

# 拡張現実のための直感的クリックインタフェース

杉浦 篤志<sup>†</sup> 豊浦 正広<sup>†</sup> 茅 暁陽<sup>†</sup>

**概要:** 拡張現実においてボタンのクリック動作は基本で最も重要なインタラクションである。単独のカメラ画像から奥行きを推定することは難しく、直感的なクリック動作を認識することは困難である。従来は、認識しやすいジェスチャとして親指と人差し指を挟むという特別な動作をクリックに割り当てていた。我々は単独のカメラによる直感的なクリックインタフェースを提案する。隣接するフレーム間の時間差分に注目し、指先の速度と加速度を推定する。動作状態を分類し、クリック動作検出のために状態遷移を認識する。我々はまずユーザがどのように仮想ボタンをクリックするのかわかり、次にクリック動作の検出アルゴリズムを設計、実装した。また、クリック検出アルゴリズムの有用性を文字入力および電卓インタフェースによる実験により確認した。

## 1. はじめに

本研究では、拡張現実環境で用いる自然な入力インタフェースを提案する。カメラで利用者の指先を検出し、その速度および加速度を調べることで、クリックの動作を検出する。従来手法では、指先の位置(0次情報)のみに注目してきたために、指先の動作の認識が難しく、トポロジー変化のないようなクリック動作の認識は実現されなかった。我々は、速度(1次微分量)および加速度(2次微分量)に注目することで、クリック動作の検出を可能にした。

拡張現実関連の技術は、携帯端末アプリケーションなどにも広く一般に利用されるようになり、社会的な注目が集まっている。Googleは2014年当初にヘッドマウントディスプレイ GoogleGLASS[1]を発売予定である。GoogleGLASSは外向きに取り付けられたカメラと、内向きに取り付けられた透過型ディスプレイを備えており、拡張現実のアプリケーションサービスを行うには格好の装置である。カメラによるジェスチャ認識の難しさから、GoogleGLASSは専ら音声入力によるコマンド実行や、眼鏡のフレームの部分に取り付けられた小さなタッチパネルによって、コマンドの実行を行うようである。ディスプレイに提示される仮想世界と、カメラに映る現実世界をシームレスにつなぐためには、ジェスチャによる仮想物体の直接操作が欠かせないと、著者らは考える。

カメラを使った自然な入力装置(NUI, Natural User Interface)についても、タンジブルインタフェース[2-3]、テーブルトップインタフェース[4-5]、プロジェクトカメラシステム[6-7]、Kinect[8]、iPad[9]などが牽引役となり、活発な議論が行われている。中でもプロジェクトカメラシステムでは、手指の動作をカメラ画像から認識することで、コマンド入力を行う手法が提案されてきた。カメラから得られる手指の領域は、安定した照明環境が得られなかったり、背景領域に想定が置けなかったり、手指による自己隠蔽が

大きかったり、動きぶれも大きかったりすることから、欠損のある不完全なものであることが多い。これらによる問題を避けるために、従来のプロジェクトカメラシステムでは、自然さと頑健性のバランスの取れた、手指の形によるコマンド入力[10]やピンチ動作による平面位置指定[11]などで採用されてきた。

我々は、利用者視点で観測される1台のカメラから手指の自然な動作を認識することを目指す。このうち本研究では、最も基本的な動作として、クリックを認識することを目標とする。クリック動作は、上述のコマンド入力やピンチ動作よりもより自然であるが、そのパターンの認識が困難であり、頑健性を確保することが難しい。また、1台の日常空間を移動するカメラを想定しており、従来の拡張現実アプリケーションに見られるようなほぼ完全な手指の領域が得られることも想定しないし、複数カメラから三次元的な情報が得られることも想定しない。これにより、GoogleGLASSの利用が想定されるような環境での適用を可能にする。

以降、第2章では関連研究を挙げ、本研究との関連について説明する。第3章では提案手法を述べる。第4章では、評価実験を示し、第5章で本研究をまとめる。

## 2. 関連研究

画像からの手のジェスチャ認識はコンピュータビジョンの古典的な問題のひとつであるが、まだ多くが未解決である。近年の手の画像認識手法については、[20]のサーベイ論文に詳しい。

画像からの手のジェスチャ認識が難しい理由は、1)関節が多いために形状変形の自由度が高く、形状からの推定ができない、2)関節が多いために、自己隠蔽が起こりやすい、3)照明変動によって、肌色による領域抽出が難しいなどの理由による。

仮想物体とのインタラクションを実現するとき、手指の三次元位置を獲得するのが、最も正攻法の方法である。ステレオカメラを利用し、それぞれの画像を取得し、特徴点

<sup>†</sup> 山梨大学  
University of Yamanashi

を対応させる。対応させた特徴点とステレオカメラの内部パラメータを利用することで画像内の三次元位置を獲得する。Leeら[12]による仮想物体と手とのインタラクションの研究では、カメラから取得した画像より肌色領域を抽出する。ステレオカメラによる視差を利用することで手の重心と指先の三次元位置を取得する。その二点を結ぶ線分により手の方向を算出し、その線分と仮想物体の接触判定を行うことで手と仮想物体とのインタラクションを実現している。

また、加茂ら[13]はステレオカメラにより指の三次元位置を求め、手のジェスチャにより仮想画面を表示させる。別のジェスチャにより画面を拡大縮小することも可能である。仮想画面と手とのインタラクションでは仮想画面の手前から奥へ指を交差させることで交差点にあるオブジェクトを操作することができる。しかし、これらのシステムはステレオカメラを使用するために事前にカメラ位置の情報が必要であり、設備が大掛かりとなり持ち歩くことが困難である。

三次元位置の獲得が難しいと判断すれば、特定のジェスチャにコマンドを割り当てて、仮想物体操作を実現することもできる。Kimら[14]の研究はプロジェクタに投影された仮想物体を手のジェスチャによって操作するシステムである。手指の姿勢推定はユーザの指先にマーカとなる指サックを取り付ける。ユーザは特定のジェスチャを行い、そのジェスチャをカメラで取得し、指サックの位置を判定することでジェスチャに応じた仮想物体の操作を行う。しかし、指サックを装着する制限があり、特定のコマンドのジェスチャを習得しなくてはならないため直感的な操作には不十分である。

Kölschら[15]の研究では様々な手法を組み合わせることで頑健な手の検出およびジェスチャ認識を実現している。手には何も装着することなく特定のジェスチャを認識することで仮想物体操作を実現している。しかし、コマンドだけでは、仮想物体を直接操作するためには不十分である。

1台のカメラから自己隠蔽の大きな手指の姿勢を獲得しようとする試みをWangら[19]は行った。Wangらの研究では独自に配色されたカラーグローブを装着する。カメラからグローブを装着した手の画像を取得する。手の領域の重心を最近傍法により推定し、仮想の手のモデルに対応させて仮想物体を操作する。手に合わせた直感的な仮想物体操作が可能となる。しかし、正確な手の形状を推定するためには、グローブが必要となる。ユーザの動作にはグローブの装着のため制限があり拘束感を与えてしまう。

LeapMotion[17]では、赤外線センサを下から照射することで、平面内の指先位置を検出し三次元の指先の動作をコンピュータに入力するための装置である。これを利用することで指の動きをコンピュータ内で再現したり、宙で記載した文字がコンピュータ内にも表示されることが可能とな

る。1/100ミリ単位の動きまで認識可能である。赤外線センサを照射する装置が必要となり利用環境も制限される。

加速度からの動作推定では、Aki[18]によるWiiリモコンを利用したジェスチャ分類がある。Wiiリモコンは任天堂より発売された据え置き型ゲーム機Wiiのコントローラで動きを検知する3軸加速度センサが搭載されている。その加速度センサを利用し、手の動作を認識している。18種類の動作を識別することで様々な動作を可能としている。手にリモコンを装着しなければならないが加速度を利用することはジェスチャ認識に有効であり本研究でも採用したいと考える。

速度・加速度を使うことで、手の正確な三次元位置を求めずにジェスチャ動作を認識したい。そこで本研究ではカメラ画像から取得した手指領域の隣接するフレーム間差分より手指の速度・加速度を算出し、手の動作を推定する。

### 3. クリック動作の認識

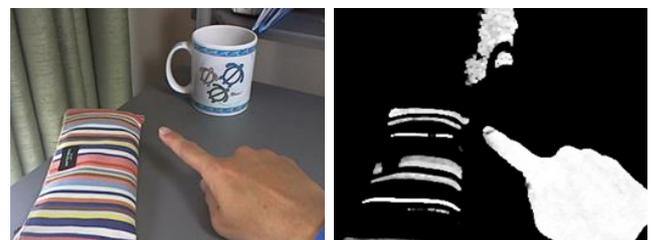
#### 3.1 指先の認識

各フレームにおける指先位置を調べるためには、まず肌色領域抽出を行う。最初のフレームでは、HSV色空間上で代表的な肌色  $S=(h_s, s_s, v_s)$  が与えられ、 $S$  から一定距離内にある領域が肌色として判定できる。最初のフレームでは、代表的な肌色の値  $S$  や許容閾値は先験的に与えるしかない。各成分に対する許容閾値を  $h_{th}, s_{th}, v_{th}$  とすると、ある色  $P=(h_p, s_p, v_p)$  を持つような画素が以下のすべての式を満たすときに、その画素が肌色であると判定される。その結果を図1(b)に示す。

$$h_s - h_{th} \leq h_p \leq h_s + h_{th} \quad (1)$$

$$s_s - s_{th} \leq s_p \leq s_s + s_{th} \quad (2)$$

$$v_s - v_{th} \leq v_p \leq v_s + v_{th} \quad (3)$$



(a) 原画像 (b) 肌色領域

図1 肌色領域分割

手指以外にも肌色を持つ領域は映っていることが考えられるので、領域の大きさによって手指の領域を絞り込む。肌色であると判定された領域はラベリングによって、最大の面性を持つ領域のみを手指の領域であるとし抽出するものとする。その結果を図2(a)に示す。

2 フレーム目以降は、最初のフレームで手指の領域として抽出された領域に含まれるがその値を学習データとして、平均 $\mu=(\mu_h, \mu_s, \mu_v)$ と分散協分散行列 $\Sigma$ を求めることで抽出する。前フレームまでの肌色領域の値をこれらに反映させることで、その環境に応じた肌色抽出が可能となる。求められた平均と判定対象となる画素の色  $P$  とのマハラノビス距離を以下のように計算することで、肌色かどうか判定できる。 $d_{th}$  はマハラノビス距離に対する閾値である。

$$(P - \mu)^T \Sigma^{-1} (P - \mu) \leq d_{th} \quad (4)$$

指先中心は手指の領域の先端部分だけを見ることで、指の腹の位置を調べる。爪の領域が肌色であると判定されないことがあり、指先に穴があくことがあるので、モルフォロジー処理によって穴を生めておく。画像の左上を原点、 $y$  軸を垂直下向きに取るとき、肌色領域に属するがそのうち、 $y$  が最小となるような画素を仮に指の先端であるとする。手指は画像の下から上に伸びることが期待できるためである。指先の画素を中心に半径  $R$  の範囲を取る。 $R$  はあらかじめ設定するものとする。範囲内の手指領域画素について、距離変換を行うことで、手指領域境界が 0、奥に行くに従い値が大きくなるような値を設定する。その結果を図 2(b) に示す。指の腹は、求めた範囲の中央に位置することが期待できるので、2 次曲面当てはめを行うことで、大域的に見たときの距離変換値のピーク位置を求め、これをそのフレームでの指先位置とする。



(a) 手指領域抽出 (b) 指先領域の距離変換

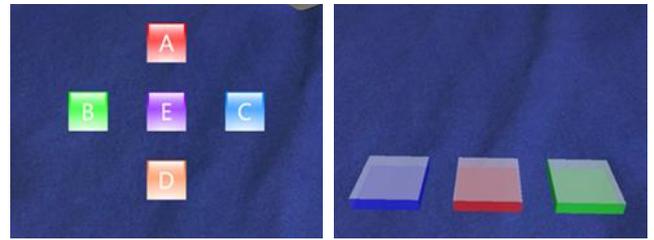
図2 手指領域の指先位置推定

### 3.2 仮想ボタンの押し方に関する予備調査

3.1 章で求めた指先位置の情報から、どのようにクリック動作を検出するかを検討するために、予備実験を行った。被験者 12 名(70 代 1 名, 60 代 2 名, 30 代 3 名, 20 代 6 名) がどのように仮想ボタンをクリックするかについて調べた。被験者には映像透過型 HMD を装着してもらい、ディスプレイに表示される仮想ボタンを押すように指示し、具体的な操作の仕方については何も指示を与えなかった。

ボタンの配置による押し方の違いを検証するために、仮想ボタンは図 3(a) のような画像平面と平行に配置するものと、図 3(b) のように画像平面とは異なる平面に配置するもの

の 2 種類を用意した。それぞれの仮想ボタンは、被験者の動作に対して、いかなる変化もしない。これは、ボタンクリックの誤認識・誤検出によって、押し方が影響を受けないようにするためである。



(a) 平面的ボタン (b) 立体的ボタン

図3 仮想ボタン形状

予備実験からは、すべての被験者で共通した以下のクリック動作が観測された。

- 被験者には仮想ボタンの深さ情報が把握できないために、被験者ごとに異なる奥行き位置でボタンを押す動作を行った。また、被験者は指を画像平面と平行に移動させて、ボタンと指先の画面上での位置を重ねた。
- ボタンを押す場合にも同様に、ボタンと指先の前後関係が把握できないために、指を素早く振り上げたり、素早く振り下ろしたりして、仮想ボタンを押そうとした。被験者のうち 11 名は、図 4(a) に示すようにまず指をゆっくり振り上げ、次に素早く振り下ろした。被験者のうち 1 名は、逆に、図 4(b) に示すようにまずゆっくり振り下ろし、次に素早く振り上げた。
- ボタンの位置によって、指先位置の水平成分の変化量が異なる。これは、ボタンを押す方向が、画面内で一定でないためだと考えられる。

以上のことから、以下のように設計することで自然なクリック動作の検出ができると予想できる。

- 異なる奥行き位置でのクリック動作を検出する必要がある。画面上の指領域の大きさによって、おおよその指の奥行き位置を求めて、クリック動作の検出アルゴリズムを導入する。
- 指先とボタンの位置関係を認識して、その時点で仮想ボタンを押すことができるかどうかを利用者に可視化する。これにより、ボタンを押せることのアフォーダンスを利用者に提供する。
- 素早く振り下ろす、素早く振り上げるといった、急減速する動作を検出することで、クリックを認識する。

指先の水平位置およびその変化は、同じクリック動作に対しても異なる値を取るために、それだけではクリック動作の検出は難しい。

### 3.3 速度および加速度によるクリック動作検出

従来のジェスチャ認識では、2 次元位置の軌跡(trajecory)

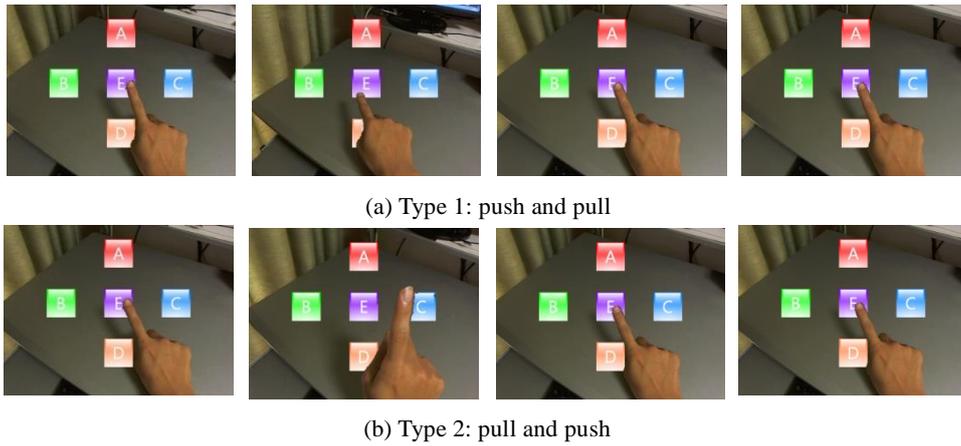


図4 クリック動作例

やその変化量(1次微分量)に着目したものがほとんどであった。しかし、前述の予備実験結果の考察によると、奥行きの変化量や、位置の加速度(2次微分量)に、クリック動作を認識するのに重要な情報が含まれていることが示唆された。従来のジェスチャ認識では、以下のような信号を利用していたことになる。

$$\{x_i, x'_i\}_{i \in [t-(n-1), t]} \quad (5)$$

ただし、 $x_t$  は時刻  $t$  における 2次元位置、 $x'_t = \partial x_t / \partial t$  は時刻  $t$  における 2次元位置の 1次微分量を表す。時刻  $t$  でクリック動作が行われたかどうかは、時刻  $t$  から過去  $n$  フレームの 0次および1次の情報のパターン認識の問題として扱われてきた。これに対して、2次微分量  $x''_t = \partial^2 x_t / \partial t^2$  と奥行き位置  $d_t$  も考慮することにする。

$$\{x_i, x'_i, x''_i, d_i\}_{i \in [t-(n-1), t]} \quad (6)$$

ただしここで、単独のカメラからは奥行き位置  $d_t$  は正しく得られないことに注意されたい。そこで画像上で観測される指領域の太さ  $w_t$  でこれを代用するものとする。

$$\{x_i, x'_i, x''_i, w_i\}_{i \in [t-(n-1), t]} \quad (7)$$

以下に具体的なクリック動作検出手法を示す。

1. ボタンを押せることの通知

あるボタンが画像上で占める範囲を  $B \in \{R^2\}$  とする。  $B$  は画像上にボタンがレンダリングされるときに、簡単に求めることができる。時刻  $t$  での指先位置  $x_t$  が  $x_t \in B$  であれば、そのボタンを押すことができる状態であると判定する。

仮想ボタンと指先の位置関係は、被験者自身も把握することが困難でありユーザに仮想ボタンと指先の位置関係を提示する視覚フィードバックが必要となる。そこで、図 5

に示すように、指先と仮想ボタンが重なったときに、視覚的な変化をユーザに提供することでそのボタンが選択されていることを通知する。

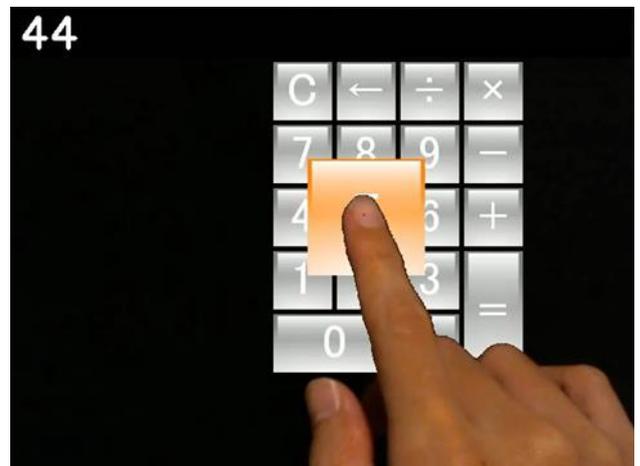


図5 ボタンの視覚的フィードバック

2. 指先の奥行きおよび加速度からの動作検出

予備実験により得られた知見から、急減速するとき利用者がクリック動作を意図すると予想される。加速度  $x''_t$  によって、急減速を判定したい。ただしこのとき、3次元空間中で同じ加速度を持つジェスチャを行った場合でも、奥行き位置によって画像平面上で観測される加速度が異なることを考慮せねばならない。一方で、単独のカメラからは正確な指先の奥行き位置を求めることはできない。そこで、これらを考慮した値を得る近似的な方法として、図 6 に示す指先中心円の時間差分量による推定を提案する。

図 6 には、隣接する 2 フレームにおける指先中心位置  $x_t$  および  $x_{t-1}$  を示している。これを中心にして、半径  $w_t$  および  $w_{t-1}$  を持つ円を考える。  $w_t$  は時刻  $t$  の指先位置から 5 つの半径を持つ同心円上での指先領域が横切る長さから、中央値を求めた値である。この 2 つの円の差領域の面積は、指先の 3次元移動量が大きいほど大きく、また、奥行き位

置が大きいかほど小さくなる値となる。求められる値は必ずしも指先の3次元移動量に比例するとは言い難いが、少なくとも正の強い相関を持つ値を示す。この関数は我々が経験的に得た関数であり、その正当性については、実験結果によって示す。

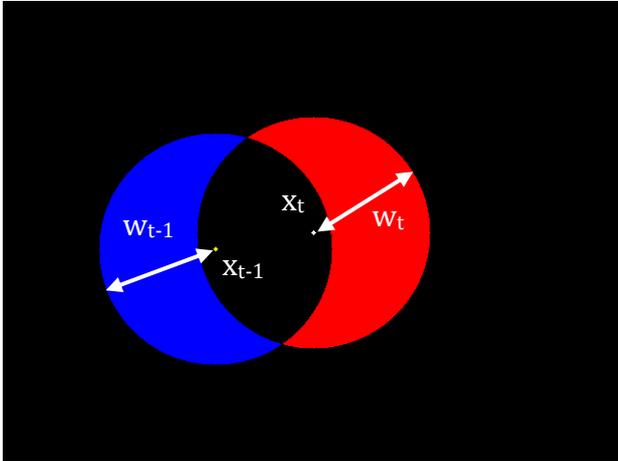


図6 指先中心円の時間差分

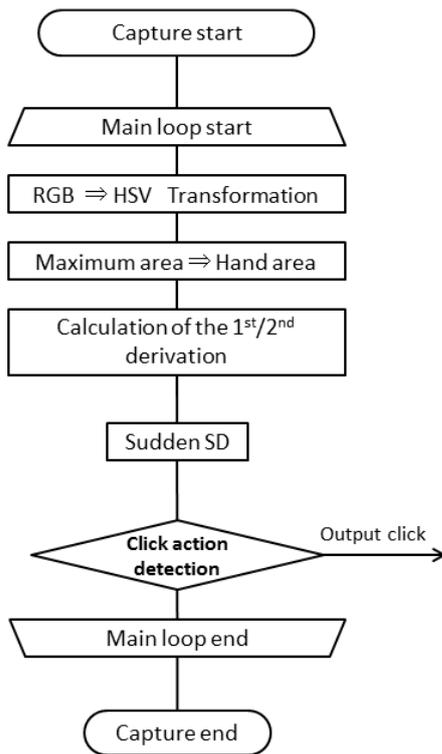


図7 システムフロー

これらの処理を取得した画像ごとに繰り返し行う。そのシステムフローを図7に示す。指先領域を抽出し、指先中心位置から円の差領域より1次微分量、2次微分量を求める。2次微分量の変化よりクリック動作を認識する。本章ではクリック動作のみについて検出を行うために、一般的によく利用される1次微分量  $x'_t$  は利用しないが、今

後、その他のジェスチャ動作認識への拡張を考え、次章では1次微分量  $x'_t$  を考慮したクリック動作認識について検証する。

### 3.4 クリック動作の拡張

仮想物体を操作するための入力にはクリック動作以外にもドラッグ操作等が挙げられる。将来的にはドラッグ操作などの拡張も考慮し、ドラッグ操作との識別が必要となる。急減速によるクリック動作の認識は可能であるがクリック動作以外の入力操作であるドラッグ動作等との区別をするためにはより条件を増やす必要がある。クリック動作を予備実験より得られた4つの状態で認識が可能かどうかの判定を行った。

3.2章の予備実験よりクリック動作が停止状態から次に通常動作状態、速い動作状態、最後に停止状態へ状態が移行していくため、加速度変化のみではなく速度変化も考慮することで他の動作との識別の向上を図る。図8に示す状態遷移により一連の状態変化が実行されるときにクリック動作判定を行う。

図8に示すように、手指の速度、加速度の変化により停止状態、通常動作状態、速い動作状態、急減速状態の4つの状態に分類する。停止状態と通常動作状態の判定には速度変化を利用し、一定の閾値を超えると停止状態から通常動作状態へ移行する。通常動作状態と速い動作状態の分類には速度と加速度の変化を利用する。また、最後の急減速状態は加速度の変化を利用し、閾値により急減速状態を分類する。これらによりクリック動作を4つの状態の一連の変化により他の動作と区別することが可能となる。

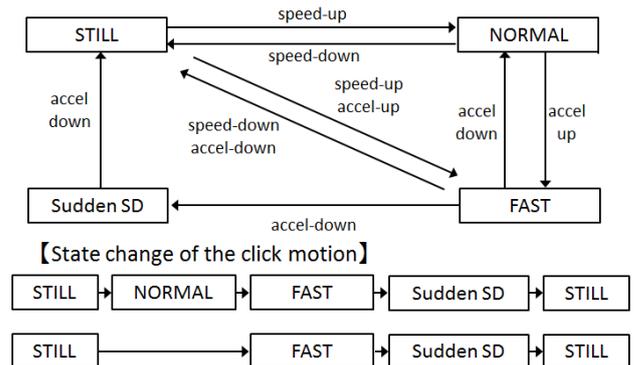


図8 状態遷移

## 4. 実験

4.1.1 仮想ボタンと指の位置関係を視覚的に提示し加速度による直感的なクリック動作認識のインタフェースの評価実験を行った。まず、指先の加速度変化とボタンの視覚変化によるクリック動作の評価実験を行い、その有効性を確認した。また、一般的な利用を想定しパソコン操作に不慣

れな年代の被験者による評価実験を行い、直感的なクリック動作によるシステムの有用性を確認した。最後に加速度変化だけでなく速度変化も考慮したクリック動作認識の評価実験を行った。

#### 4.2 実験環境

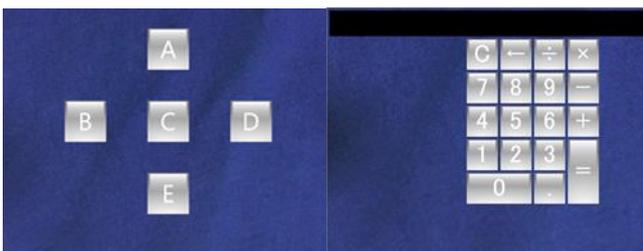
操作風景を図9に示す。パソコンを1台準備しカメラ付HMDを接続する。入力装置としてHMDのカメラを利用する。HMDにはVuzix社製のWrap920ARを使用した。解像度はSVGA(800×600)でディスプレイ正面にはUSBビデオカメラが2台組み込まれており、2眼式のVGA(640×480)の映像を取得することができるが本システムでは単独のカメラによる入力とするため1台のカメラのみ利用する。



図9 操作風景

#### 4.3 評価実験

カメラと手指の位置関係により手指の映り方が変化し、クリック動作の方向も変化する。そこで仮想ボタンを十字型に配置することでクリック動作の方向の変化の影響も確認する。そこで図10(a)に示すA~Eまでの文字を入力できる仮想ボタンを画面内に十字型に配置した文字入力インタフェースを設計した。次により一般的な利用を想定し仮想ボタン同士を近くすることで隣接することによる干渉の確認を行うため図10(b)に示す電卓型インタフェースを設計した。また、実験の前に被験者には本手法の説明を行い、1~2分程操作の練習を行った。



(a) 文字入力 (b) 電卓

図10 検証実験インタフェース

20歳~30歳代の男女5人の被験者に本システムを使用し、クリック動作の検証実験を行った。タスク1では図10(a)の文字入力インタフェースを利用する。ユーザには10個のA~Eの文字列を入力してもらうタスクを5回行う。また、タスク2では図10(b)の電卓インタフェースにより3桁の四則演算を10回行ってもらう。それぞれのタスクの操作中のクリック動作の誤入力fと未入力nの回数を集計し、全入力数Aに占める誤字数の割合である誤り率Eの比較を行った。誤り率よりクリック動作認識の正確性を確認する。その結果を表11に示す。

$$E = (f + n) / A \times 100 \quad (8)$$

表11 検証実験結果

		入力文字	クリック(回)	誤入力(回)	未入力(回)	誤り率(%)
5人平均	文字	50	53.8	0.6	3.2	7.06
	電卓	80	84.4	0.4	4.0	5.21

検証結果より誤り率が8%以下であった。タスク1の文字入力インタフェースによりカメラと指先や仮想ボタンの位置変化によるクリック動作の方向変化による動作判定の影響はなかったことが確認できた。タスク2の電卓インタフェースの隣接する仮想ボタンへの影響の確認では仮想ボタンの選択時に拡大表示されることで判定領域を広げているためより正確にクリック入力することが可能であったことが確認できた。また、指先位置から同心円上での指先領域を横切る長さから求めた値の円による差分領域の面積の正当性も検証結果より正当であることが確認できた。

入力動作が現実空間でのボタンを押す動作と類似しておりユーザに理解してもらいやすかった。ボタンとの接触については触覚による感覚が全く無いがボタンの色変化による視覚的フィードバックのみでもユーザにとって十分、押す感覚を提示することが可能であることがわかった。

さらに一般的な利用を想定し、パソコン操作に不慣れた年代のユーザである60-70歳代の男女2人、小学生の男女3人に文字入力インタフェースによるA~Eの文字入力タスクを3回、電卓インタフェースによる計算タスクを5回行ってもらった。実験風景を図12に示し、その結果を表13に示す。

誤り率は18%程度に留まったが現実世界のボタンの押す動作と類似しているため、パソコン操作に不慣れた年代でもクリック動作を短時間で理解してもらうことができ、直感的な入力動作であることが確認できた。また、HMDという装置も触れたことがあまりないため眼前に提示されるディスプレイ操作に不慣れた部分があった。



(a) 70代男性 (b) 小学生

図12 実験風景

表 13 年代別検証実験結果

		入力 文字	クリッ ク(回)	誤入力 (回)	未入力 (回)	誤り率 (%)
60-70 歳代	文字	30	36.0	1.0	5.0	16.67
	電卓	40	48.5	2.5	6.0	17.53
小 学 生	文字	30	37.7	2.0	5.7	20.35
	電卓	40	49.6	2.3	7.3	19.46

#### 4.4 クリック動作の拡張の評価実験

4つの状態変化によるクリック動作判定の検証実験を行った。20歳～30歳代の男女3人に文字入力タスク5回と計算タスク10回を行い、そのときの誤入力、未入力の回数を計測し誤り率を算出した。表14に結果を示す。

表 14 4つの状態遷移による検証実験

		入力 文字	クリッ ク(回)	誤入力 (回)	未入力 (回)	誤り率 (%)
4つの 状態	文字	50	61.0	5.3	5.7	18.03
	電卓	80	97.0	7.7	9.3	17.52

誤り率は18%程度であった。これはユーザごとにクリック動作の速度、加速度がことなるため、4つの状態をうまく識別することができず、誤入力、未入力の回数が増えてしまったためであると考えられる。

停止状態、通常動作状態、速い動作状態、急減速状態の4つの状態を分類するためにそれぞれに閾値を設定する必要がある。実際のユーザのクリック動作を観測したデータよりある一定の閾値を決定していたがユーザそれぞれにクリック動作をはじめとする指の速度、加速度はわずかに異なる。そのため、ユーザによりクリック動作の認識の差異が生じていた。キーボード入力により閾値の値を一定の割合で増減させることでユーザによる速度、加速度の差異に対応していたが設定の時間と手間がかかっていて有効な手段ではなかった。

そのため、図15のようなユーザの指の停止、指の移動、クリック動作を観測し、そこから閾値を推定するシステムを構築した。これによりユーザは本システムを利用する前

に閾値推定システムを使用してユーザごとの閾値を推定し、本システムに設定することでクリック動作の認識率の向上を可能とする。

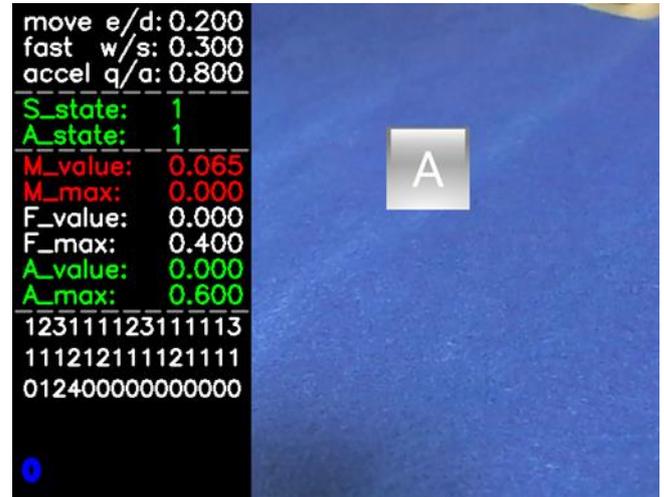


図15 クリック動作の閾値推定システム

ユーザのクリック動作による閾値推定システムの有効性を確認するための実験を行った。実験タスクは文字入力のタスクと電卓による計算タスクを行った。そのときの操作中の誤入力とクリック動作の未判定の回数を集計し、誤り率の比較を行った。その結果を表16に示す。

表 16 閾値設定システムによる改善の実験結果

		入力 文字	クリッ ク(回)	誤入力 (回)	未入力 (回)	誤り率 (%)
閾値	文字	50	58.3	2.3	6.0	14.29
	電卓	80	95.3	4.7	10.6	16.08

表16の結果より閾値設定のシステムを利用し、ユーザに合わせた閾値を設定することでより文字入力タスク、電卓による計算タスクともにクリック動作の認識率の向上を確認することができた。閾値を合わせることで誤入力、未入力の回数を減らすことができた。

## 5. おわりに

本研究では拡張現実を利用して現実空間に仮想ボタンを表示させ、直感的な動作でクリック入力するインタフェースを提案した。指の状態を2次微分による急減速の動作を認識することでクリック動作を判定することが可能となった。仮想ボタンの触覚的フィードバックがないため、視覚的フィードバックであるボタンの色変化をユーザへ提示することで押す感覚を疑似的に与えることができた。

また、ドラッグ動作等と区別するためにユーザのクリック動作を4つの状態に分類し、その状態遷移によりクリッ

ク動作を判定することが可能となった。ユーザの実際の指によるクリック動作により閾値を推定するシステムを構築し、ユーザごとに閾値を簡単に設定することができ認識率を向上させることができた。また、簡単な文字入力インタフェースと電卓インタフェースを操作してもらい、指による直感的なクリック動作入力の有効性を示すことができた。

今後の課題は操作環境の影響を受けない手領域の識別法の確率や指によるクリック操作の識別率向上、また、クリック動作に特化するのではなく、指の状態変化により携帯端末の他の操作方法であるタップ動作やドラッグ動作なども可能なシステムへ拡張し、操作性の向上へ繋げる。

## 参考文献

- [1] GoogleGlass, Google, <http://www.google.com/glass/start/>.
- [2] Taehee L., Tobias H.: Handy AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking, *IEEE Wearable Computers, 2007 11th IEEE International Symposium on*, pp.83-90 (2007)
- [3] 橋本直, 石田明彦, 稲見昌彦, 五十嵐健夫: TouchMe: CG 重畳表示を用いたロボットの直接操作手法, *The 21st International Conference on Artificial Reality and Telexistence, Proceedings of ICAT2011* (2011)
- [4] Murase T., Moteki A., Ozawa N., Hara N., Nakai T., Fujimoto K.: Gesture Keyboard Requiring Only One Camera, *Proceedings of the 24th annual ACM symposium adjunct on User interface software and technology*, pp. 9-10 (2011).
- [5] Terajima K., Komuro T. and Ishikawa M.: Fast finger tracking system for in-air typing interface, *In Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, pp.3739-3744 (2009).
- [6] Harrison C., Benko H. and Wilson D. A.: OmniTouch: Wearable Multitouch Interaction Everywhere, *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pp.441-450 (2011).
- [7] Colombo C., Bimbo D. A. and Valli A.: Visual capture and understanding of hand pointing actions in a 3-D environment, *Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, Vol.33, No.4, pp.677-686 (2003).
- [8] Kinect, Microsoft, <http://www.microsoft.com/en-us/kinectforwindows/>.
- [9] iPad, Apple, <http://store.apple.com/us>
- [10] Kim H., Albuquerque G., Havemann S. and Fellner W. D.: Tangible 3D: Immersive 3D Modeling through Hand Gesture Interaction, *Proceedings of the 11th Eurographics conference on Virtual Environments*, pp.191-199 (2005).
- [11] A. Wilson, Robust.: Vision-Based Detection of Pinching for One and Two-Handed Input, *UIST*, (2006).
- [12] M. Lee, R. Green, and M. Billinghurst.: 3D Natural Hand Interaction for AR Applications, *Image and Vision Computing New Zealand 23rd International Conference*, pp.1-6, 2008.
- [13] H. Kamo, J. Tanaka.: AiR surface: Virtual Touch Panel Interface by Augmented Reality, *Proceedings of the 73th Annual Convention IPS Japan*, pp.“4-299”-“4-230”, (2009).
- [14] H. Kim, G. Albuquerque, S. Havemann, and D. W. Feller.: 3D Modeling with Hand Gesture Interaction in a Semi-Immersive Environment, *Technical Report TUBSCG-2004-01, Institute of Computer Graphics*, (2004).
- [15] M. Kolsch, R. Bane, T. Hollerer, and M. Turk.: Touching the Visualized Invisible: Wearable AR with a Multimodal Interface. *IEEE Computer Graphics and Applications*, May/June (2006).
- [16] K.Hoshino and M.Tomida.: 3D hand pose estimation using a single camera for unspecified users, *Journal of Robotics and Mechatronics*, Vol.21, No.6, pp.749-757, (2009)
- [17] Leap Motion, Leap Motion, Inc, <https://www.leapmotion.com/>.
- [18] A Novel Accelerometer-Based Gesture Recognition System
- [19] R. Y. Wang, and J. Popovic.: Real-Time Hand-Tracking with a Color Glove, *Journal of ACM Transaction on Graphics*, Vol.28, No.3, 2009.
- [20] Chaudhary, A., Raheja, J. L., Das, K., & Raheja, S. (2011). Intelligent Approaches to interact with Machines using Hand Gesture Recognition in Natural way: A Survey. *International Journal of Computer Science & Engineering Survey*, 2(1), 122--133. doi:10.5121/ijcses.(2011).