

Web カメラを用いたマルチポインティングインタフェースの構築

平間 浩二 杉田 拓也 田村 仁†

日本工業大学工学部情報工学科†

1. はじめに

ポインティングデバイスとは直感的で使用者の思い通りに操作が出来、複数点同時入力が可能である。しかし、マルチタッチパネルにも問題が存在し、下記の問題点が挙げられる。

- (1) 市販されている低解像度の商品でも価格が7万円前後と非常に高価
- (2) 画面に触れて操作する為、操作時に画面が隠れてしまうため操作が困難
- (3) 新たに画面に触れる場合、使用者の視点から触ろうとしている位置と実際に触れる位置が異なる視差の問題が発生

以上の問題点を解決する為、本研究では低価格でマルチポインティングデバイスと使用者に支障が出ない GUI の構築を提案する。ここでは「マルチポインティングインタフェース」と呼ぶことにした。関連するシステムとしては Tactiva 社の「TactaPad」[4]と Seth Sandler 氏が考案した Web カメラを使用したマルチタッチパネル[5]などがある。[4]では使用者の手を上から撮影し、撮影画像を薄くモニターに重ねて描画し、タッチパッド部分はモニターに表示されているアプリケーションの部分のみタッチできる様になっている。[5]は Web カメラを使用して手の影を撮影し、撮影した画像を処理することでマルチタッチパネルを実現している。

2. システム構築

本研究では Web カメラを使用して手の影を撮影し、撮影した画像を処理することでマルチタッチパネルを実現している。ハードウェアの実現として、1 面のみ開いている不透明なプラスチックケース内部に USB Web カメラを上向きに設置し、プラスチックケース上部に透明なプラスチック板に薄い紙を貼り付けた不透明なパネルを取り付けた(図 1)。また、プログラミング言語として C++を使用し、画像処理ライブラリである

OpenCV ライブラリを使用した。尚、Web カメラの解像度は OpenCV の現在の仕様で解像度を変更できない為、本研究では 320×240 で処理を行いフレームレートは 20[fps]とした。動作環境として Pentium 4 3.0GHz を搭載したデスクトップパソコンを使用した。使用 OS は WindowsXP である。

使用者の操作性を向上するために、Web カメラで撮影した手の影をディスプレイ上に合成表示させるオーバーレイ動作を構築した(図 1)。

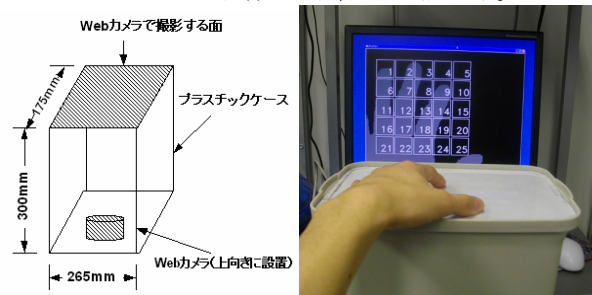


図 1 構築図と動作例

3. 実装

マルチポインティングインタフェースを構築する為には、上記条件で撮影した画像から複数の指先を抽出する必要がある。本研究では室内照明条件下を基準に下記の手順で処理を行った。

- (1) 撮影した画像の取り込み
- (2) 取り込んだ画像の左右反転
- (3) 画像の黒い部分を検出(図 2 参照)

256 段階のグレースケールで明度が 85 以下の部分を白色、明度が 86 以上に部分を黒色の二値画像で抽出する。

- (4) 白領域の収縮膨張処理

抽出した二値画像には面積の 10 画素以下の白領域(ノイズ)が発生してしまう。その為、収縮膨張処理でノイズを消去する必要がある。

- (5) 二値画像から不要な白領域の消去

抽出した二値画像は、使用者の腕や手なども抽出されている。指は映像面の大きさに対して全体の解像度の 5%以下の面積で抽出される為、5%以上の白領域は消去すれば良い。

- (6) 抽出した領域の重心座標の算出

抽出した二値画像から各白領域の重心座標を計

Implementing Multi-Pointing Device used by Web Camera

†Koji HIRAMA, Takuya SUGITA, Hitoshi TAMURA

Department of Computer and Information Engineering, Faculty of Engineering, Nippon Institute of Technology

算する。

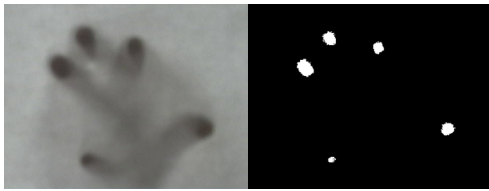


図2 接触部分の検出

また、ポインティングデバイスとして必要不可欠なのが、クリック、ドラッグ、移動といった動作であり、処理は下記の通りである。

(1) クリック動作

指が触れている状態から離し、もう一度触れる動作をクリックして認識する。その為には、予め 20 フレーム前までの指先の重心座標を記録しておき、撮影画面上で現在の重心座標から半径 15[px]以内に 1 フレーム前に重心座標が存在しておらず、2 フレームから 20 フレーム前に重心座標が存在していたら、クリックする座標として扱う。

(2) ドラッグ動作

1 フレーム前に撮影した指先の重心座標と現在の重心座標を比較し、一番近い座標を同じポイントとして認識する。2 点間の座標距離をドラッグする距離として扱う(図 3)。

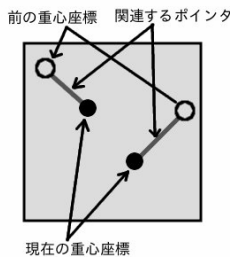


図3 ドラッグするポイントの関連図

(3) 移動動作

移動使用者が操作する際に手の位置が把握できれば、操作性が向上すると考えられる為、Webカメラで撮影した手の影をディスプレイ上に合成表示させる。

4. 評価実験

操作性を評価する手法として、メニューを操作する仮想アプリケーションを構築した。本実験で構築した仮想アプリケーションの仕様は下記の通りである。

- (1) メニューは画面全体を使って表示
- (2) 1 点の座標でドラッグした場合にはメニューを移動
- (3) 2 点以上の座標でドラッグした場合にはメニューの縮小・拡大を行う
- (4) クリックした場合には、該当するメニューの

ボタンを点灯

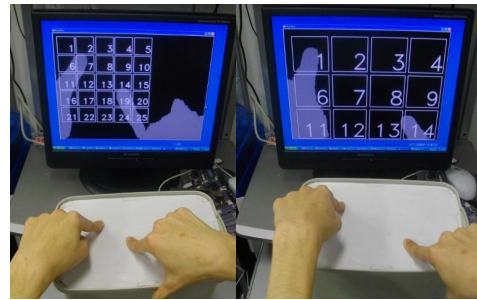


図4 メニューの縮小拡大

実験には、マルチポインティングデバイスの使用経験のない学生に操作して貰い、使用者が思い通りに動かせるか評価実験を行った。実験を行った結果は下記の通りである。

(1) 誤認識

ポインティングデバイスに触れる際に、使用者の手のかざす角度によっては誤認識した。

(2) 操作性

本研究の実験で、使用者が押しやすいと感じたボタンサイズは撮影画像上で 50[px]以上 90[px]以下であった。

実験結果より、画面上の 1/5 の面積を占めることとなり一般的なアイコンやメニューを操作するには解像度が不足することが明らかである。今後このインターフェースに対応する新たなメニューが必要である。

5. おわりに

マルチポインティングインターフェースを構築子評価実験を行い、マルチタッチパネルの欠点である価格・視差・操作性を検証した。価格を約 10%まで抑えることができ、視差を GUI により手をオーバーレイ表示させることで被験者の 80%がマルチタッチパネルより視差を抑えることが出来た。操作性においては誤認識が多く見られ改善の余地がある。

参考文献

[1] 満田 成紀, 鯉坂 恒夫, 沢田 篤史「情報相分割に基づくユーザインターフェース設計」
電子情報通信学会論文誌 D Vol. J85-D1 No. 11 pp. 1047-1056
[2] 佐藤 洋一郎, 横平 徳美, 籠谷 裕人, 岡本 卓爾, 茅野功「描画時合成方式と表示時合成方式の併用によるスムーズ操作が可能なマルチウィンドウシステム」
電子情報通信学会論文誌 D Vol. J86-D1 No. 9 pp. 650-660
[3] 綱手 雅彦, 佐藤 基次, 森本 大資, 藤川 浩一「上肢機能障害者の文字入力のためのマウス軌跡平滑化機能を有するソフトウェアキーボード」
電子情報通信学会論文誌 D Vol. J90-D No. 3 pp. 763-770
[4] Tactiva, TactaPad, <http://tactiva.com/index.html>
[5] Seth Sandler, Mtmini, <http://ssandler.wordpress.com/>