

複数大画面に対する絶対及び相対座標指定に基づく ジェスチャ入力手法

福井 辰哉^{1,a)} 山本 豪志朗^{1,b)} 武富 貴史^{1,c)} Christian Sandor^{1,d)} 加藤 博一^{1,e)}

概要: 本稿では周囲を取り囲むように複数の壁一面に配置されたディスプレイに対して、そこに表示されたコンテンツを素早く、正確に操作するジェスチャ入力手法を提案する。近年、ディスプレイ技術の発達によって、複数の大型ディスプレイを安価に配置できるようになり、より多くの情報を同時に提供できるようになった。その一方でディスプレイ領域を物理的に拡張し、室内空間を取り囲む様にディスプレイを配置した環境下では、従来のポインティング手法はユーザの視界よりもディスプレイ領域が広いため、カーソルが見つからないなどの問題が生じて操作性が著しく低下する。本稿ではそのような問題を解決するため、絶対及び相対座標指定に基づくポインティングジェスチャ入力によって実現する手法を提案する。そして素早く正確な座標指定インターフェースの実現を目指し、設計に関わる課題点について言及する。また試作システムの動作についても報告する。

Absolute and Relative Gesture Pointing on Large Wall-sized Displays

TATSUYA FUKUI^{1,a)} GOSHIRO YAMAMOTO^{1,b)} TAKAFUMI TAKETOMI^{1,c)} CHRISTIAN SANDOR^{1,d)}
HIROKAZU KATO^{1,e)}

Abstract: Nowadays, physically large displays are being installed in many places thanks to the advancements in display technology and reduced costs. Although traditional interaction devices such as mouse, joystick, keyboard, etc. are proven to be useful when there is only one display, large wall-sized displays demand a more intuitive way of controlling the cursor. In order to solve this problem we propose a new gesture pointing interface which allows rapid and accurate two dimensional pointing on large wall-sized displays. The proposed system combines the absolute and relative positioning methods with the capabilities of gesture interfaces. In this paper we describe the design and implementation of our system.

1. はじめに

近年映像表示装置の低価格化や高性能化によって様々な場所でデジタルディスプレイ（以下、ディスプレイ）を配置することが可能になり、部屋の壁一面にディスプレイを配置することも容易になった（図 1 参照）。このように囲むようにディスプレイが配置された環境（以下、複数大画面環境）は、多人数でのミーティング、ディスカッションの

場において有効に活用されることが期待されている。しかし複数大画面環境にてデジタルコンテンツを操作するためにマウスやタッチパネルといったポインティングデバイスを従来と同様に使用すると、ディスプレイ領域が大きいことにより操作に時間がかかる、カーソルが消失するなどの問題を引き起こすことが知られている（図 2 参照）。これは目的のポインティング位置が現在のカーソル位置から遠いほどカーソルの移動に時間がかかったり、ディスプレイ領域が拡大することによってカーソルがユーザの視界から外れることが多くなり、カーソルの現在位置発見が遅れがちになるためである。つまりディスプレイサイズ、配置方法はカーソルの操作性に大きな影響を与えることになる。従来ディスプレイ上をポインティングする手法は絶対座

¹ 奈良先端科学技術大学院大学
Nara Institute of Science and Technology

a) fukui.tatsuya.fl3@is.naist.jp

b) goshiro@is.naist.jp

c) takafumi-t@is.naist.jp

d) sandor@is.naist.jp

e) kato@is.naist.jp

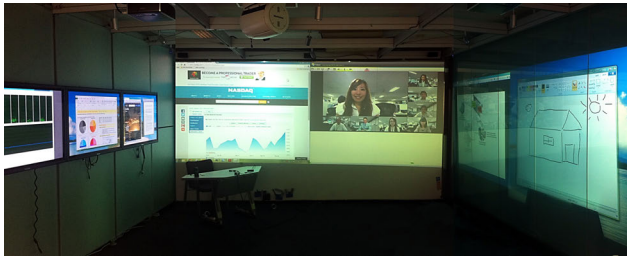


図 1 壁一面に配置されたディスプレイ環境
Fig. 1 Large Wall-sized Displays

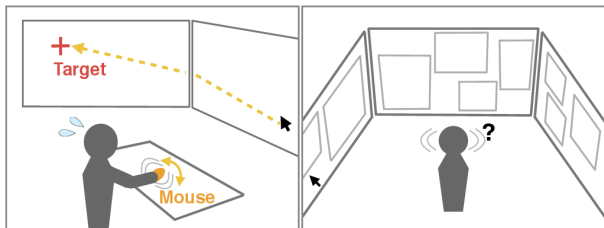


図 2 ディスプレイ領域の拡大による問題点。(左)カーソルの移動量の増加(右)カーソルの消失
Fig. 2 Problems of large display environment. left: Increase of an amount of cursor movement, right: Lost cursor

標指定方式と相対座標指定方式に大別され、大画面を操作するための状況下で、それぞれ異なる利点欠点を有する。

相対座標指定方式は、マウスなどに用いられており、主に現在位置からの移動量を用いる。複数大画面環境であっても、カーソルの現在位置を把握していれば、グラフィカルユーザインターフェースに対して十分な精度を有している。現在のカーソル位置がわからない場合は操作に対して座標を指定する操作性が低下する。一方で絶対座標指定方式はタッチディスプレイやレーザーポインタなどが例としてあげられ、表示領域が空間的に一対一対応であることが特徴である。これらは相対座標指定のように操作開始時のカーソル位置に依存せず、瞬時に目的の領域をポインティングできる。ただしタッチディスプレイはディスプレイの前にいないと操作できず、またレーザーポインタのようなレイキャスティング方式では、前提に離れた位置から操作することが想定される複数大画面環境において、カーソルを高精度に制御しにくい。

本研究では複数大画面環境下での従来のポインティング方式が有する上記の問題点を解決すべく、絶対座標指定と相対座標指定の入力手法を組み合わせ、素早く正確に操作できるインタフェースの実現を目指す。特に会議室にて利用されることを前提に、その室内であればどこからでも操作でき、また利用者を限定することのない環境を想定する。そのため特別な機器の持ち込みを利用者に要求しないハンドジェスチャに注目し、本稿では素早く正確な操作が可能なインタフェースの設計について考察する。

2. 関連研究

本研究では大きく二つの特徴をもつ。一つは、絶対座標指定と相対座標指定を組み合わせるポインティング手法で、もう一つがハンドジェスチャを用いたポインティング手法である。以下、各々について関連研究を記し、本研究の位置づけを行う。

2.1 絶対・相対座標指定の複合ポインティング

大画面環境をより素早く正確に操作するために、絶対座標指定と相対座標指定を組み合わせたインタフェースが数多く提案されている。McCallum ら [2] や Debarba ら [3] はタッチ機能を有した携帯端末を大型のディスプレイ上のカーソル操作に用いることをそれぞれ提案しており、携帯端末の画面を従来のタッチパッドの様に使えるだけでなく、McCallum ら [2] は携帯端末の画面をタップする、Debarba ら [3] は携帯端末をレーザーポインタの様にディスプレイに向けることで、大型ディスプレイ上の一対一に対応付けられた位置にカーソルを瞬時に移動させることができ、大型ディスプレイ環境においても素早く正確な操作を可能にしている。Clifton ら [9] はペン型のデバイスを使って、複数の操作法を切り替えながらカーソルを操作するシステムを提案している。手の届く範囲を操作する場合は直接ペンでポインティングし、手の届かない範囲を選択する場合はペンをタッチパッドの様に操作することで、カーソルを相対移動量で動かす事ができるため、複数大画面環境においてもディスプレイの前を歩きまわることなく、全てのディスプレイ領域を操作することを実現している。

ただし、これらの手法は全て入力のためのデバイスを必要とするインタフェースであり、ユーザがデバイスを保持していない時には用いることができないという欠点もある。特に複数大画面環境においては複数人で表示された情報を同時に閲覧するという使用用途も考えられるため、デバイスを用意するためのコスト、接続する為の手間も大きくなり不便が生じる。

2.2 ハンドジェスチャインタフェース

デバイスを必要としないインタフェースとして良く挙げられるものにハンドジェスチャを用いたインタフェースがある。Jeffrey ら [1] は指差しなどのハンドジェスチャを用いることで手の届かない位置に表示された仮想オブジェクトともインタラクションすることができるインタフェースを提案している。ユーザ視点で仮想オブジェクトを指さずジェスチャをすることで、ユーザはそのオブジェクトにタッチする感覚で操作でき、直感的なインタフェースを実現している。この指差しジェスチャを用いた操作はレーザーポインタを用いたポインティングに酷似しているため、遠

くからでも素早く目的の位置をポインティングできるが、複数大画面環境においては十分な精度を確保できない。

Chi-Chiang ら [6] は深度カメラである Microsoft 社の Kinect センサを用いて、同様の指差しジェスチャを用いた絶対座標指定を提案しており、Kinect を用いる事でユーザは何もデバイスを身につける事なくディスプレイを操作することができる。また顔が腕により Kinect から隠れてしまった場合でも遮蔽領域周辺のオプティカルフローを元に位置推定を行うことでより安定で高精度なポインティングを目指している。このようなハンドジェスチャを複数大画面環境下で用いる事で、デバイスを必要とせず、マウスカーソルが見つからないという複数大画面環境における問題も克服できるが、絶対座標指定方式の根本的な問題である、ディスプレイからの距離によって精度が落ちてしまうという問題は未だ解決されていない。

3. 絶対及び相対座標指定に基づくジェスチャ入力手法

複数大画面環境において、素早く正確なポインティングを可能とするインタフェースを実現することを目指し、本稿では絶対座標指定及び相対座標指定を組み合わせたジェスチャ入力手法を提案する。ここではそのジェスチャインタラクションの設計について述べる。

本研究では、ポインティングタスクをカーソル移動、イベントトリガの二つに大別する。それぞれ、マウス操作に例えるとマウスカーソルを目的の位置に移動し、マウスボタンのクリックに相当する。さらに、カーソル移動を絶対座標指定 (AP: Absolute positioning), 相対座標指定 (RP: Relative positioning) の 2 種類の操作を適宜組み合わせることで実現する。また、イベントトリガはオン (T_{on} : Event trigger on) とオフ (T_{off} : Event trigger off) とする。この際、カーソル移動を行うジェスチャとイベントトリガを扱うジェスチャを同一の手で行うと、イベントトリガのオン・オフの際、カーソル位置がずれてしまう可能性があり、この 2 種のジェスチャは別々の手に割り当てられるべきである。図 3 に各ジェスチャ入力の概観を示す。

3.1 指差し動作によるカーソル移動

図 3 (AP) のように絶対座標指定を行い、カーソル位置を直接的に与える。ユーザはディスプレイ上の指定したい領域を一定時間指差すことで、現在のカーソル位置に影響を受けず、素早く目的の位置までカーソルを動かすことができる。既述のとおり、高精度なカーソル移動は困難であるが、複数画面があったとしても空間的な幾何関係を把握していれば直感的にカーソルを操作することができる。

3.2 机上面での手の動作によるカーソル移動

相対座標指定における、大きな問題として現在のカーソ

ル位置を見失うことが挙げられたが、指差し動作によってカーソル移動を行った後は、容易に利用可能であり、高精度なカーソル操作も十分に期待できる。ここでは、図 3 (RP) のように机上面で手をマウスのように滑らせて動かし、その移動ベクトル等をカーソルの移動ベクトルへと変換することでカーソル位置を制御する。本稿では、カーソルの移動ベクトルの指定方法として、移動量を扱う 3 種 (距離に基づく方式, 時間に基づく方式, 距離・時間に基づく方式), 方向を扱う 2 種 (手中心座標系, 机中心座標系) に分類し、方式毎にジェスチャを以下のように設計する。

3.2.1 距離に基づく方式

この方式は主にマウスやトラックパッドにて使われている移動量指定方式であり、操作開始位置には関係なく、そこからの移動量をカーソル移動量に反映する。そのため、本システムでは机上面に手が触れた場所を始点とし、そこから動いた距離を利用する。

3.2.2 時間に基づく方式

この方式は、レバーコントローラに代表される指定方法であり、レバーの傾き方向に応じてカーソルが一定速度で動き続ける。本システムでは手が机上面に置かれたとき、その場所を始点と設定し、そこから手が一定距離以上離れると、対応する方向にカーソルを一定速度で動かし続ける。

3.2.3 距離・時間に基づく方式

上記 2 種を組み合わせた方式でありジョイスティックなどで用いられる。ある方向に一定速度で動くのではなく、その速度を調整できるようにする。つまり、手を机上面においた位置を始点とし、そこからの距離を速度量に割り当てることで、速度可変の時間に基づく移動量の決定を行う。

3.3 相対座標指定による移動方向のマッピング

上記の移動量の指定方法について述べたが、複数の画面がある場合、手の移動方向とディスプレイ上のカーソルの移動方向との対応関係は操作性に関して重要である。複数大画面環境では、ユーザとディスプレイが正対しているとは限らず、手の移動方向を扱う座標系と、ディスプレイ上での移動方向を扱う座標系を違和感なく対応付ける必要がある。そのため、現存するインタフェースを参考に、以下二つの対応関係を設計する。

3.3.1 手中心座標系

マウスを把持した状態でカーソル操作するときの挙動を再現する対応関係であり、現在の手の位置を基準にディスプレイに対応する座標系を設定する。そのため、机上面の手の前左右方向とディスプレイ上の上下左右方向が常に一致するよう座標関係が対応付けられ、図 4 のように、操作者に対して手を右に動かすとカーソルも右に移動する。

3.3.2 机中心座標系

ユーザやユーザの手の向きに関係なく、ディスプレイ中心での幾何的対応関係を用いた設定である。つまり、ディ

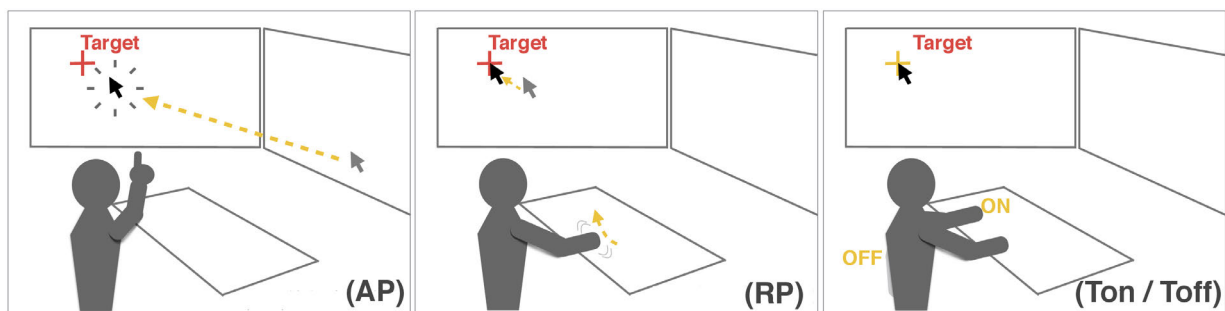


図 3 本提案ジェスチャ入力によるポインティングタスクの各要素：(左)指差し動作による絶対座標指定 (AP),(中央)机上面での手の動作による相対座標指定 (RP),(右)もう片方の手を机上面に置くか否かによるイベントトリガのオン (T_{on})・オフ (T_{off})

Fig. 3 Elements of a pointing task on our gesture method: (Left) Absolute positioning by pointing with one finger (AP), (center) relative positioning by moving a hand on a table (RP), (right) event trigger by putting the another hand on a table (T_{on}/T_{off}).

スプレイに対する垂直方向で、近づく動作が上方向、離れる操作がした方向、それに応じて左右方向が定まる。図 4 に示す通り、ユーザの左側のディスプレイ内のカーソルを動かす際、テーブル上を右に動かすと、カーソルはディスプレイの上方向へと移動する。

3.4 イベントトリガ

目的の位置へカーソルを移動させたのち、イベントトリガのオン・オフを行う(図 3 (T_{on}/T_{off}) 参照)。イベントトリガはマウスの左クリックに相当し、カーソル移動に利用した手と別の手が机の上に置かれた際に左クリックボタンが押された、もしくは離れた際に左クリックボタンが離れたとする。

4. 試作システム

試作システムの実装とその動作について述べる。

4.1 システム構成

本システムは、部屋の一面に設置された情報表示用複数大画面ディスプレイ、ユーザをセンシングするための距離画像センサ (Microsoft 社 Kinect for Windows v2 Sensor)、コンピュータ (Intel(R) Core(TM) i7-3960K 6 コア 3.20GHz, 16GB) で構成される。全体の処理を通して CPU を用いた演算で 30fps で動作することを確認した。本システムの外観を図 5 に示す。

4.2 各ディスプレイと実空間の幾何キャリブレーション

ディスプレイに対する指差し位置や机の上に置かれた手の位置を推定するために、各ディスプレイと机、実空間の幾何的性を求めておく必要がある。つまり、Kinect の深度カメラ (以下、深度カメラ) 座標系 Σ_{Depth} から各ディスプレイ座標系 Σ_{d_i} への変換行列 T_{d_i} と深度カメラ座標系

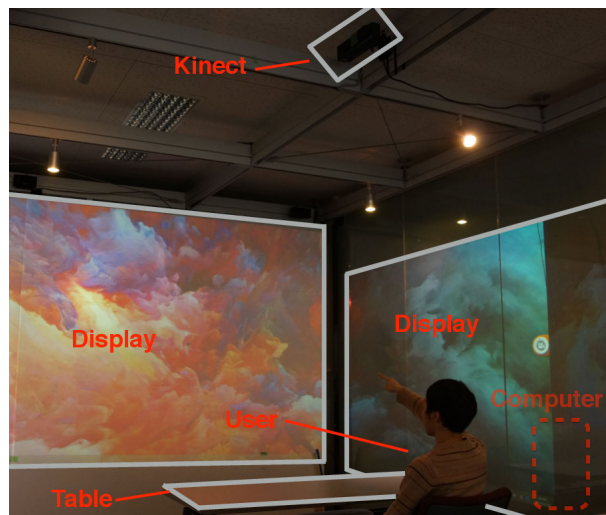


図 5 本システムの外観

Fig. 5 System Configuration

Σ_{Depth} における各機の平面パラメータ A_i を事前に求めておく。

4.2.1 ディスプレイ座標系への変換行列の算出

まず深度カメラ座標系 Σ_{Depth} とあるディスプレイ座標系 Σ_{d_n} の変換行列 T_{d_n} を求める。深度カメラ座標系 Σ_{Depth} の三次元点 x をディスプレイ座標系の二次元点 p へと変換する行列は、カメラの内部パラメータ ϕ 、カメラの位置姿勢 $[R|t]$ によって $p = \phi(Rx + t)$ のように表され、ARToolKit[11] などを用いて求めることができる。しかし本システム的设计上、深度カメラからディスプレイは映らないため、図 6 の様にマーカを介して T_{d_i} を求める。

4.2.2 机平面パラメータの算出

手の机との接触判定に、深度カメラ座標系における机平面的パラメータ $A = (a, b, c)$ を用いる。深度カメラ座標系における机の平面方程式を $aX + bY + cZ = 1$ とし、深度カメラから実際に計測された平面上の N の点 (X_i, Y_i, Z_i)

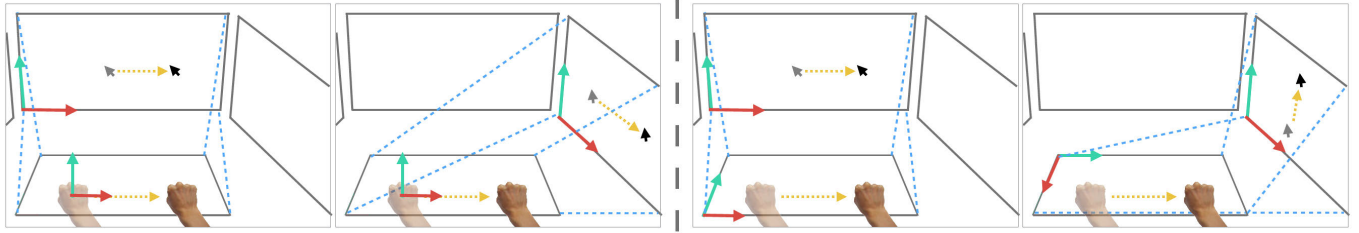


図 4 各座標系上でのカーソルの動き (左) 手中心座標系 (右) 机中心座標系
Fig. 4 Cursor movements on each coordinate

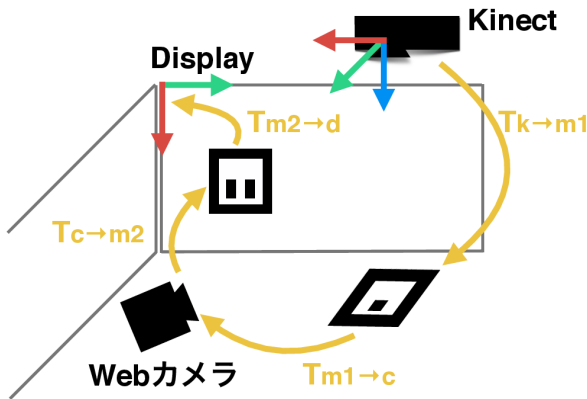


図 6 ディスプレイ座標系のキャリブレーション
Fig. 6 Calibration for displays

とおき、擬似逆行列を用いて連立方程式を解くことで A を求める。

4.3 各ジェスチャ認識の実装

図 7 は実際に本手法によりディスプレイ操作を行っている様子である。各々の操作法について、どのように実装したかについて述べる。

4.3.1 ユーザ領域の抽出

指差しや机上手を動かす動作をユーザごとに検出するため、まずユーザ領域の抽出を行う。今回、ユーザ同士のオクルージョンを避けるため、Kinect を天井から見下ろすように設置したため、提供されている Kinect SDK を用いてのユーザ認識はできない。そのため、机平面より Kinect に近い領域 (机より上の領域) にある物体をユーザ領域の候補とし、ユーザ領域とそうでない領域分けた二値画像を作成、その画像に対して領域分割、ノイズ領域の除去を行うことで、それぞれのユーザの領域を切り分けた (図 8)。その後、Kinect から得たデプス画像中でのそれぞれのユーザ領域に対して処理を行うことにより、ユーザの頭、手の三次元位置、胴体の領域、手の領域を計測した。リガのオン・オフをポインティングと同じ手で行うと、手が動き、ポインティング位置がずれてしまう恐れがあるため、ポインティングを行った手とは逆の手によってトリガのオン・オフを切り替える。各々の操作法について、どの

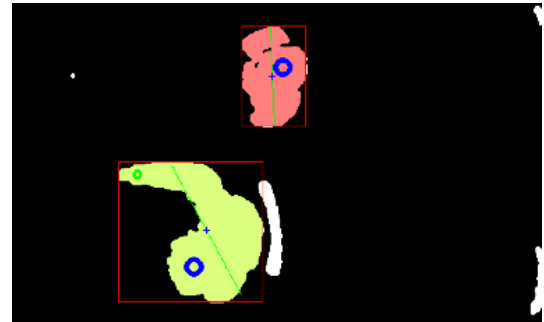


図 8 ユーザ領域の抽出結果
Fig. 8 Result of user area extraction

ように実装したかについて述べる。

4.4 絶対座標指定の実装

指差されたディスプレイ上の点はユーザの頭と右手の成す直線とディスプレイの交点として計算した。そのため、まず深度カメラ座標系におけるユーザの頭、手の三次元位置 x_k を求め、2点を $T_{k \rightarrow m2} = T_{c \rightarrow m2} T_{m1 \rightarrow c} T_{k \rightarrow m1}$ によってディスプレイ平面上のマーカ座標系上の点 x_{m2} に変換した。その後、2点が xy 平面と交わる点を求め、 $T_{m2 \rightarrow d}$ によってディスプレイのピクセル座標へと変換することで、指差された点を求めた。

4.5 相対座標指定の実装

相対座標指定の際、まず机上手に右手があるかを判別するために、式 1 によって机平面 $aX + bY + cZ = 1$ と右手の座標 (x_0, y_0, z_0) との距離 d を求め、しきい値以下であった時、右手が机の上にあると認識する。またその時前フレームでの右手と機の接触状態を確認し、右手が机に接触し始めた (前フレーム: 非接触, 現フレーム: 接触) 時点での点を相対座標系の原点とおいた。

$$d = \frac{|ax_0 + by_0 + cz_0 + 1|}{\sqrt{a^2 + b^2 + c^2}} \quad (1)$$

4.6 クリックの実現

右手と同様に式 1 によって机平面と左手の座標との距離を求め、その距離がしきい値以下の場合、トリガ操作が行われたと認識する。またその時前フレームでの左手と機の接触状態を確認し、左手が机に接触し始めたとき (前フ

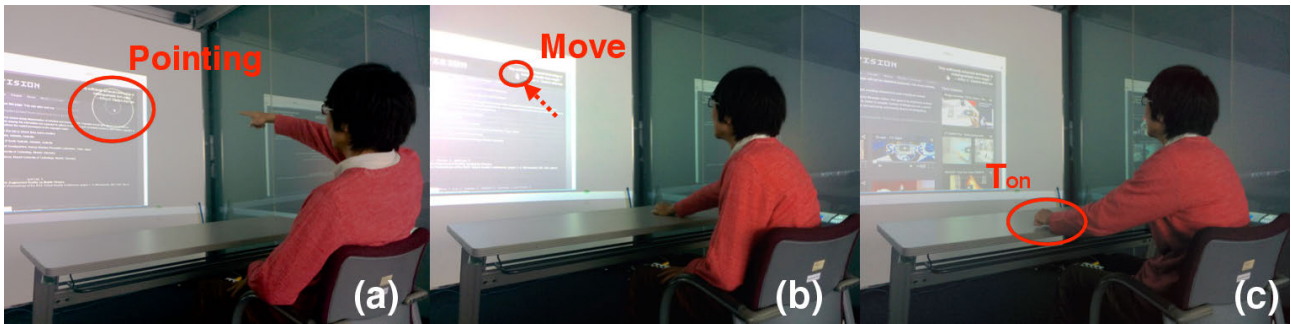


図 7 試作システムを操作する様子。(a) 右手の指差しによる大まかな位置指定 (b) 右手を机上で動かすことで細かい位置調整 (c) 左手を机上に移動させてトリガオン

Fig. 7 Process flow. (a) Rough pointing by a right hand (b) Accurate pointing by moving a right hand on the table (c) T_{on} by putting left hand on the table

レーム：非接触，現フレーム：接触）にトリガオン，左手が机から離れた時（前フレーム：接触，現フレーム：非接触）にトリガオフと認識した。

5. まとめと今後の展望

本稿では室内の壁一面に複数の大型ディスプレイが配置された環境において生じるカーソル消失問題を解決するためのジェスチャポインティングインタフェースを設計し，その試作システムの実装について述べた．今後，従来用いられているマウスとの操作性比較のための被験者実験を行い，複数大画面環境において，カーソル位置がわかりにくい状況でも素早く正確なディスプレイ操作を行うことができるのか，また，正対していない複数のディスプレイを操作するときの手の移動とカーソルの移動の対応関係についても同時に調査する．

参考文献

- [1] Jeffrey S. Pierce, Andrew Forsberg, Matthew J. Conway, Seung Hong, Robert Zeleznik, and Mark R. Mine. Image Plane Interaction Techniques in 3D Immersive Environments. In Proc. of I3D 1997, pp.39-43, 1997.
- [2] David C. McCallum and Pourang Irani. ARC-Pad: Absolute+Relative Cursor Positioning for Large Displays with a Mobile Touchscreen. In Proc. of UIST 2009, pp.153-156, 2009.
- [3] Henrique Debarba, Luciana Porcher Nedel and Anderson Maciel, LOP-Cursor: Fast and Precise Interaction with Tiled Displays using One Hand and Levels of Precision. In Proc. of 3DUI 2012, pp.125-132, 2012.
- [4] Mathieu Nancel, Emmanuel Pietriga and Michel Beaudouin-Lafon. Precision Pointing for Ultra-High-Resolution Wall Displays. Research Report of Institut National de Recherche en Informatique et Automatique, pp.1-8, 2011.
- [5] Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga and Xing-Dong Yang. High-Precision Pointing on Large Wall Displays using Small Handheld Devices. In Proc. of CHI 2013, pp.831-840, 2013.
- [6] Chi-Chiang Huang, Rong-Hao Liang, Liwei Chan and Bing-Yu Chen. Dart-It: Interacting with a Remote Display by Throwing Your Finger Touch. Poster in Proc. of

SIGGRAPH 2014, 2014

- [7] Se'bastien Carbini, Jean Emmanuel Viallet and Olivier Bernier. Pointing Gesture Visual Recognition for Large Display. In Proc. of ICCVG 2004, pp.203-208, 2004.
- [8] Yan Zhai, Guoying Zhao, Toni Alatalo, Janne Heikkila, Timo Ojala and Xinyuan Huang. Gesture Interaction for Wall-Sized Touchscreen Display. In Proc. of UbiComp 2013, pp.175-178, 2013.
- [9] Clifton Forlines, Daniel Vogel and Ravin Balakrishnan. HybridPointing: Fluid Switching Between Absolute and Relative Pointing with a Direct Input Device. In Proc. of UIST 2006, pp.211-220, 2006.
- [10] Brad Johanson, Armando Fox and Terry Winograd. The Stanford Interactive Workspaces Project, Designing User Friendly Augmented Work Environments, Springer, pp.31-61, 2010.
- [11] ARToolKit: <http://www.hitl.washington.edu/artoolkit/>