

J-033

キーボード上の指の動きによるマウス機能

Mouse emulation by using finger movements on a keyboard

及川 隼人
Hayato Oikawa

高橋 正信
Masanobu Takahashi

1. まえがき

現在 PC を利用する際、主にマウスが利用されている。しかし、マウスを利用するには、キーボードとマウス間で手を頻繁に移動させる必要があり、また、マウスを移動させるスペースが必要となる問題がある。

キーボードから手を離さず操作が可能なポインティングデバイスとしては、タッチパッド、ポインティング・スティック、トラックボールなどがあるが、マウスに比べて操作性が劣るため、マウスを利用できる環境ではほとんど利用されない。

他のポインティングデバイスとして、ユーザーの体に機器を装着して使用するタイプのものである。荻野らは人間の筋収縮レベルに応じて発生する EMG 信号を利用したポインティングデバイスを報告している[1]。しかし、このシステムを利用するにはマルチテレメータという特殊な機器を必要とし、また電極を両腕に身につけなければならない。

また、指に小型の光学式のマウスを装着して使用するフィンガーマウスが市販されている[2]。これは凹凸のある面上では使用できず、利用するにはキーボードから平らな面が存在するスペースへ手を移動させる必要がある。

我々は、マウスの機能をキーボード上の手の動きで実現することでこれらの問題を解決できると考えた。カメラで撮影した手の動きからマウス機能を実現する研究としては、FingerMouse [3]が報告されている。これは、プレゼンテーションなどを投影したスクリーン上で手のジェスチャーや手の位置を認識することで、プレゼンテーションをサポートするものである。また、ディスプレイの前を2台のカメラを用いて真横と真上から撮影し、そのディスプレイの前で指差しを行うことで指差しを行った先のディスプレイ上をポインティングできるシステムについての報告もある[4]。しかし、何れのシステムもキーボードを用いた文字入力と併用することは考慮されておらず、マウス機能と文字入力を同時に切り替えるのは困難である。

本研究では、カメラで撮影した手の形状から文字入力モードとポインティングモードを認識する機能、ポインティングモードにおいて指先の位置でポインティングできる機能、手の形状でクリックを入力できる機能を実現した。これにより、キーボード以外のスペースを必要とすることなく、キーボードから手を離さずに右手のみで操作できるマウス機能を実現した。

2. 処理内容

2.1 機能の概要

図1に実現した機能の概要を示す。図1左の手の形状にすることで通常の文字入力を行うモードから“ポインティングモード”に切り替わる。このモードでは指先位置を移

動させることでディスプレイ上のポインタの移動を行うことができ、親指を図1右のように動かすことでクリックを入力できる。

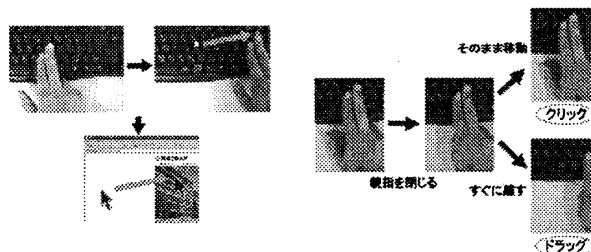


図1 マウス機能 (左: ポインタの移動 右: クリック)

2.2 基準点の自動検出—射影変換パラメータの推定

キーのうち「Q, P, Z, M」を基準点とし、後述する方法で自動検出する。基準点を元にキーボード面上にキーボード座標 (u, v) と指先検出領域 (右手が存在する領域) を自動設定する (図2)。一方、撮影画像上では指先検出領域は射影変換され歪んでいるため (図3)、画像座標 (x, y) を逆変換 (射影変換) し、キーボード座標 (u, v) を取得する。射影変換の8つのパラメータは、キーボード座標の4つの基準点に対応する4点の画像座標を自動検出して推定する。

基準点の座標の検出法: 閾値処理により撮影画像から文字領域を抽出する。閾値は領域面積が既知の範囲内に収まるように局所的に自動最適化することで、照明変動にロバストな処理を実現した。抽出された各領域の中心点を求めた2値画像 (図4) に対してハフ変換を行い各文字列の直線を得る (図5)。各直線がどの文字列を通る直線であるのかはカメラ位置より決定できるため、文字列の直線状の文字領域を端から順にカウントすることにより基準点を検出する。

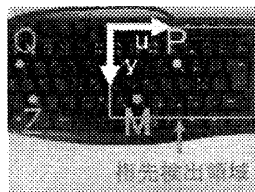


図2 キーボード座標

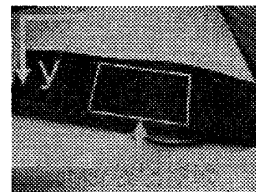


図3 撮影した画像



図4 文字領域中心点

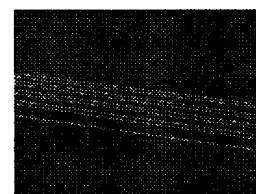


図5 文字列検出結果

2.3 指先座標の検出

指先検出領域をキーボード上部から探索し、最初に見つけた肌色画素を指先の位置として検出する。探索はキーボード座標(図6)ではなく、キーボード座標上の指先検出領域の左右の端に対応する画像座標を射影変換により求め、画像座標上で行った(図7)。これにより、キーボード座標で探索する場合に必要な画像全体の射影変換を不要とし、処理時間を短縮した。

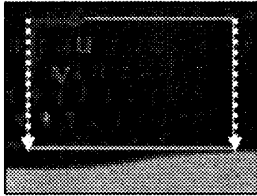


図6 キーボード座標

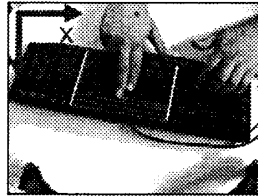


図7 画面座標上での探索

2.4 手の形状によるモードの判別

検出した指先位置を基準として、手の形状を認識する領域をキーボード座標上に自動設定する。この領域の左右の端を画像座標上に射影変換することで画像座標上の対応する領域を求める。図8のように、各線分について W_s / W_f (W_s : 肌色の左端と右端の幅, W_f : 線分の幅) を算出する。それらの平均値からポインティングモード(図8)と文字入力モード(図9)を判別する。文字入力モードと判別された場合、そのフレームでの処理を終える。ポインティングモードと判別された場合、ディスプレイ上のポインタの移動処理とクリック入力の判断処理を行う。

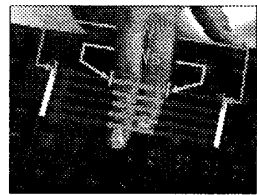


図8 ポインティングモード

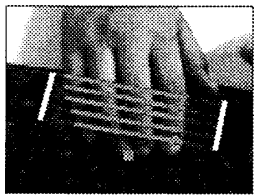


図9 文字入力モード

2.5 二本の指が成す角度を用いたクリック入力

親指と人差し指の角度を認識する領域を2.4と同様に画像座標上に自動設定する。図10に示すように、その領域内を複数の位置で右側から肌色を探索し、最初に見つけた肌色位置(画像座標)を射影変換してキーボード座標を得る。キーボード座標上で上部から各点間を順に直線で結んでいき、傾きが大きく変わる点間を境に直線を2グループに分ける(図11)。各グループの直線の傾きを親指と人差し指の角度とし、その平均値の差を2本の指が成す角度とする。その角度が閾値以上かどうかで、クリックした状態か放した状態かを認識する。

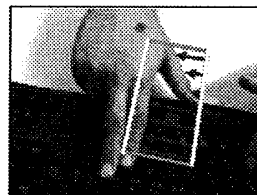


図10 角度検索領域

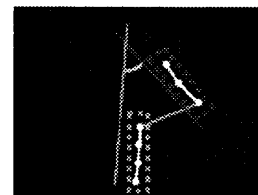


図11 角度の算出

3. 実験

3.1 指先位置検出の精度評価

図12のように指先検出領域(縦9.5cm, 横15.2cm)に格子状に評価位置を設定し、そこに指先を置いたときに検出されたキーボード座標と正解座標の誤差を評価した。平均誤差は約1.5mm(指先検出領域の横幅の0.9%に相当)であった。ディスプレイ上のポインタを動かす場合、ユーザーはディスプレイ上のポインタの位置を見ながら相対的に指を動かす。即ち、視覚によるフィードバックがあるため、この誤差はあまり問題とはならない。



図12 指先位置検出実験に用いた評価位置(黄点)

3.2 描画実験

ペイントを使い描画実験を行った。ペイントを用いることでクリック時のみ線を描画することができる。

図13の左は一般のデスクトップ用のフルキーボード上で漢字「芝」を描いた結果であり、描画時間は約4秒であった。図13の右は別のキーボードでも利用できることを確認するために、小型のキーボードを用いて「渦」を描いた結果である。描画時間は約3秒であり、マウスと同等か、より短い時間で描画できた。なお、フレームレートは37fps(CPU: Core 2 Duo E8500)であった。



図13 描画例(描画時間 左:約4秒, 右:約3秒)

4. あとがき

キーボード上の指の動きを利用したマウス機能を実現した。カーソルの位置は指先の位置によりコントロール可能であり、手の形状を判別することによりポインティングと文字入力の機能の切り替えがスムーズにできる。また、親指の開閉動作によりクリック入力を行える。

今後は、様々な手の形状を認識することにより新しい機能を追加したり、様々なタイプのキーボードへ対応できるようにしたいと考えている。

参考文献

- [1]Hiroataka Ogino, Jun Arita, and Toshio Tsuji, *Journal of Robotics and Mechatronics*, "A Wearable Pointing Device using EMG Signals", Vol. 17, No. 2, pp. 173-180 (2005)
- [2]LOGISYS Corporation Web site, "<http://www.logisyscomputer.com/viewsku.asp?SKUID=MS601BK&DID=KEYBOARD>".
- [3]Christian von Hardenberg, François Bérard, "Bare-hand human-computer interaction", *ACM Workshop on Perceptive User Interfaces*, ACM Press, pp.113-120 (2001)
- [4]Noor Shaker, M. Abou Zliekha, "Real-time Finger Tracking for Interaction", *5th International Symposium on image and Signal Processing and Analysis*, pp.141-145 (2007)