広がっている被験者D～Gでは凹凸が目立っている。これらの被験者ではY座標の範囲が広がっているため、1区間を1mmとしたヒストグラムでは、約400個のサンプルに対して区間の数が多すぎるため、ばらつきの影響が消えないためと考えられる。そこで、1区間の幅を2mmに変えて区間数を半分にしたところ、図6(c)となった。被験者D～Gでも、中央付近をピークとして、そこから離れるにつれて相対度数が減少する傾向が見られる。

そこで本研究では、タップ位置の平均と標準偏差を使って縮小領域を決める。縦方向(Y座標)と横方向(X座標)で位置のはばらつきに違いがあるので、それぞれを独立に計算する。表2に8日分のデータから求めた右手のタップ位置の平均と

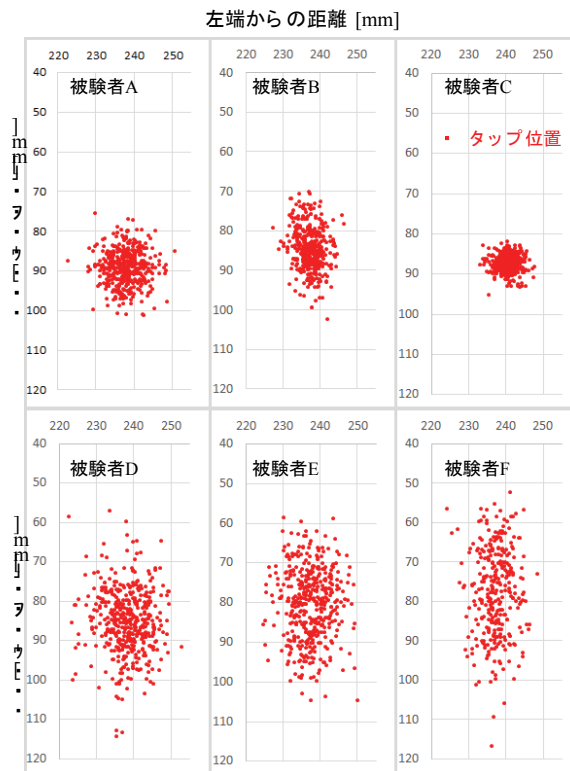
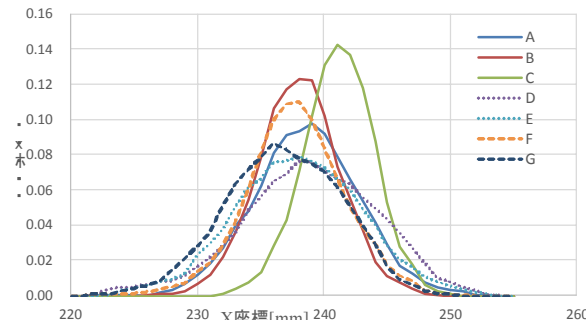
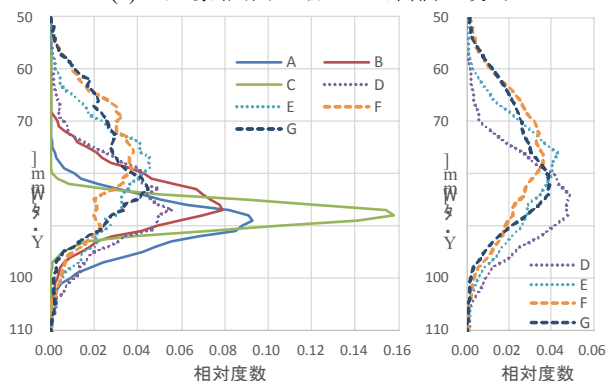


図5 被験者A～Fの右手のタップ位置の分布



(a) X座標(画面左端からの距離)の分布



(b) 1区間1mmの場合のY座標(画面上端からの距離)の分布 (c) 1区間2mmの場合の分布

図6 被験者A～Gの右手のタップ位置の相対度数

標準偏差を示す。本研究では、画面上の位置を、画面の左上の角を原点、右方向をX軸の正の向き、下方向をY軸の正の向きとして、mm単位で計測している。被験者により標準偏差の値が大きく異なっている。またY座標の差も大きい。

待機領域はタップ位置を包含するように設定されなければならない。ただし、タップ位置はばらついているので、すべてを含むように設定すると、領域が大きくなりすぎてしまい、縮小する意義が薄れてしまう。そこで、タップ位置の平均値を中心として、縦、横それぞれの標準偏差に係数をかけた値を幅とする矩形を待機領域とする。ここで問題となるのは、係数をいくつにするかである。図7は、係数の値を横軸に、実験で採集

したタップ位置の中で待機領域から外れるものの割合を縦軸にとったグラフである。被験者によりばらつきはあるが、標準偏差の6.5倍以上であれば、すべての被験者で待機領域から外れる割合が1%以下になる。そこで、本研究では係数として6.5を採用する。

以上から、縮小領域の位置と大きさを次の式で算出する。

$$\begin{aligned} \text{待機領域の中心の座標} &= \text{タップ位置の平均} \\ \text{待機領域の縦幅} &= \text{縦方向の標準偏差の } 6.5 \text{ 倍} \\ \text{待機領域の横幅} &= \text{横方向の標準偏差の } 6.5 \text{ 倍} \end{aligned}$$

図8では、この式を用いて求めた待機領域を青色の矩形で描き、図5で示した分布に重ねて表示している。どの被験者でも、タップ位置のほとんどが待機領域に含まれており、かつ、タップ位置が集まった領域と矩形との間に大きな隙間が無いので、提案手法の算出式は妥当だといえる。

ただし、利用者によっては、待機領域が小さすぎて、そこから外れる割合が高くなる可能性がある。また、待機領域が無駄に大きくなる恐れもある。これに対処するため、係数6.5はデフォルトとし、利用者が値を調整できるようにする。

表2 使用者ごとのタッチ位置の平均と標準偏差

被験者	位置の平均[mm]		標準偏差[mm]	
	X	Y	X	Y
A	238.1	89.0	4.0	4.4
B	237.5	84.5	2.9	5.4
C	240.8	87.4	2.4	2.1
D	238.4	85.2	5.2	8.2
E	237.5	80.4	4.7	9.0
F	237.3	77.2	3.6	10.5
G	236.3	77.5	4.3	9.8

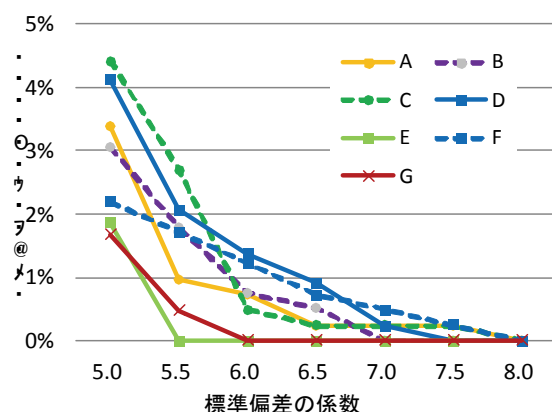


図7 係数を変えたときの待機領域を外れる割合

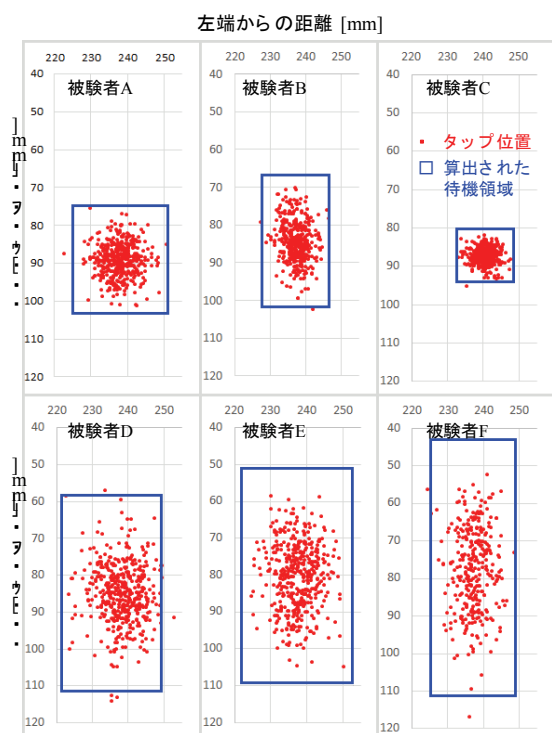


図8 算出された被験者A~Fの待機領域

6. 入力領域縮小機能の実装

改良したシステムには、一定時間入力がないと入力領域を自動で縮小する機能を追加する。縮小された領域(待機領域)にタッチすると、その時点で通常の大きさに復帰する。また、タッチ位置も引き継がれる。このため、待機領域にタッチすることができれば、縮小しない場合と同じようにジェスチャーを入力することができる。

前章では、タップ実験の結果から待機領域を求めた。しかし、実用においては、待機領域を決めるための入力タスクを利用者に強いることはできない。そこで、利用履歴を活用する。先行研究のシステムには、タッチ位置や指の動き、判定結果などをログファイルに残す機能があるので、そのデータからタッチ位置のサンプルを抽出する。

待機状態から復帰する場合のタッチ位置では、待機に入った回数しかサンプルを採集できない。また、タッチ位置が待機領域から外れた場合には記録が残らないので、待機領域が小さすぎた場合は、拡大されるまでに時間がかかる。そこで、一定時間(デフォルトは10秒)入力が空いた直後のタッチ位置をサンプルとして抽出する。ジェスチャーが連続している場合には、ジェスチャーの開始位置は直前のジェスチャーの終了位置に影響されてずれてしまう。しかし、一定時間入力が止まった後では、指はホームポジションに戻ると考えられるので、影響が無いか軽微となることが期待できる。

上記の点の中にはジェスチャー入力を意図しないタッチ位置(たとえば入力領域の背後にあるアイコンをタッチしようとした場合の位置など)が含まれることがある。このため、求めた点

をすべて使って 5 章で述べた手法で待機領域を計算すると、過剰に大きくなってしまふ恐れがある。そこで、遠く離れた点はエラーとして除去する。縦方向と横方向でばらつきが異なるため、独立に処理する。方向別にすべてのサンプルの座標の平均を求め、平均値との差の絶対値が大きいほうから 5%の点を取り除く。縦と横の両方で上位 5%に入る点も存在するので、除かれるサンプル点は全体の 5%以上 10%以下となる。

7. 履歴から求めた入力領域の評価

7.1 タップ実験との比較

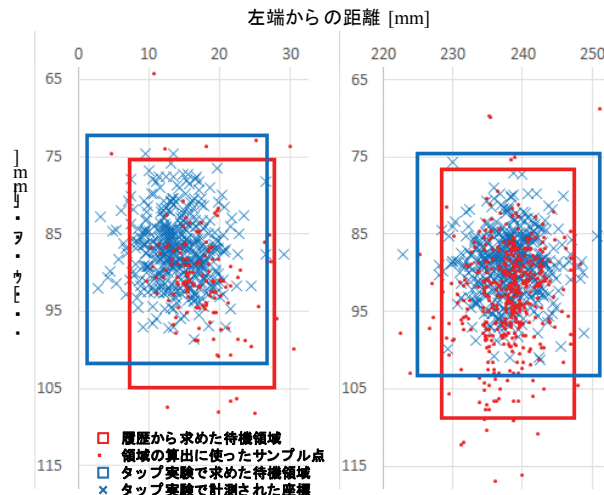
図 9 の赤の矩形は被験者 A の 1 か月間の使用履歴から前述の方法で求めた待機領域である。また、赤の点は領域の算出に使ったサンプル点である。この実験では、10 秒以上入力が途切れた直後のタッチ位置を抽出し、そこから、縦横それぞれで平均から遠い 5%を除いたものをサンプル点としている。図から、待機領域が赤の点を囲うように設定されていることがわかる。遠方の点を除かないと待機領域が過大になる。表 3 では遠方の点を除く場合と除かない場合を比較しているが、点を除くことで待機領域の面積が 1/3 程度に縮小している。

図 9 の青色の点は同じ被験者 A のタップ実験の座標で、水色の矩形はそれから求めた待機領域を示している。右側では赤と青の矩形がほぼ重なっている。青色の点の 97.6%が赤の矩形の中に含まれているので、ほぼ間違いなく待機領域から復帰できる。一方、左側では赤の矩形が青より小さくなっている。これは、使用履歴から求めたサンプル点の位置がタップ実験の分布の右下に偏っているためである。このため、青点が赤の矩形に含まれる割合は 92.2%と、少し低い値となった。実際には文字入力(右側)で待機状態から復帰することが多い。この実験でも、使用履歴から求めたサンプル点の数は右側が 305 点、左側が 150 点で、右が多くなっている。同じ比率で待機領域をタッチするとすれば、95.8%の確率で復帰できることになる。

左右ともに、履歴から求めた待機領域の横幅はタップ実験から求めた横幅より小さくなっているが、これは、次に行う動作が影響したためだと考えられる。タップ実験ではタップすれば終わりだが、文字入力ではそこからストロークを行うため、その分の指を動かす余裕が必要となる。左右の動きは親指の曲げ伸ばしで行うため、上下に比べて可動範囲が狭いので、指を動かしやすい場所にタッチする必要がある。このため、タッチ位置が中央に集まったのではないかと想像される。

7.2 縮小領域の計算に必要なサンプル数

次に、別の被験者の使用履歴を用いて、待機領域を求めるために必要なサンプル点の数の目途をつける実験を行った。この実験では、タッチ位置を漏らさずに取得するために、入力領域の自動縮小機能を OFF にして入力を行っている。また、サンプル点の数を増やすために、4 秒以上入力が途切れた直



(a) 左側の待機領域 (b) 右側の待機領域
図 9 タップ実験で求めた待機領域との比較

表 3 遠方除去による待機領域の縮小の効果

	左側			右側		
	縦幅	横幅	面積	縦幅	横幅	面積
全点(A)	56.0	30.6	1714	64.3	27.4	1762
遠方除去(B)	24.9	19.8	493	30.6	20.3	620
比(B/A)	0.44	0.65	0.29	0.48	0.74	0.35

※長さの単位は[mm], 面積は[mm²]

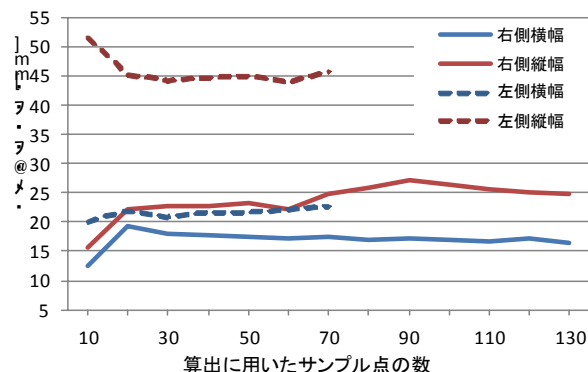


図 10 サンプル数を増やしたときの領域の幅の変化

後のタッチ位置を抽出している。縦横それぞれで、遠方から 5%の点を除いたのち、サンプル点を入力時刻の順に並べ、古いほうから一定数を用いて待機領域を求める。この処理を使う点の数を変えて行い、待機領域の大きさの変化を調べた。結果を図 10 に示す。右手の縦幅を除いて、30~50 サンプルで変化がほぼ収束している。

先頭から 40 サンプルを使って求めた待機領域を図 11 に示す。図の赤点は待機領域の算出に用いたサンプル点を示している。これまでの図と同様に、画面の左上を原点とし、右を X 軸の正の向き、下を Y 軸の正の向きとして、mm 単位で表示している。ただし、この実験では ASUS VivoTab Note 8(画面サイズ 8inch)を用いてデータを収集しているため、サンプル点

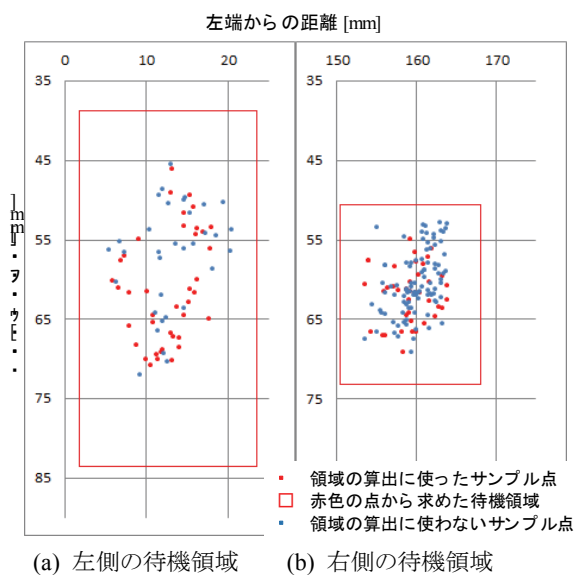


図 11 40サンプルから求めた待機領域

の座標はこれまでの実験とは異なる。

右側ではサンプル点を囲むように領域が設定できており、隙間も少ないので、待機領域が適正に設定できていると言える。左側では縦幅が少し大きく、隙間も空いているが、これは、サンプル点が上下方向にばらついているためである。右側の行選択では 1 ストロークのジェスチャーを使わないが、左側では矢印キーの入力に使っている。1 ストロークジェスチャーでは指を動かしながらタッチすることが多いため、左側のサンプルが上下にばらついたらと考えられる。なお、右側でも段選択で 1 ストロークジェスチャーを入力するが、行選択に続いて行うため、サンプルとして抽出される可能性は低い。

待機領域の算出に使わなかったサンプル点(図 11 の青色の点)が求めた領域に含まれる割合を調べたところ、右側 98 点と左側 32 点のすべてが待機領域に含まれていた。この実験では、縮小しても領域外をタッチすることは無かったといえる。この実験と前の実験の結果を合わせると、提案手法により個人ごとに適切な縮小領域の設定が可能だと考えられる。

7.3 入力領域の縮小率

図 12 に改良後のインターフェースの右半分を示す。通常の入力状態では、(a)のように入力領域とガイドが表示される。待機状態に入ると、(b)のようにガイドが消え、入力領域が小さくなる。待機状態の領域の大きさは利用者により異なるが、図 11 では表 2 に示した値の領域を表示している。この場合、左右両方を合わせても画面の 3%未満の面積しか占有しない。

8. 待機領域の背後へのタッチ方法

入力領域を縮小しても、縮小された領域(待機領域)の背後には依然としてタッチできない。しかし、待機領域の位置は目視せずにタッチできることを条件として決めているので、移動することはできない。そこで、待機領域をダブルタップすること

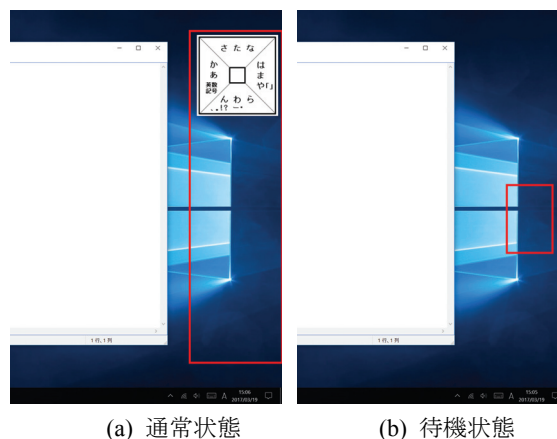


図 12 提案手法のインターフェース

で、ダブルタップした側の領域を消す機能を実装している。こうすることで、画面のすべての場所をタッチできるようになる。消えるのはダブルタップした側の領域だけなので、残ったほうをタッチすることで、通常の入力状態に復帰できる。また、文字の入力領域(デフォルトでは右側の入力領域)をダブルタップすることで、利用者の意思で領域を縮小できる機能も実装している。

9. まとめ

ジェスチャーを使用したタブレット向けの文字入力手法において、入力領域の背後がタッチできない問題を解決する方法として、一定時間入力が必要であればジェスチャーの入力領域を自動的に適切な大きさに縮小する機能を追加した。

本論文では、タップ実験により画面を最初にタップする位置や分布の傾向を求め、計測したタップ位置から、個人別に、視認せずにタッチできる領域の位置と大きさを求める方法を提案した。さらに、各自の利用履歴を使ってこれらの値を求める方法を提案した。実験により、利用履歴を使って求めた縮小領域の中にタップ実験で得たサンプル点の約 95%が包含されることを確認した。この結果から、利用履歴を使っても、視認せずにタッチできる領域を求めることができるといえる。

改良後の文字入力手法では、入力領域が縮小されていても指が触れた時点で通常の大きさに戻り、座標も引き継がれるので、縮小しない場合と同じ使い勝手でジェスチャーを入力することができる。

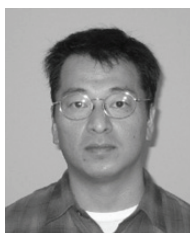
(本研究は JSPS 科研費 16K00286 の助成を受けています。)

参考文献

- [1] Windows 10 搭載タブレット PC のタッチキーボードの使い方。必ず覚えておきたい基本操作と設定方法 <http://www.ask-mswin.com/win10-touchkeyboard-tablet/> (2016/12/28 確認)
- [2] A.I.タブレットキーボード (A.I.type Tablet Keyboard) : <https://play.google.com/store/apps/details?id=com.aitype.android.tablet> (2016.3.3 確認)

- [3] 箱田博之, 志築文太郎, 田中二郎: タッチパネル端末向け縦型 QWERTY キーボードの性能評価, 情報処理学会研究報告, Vol.2014-HCI-156, No.18, pp.1-7 (2014)
- [4] 田中正人, 田川欣哉, 山中俊治: 新型親指キーボード Tagtype の開発研究, 機械力学 計測制御講演論文集, pp. 626-631 (2001)
- [5] Tagtype for VAIO: <http://www.vaio.sony.co.jp/Enjoy/Download/Tagtype/> (2016.9.3 確認)
- [6] 金子雅司, 岩田満: タブレット端末における T9 フリックキーボードの提案と実装, 平成 26 年度電子情報通信学会東京支部学生会研究発表会予稿集, p.223 (2015)
- [7] A. Oulasvirta, et al. : “Improving Two-Thumb Text Entry on Touchscreen Devices”, CHI '13: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp.2765-2774 (2013)
- [8] Gimpei Kimioka, Buntarou Shizuki, and Jiro Tanaka: “Niboshi for Slate Devices: A Japanese Input Method Using Multi-touch for Slate Devices”, Proceedings of HCI International 2011, Part I, pp.81-89 (2011)
- [9] 横井孝祐, 田中敏光, 佐川雄二: タッチデバイスのためのマルチタッチを用いた文字入力手法, pp. 49-50, モバイル'12 研究論文集 (2012)
- [10] 深津佳智, 志築文太郎, 田中二郎: No-look Flick: 携帯情報端末のタッチパネルにおけるアイズフリーな片手かな文字入力システム, WISS2012(第 20 回インタラクティブシステムとソフトウェアに関するワークショップ) 論文集, pp.133-138 (2012)
- [11] 青木良輔 他: Drag&Flick: タッチスクリーンを用いた視覚障がい者向け文字入力方式, 情報処理学会インタラクティブ 2013, pp.72-79 (2013)
- [12] 井川洋平, 宮下芳明: アイズフリーで速記できる「方向のみ」のフリック入力手法, 情報処理学会インタラクティブ 2013, pp.651-656 (2013)
- [13] 田中雄也, 田中敏光, 佐川雄二: タブレットの両側を把持した状態で親指のジェスチャーにより文字を入力する手法の提案, 電気学会論文誌 C, Vol.137, No.2, pp.370-378 (2017)

至る。CG の研究・教育に従事。情報処理学会。



佐川 雄二(非会員)

1987 名古屋大学大学院情報工学専攻前期課程修了。1992 名古屋大学大学院情報工学専攻後期課程単位取得退学。工学博士。1992 名古屋大学助手, 同講師を経て, 2000 名城大学理工学部講師, 2008 同教授。現在に至る。自然言語処理の研究・教育に従事。電気学会。

著者紹介



秋田 光平 (学生会員)

2017 年 3 月名城大学理工学部情報工学科卒業。現在, 名城大学理工学部情報工学専攻に所属。タブレットやスマートフォンの文字入力手法の研究に従事。



田中 敏光(正会員)

1984 名古屋大学大学院情報工学専攻博士(前期)修了。工学博士。1984 NTT 入社。1994 名古屋大学大型計算機センター助教授。2000 名城大学理工学部情報科学科(現情報工学科)教授。現在に