

Hough 変換を用いた手形状認識とそれを用いたユーザインタフェース

6U-8

— InteractiveDESK その2 —

待井 君吉、荒井 俊史、葛貫 壮四郎  
(株)日立製作所 日立研究所

1. はじめに

机全体を計算機との対話環境としたシステム InteractiveDESK を開発した [1, 2]。このシステムでは、正面の直立した画面に加えて机上も画面になっており、机上でも電子情報を扱うことができる。

ところで、机上で作業する場合、机上の現実物を手で扱うことが多い。InteractiveDESK でも机上の現実物を手で扱うのと同様に、机上に表示された電子情報を手で扱うことができれば、より自然で直感的なユーザインタフェースを構築することができると考え、手を使った操作に着目した。

手を使った操作の実現手法としては、データグローブなどを使うことも考えられる [3]。しかし、計算機を使うときだけ異物を手につけるといっては使い勝手は悪くなる。そこで、素手で操作可能な環境の構築を考え、机上を監視するカメラで撮影された画像から手を切り出し、その形状を認識する手法について検討した。

2. 手形状認識の課題

手形状認識の課題を次のように設定した。

(1) 応答性

現実世界で物を持って移動させる場合には、手の動きと物の動きが一致する。それと同様に、表示物の動きが手の動きに追従する程度の処理速度が必要である。従来技術として、手を細線化して指を切り出す手法 [4] があるが、処理時間がかかる。

そこで、第1の課題として、処理速度が速く、応答性のよいことが求められる。

(2) 安定性

手は紙のような薄い平面ではなく、曲面に囲まれた立体である。それをビデオカメラで撮影した場合、光の当たり具合や影などの影響によって、完全な手の形が得られるとは限らない。また、ユーザが指に絆創膏や指輪などの異物をつけている場合は、その部分が欠けてしまうことも考えられ、誤認識が起りやすい。

そこで、第2の課題として、誤認識が起きず、安定した認識結果が得られる手法が必要である。

3. 手形状認識手法

上記の課題を解決する手法として、手形状認識に Hough 変換 [5] を応用することを考えた。

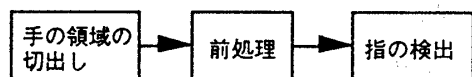


図1 手形状認識処理の流れ

3.1 処理の流れ

処理の流れを図1に示す。まず、カメラで撮影された画像から手の領域を切り出す。次に、前処理として、その形状が握り手か開き手かを判定し、どちらかであれば処理を打ち切る。最後に、手の領域から指を検出する。

これらの処理について次に述べる。

(i) 手の領域の切り出し

手の切り出しには、(1) 手がとりうる色の範囲、(2) 手の最小面積 (画素数)、(3) 手の領域の最大外接矩形の大きさ、(4) 手の最大出現個数、の4種類の特徴量を用いる。明るさの影響を小さくするために、手の色の範囲を色相、彩度、輝度それぞれの最大値と最小値で表す手法を採用した。

ビデオカメラで得た映像を順次スキャンし、ある画素の色が上記 (1) の範囲に入っていたら、その画素を中心として、上記 (3) の手の領域の最大外接矩形の4個分の大きさの領域を調べ、上記 (1) の範囲の色を持つ面積 (画素数) を求める。その面積が上記 (2) の手の最小面積よりも大きければ手が見つかったと判定する。見つかった手の個数が上記 (4) の最大出現個数になるか、画像の全領域をスキャンしたら終了する。

(ii) 前処理

(i) 開き手の判定

机上監視カメラから手までの距離は一定していると仮定すると、手の形状が同じならば切り出される手の面積も同じである。開き手の場合、あらゆる手形状の中でもっとも大きい面積が得られることから、所定値以上の面積ならば開き手とすることができる。

(ii) 握り手の判定

まず、切り出した手の領域の外接矩形をとる。次に、その外接矩形を4分割し、それぞれの矩形領域に含まれている手の画素数を数える。その画素数が手の面積の20~35%を占める領域を数え、該当する領域が3以上あった場合、握り手とみなす (図2)。

(iii) 指の検出

手の領域をHough変換し、直線を検出する。これを図3に示す。ここで検出された直線が指の候補になり、検出された直線の端点が指先の候補になる。

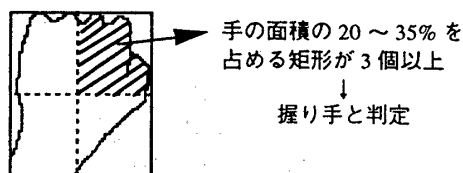


図2 握り手の判定

次に、検出された直線のうち、短い直線を除く。その後、端点どうしの距離が近いものは1本にまとめる。この直線が指になる。手がカメラ画像の下側から出てくると仮定すれば、直線の2端点のうち、指先座標は上の端点である。

### 3.2 認識結果

認識結果を図4、図5に示す。図4は1本指の認識例、図5は2本指の認識例である。

## 4. 手を使った操作の実現

手を使った動作として、1本指によるポインティングと、2本指によるシート移動を実現した。これを、図6、図7に示す。

### (1) 応答性

処理速度は、SPEC92INTが62.8、SPEC92FPが49.9のワークステーションで640×480ドットの画像に対し、秒2コマ程度の処理速度であった。これは、シート移動には不十分な処理速度であるが、ポインティング程度には実用可能という感触を得た。

### (2) 安定性

親指の認識は不安定であった。2本指によるシート移動は、親指と人差指でできるのが自然であると考えられるが、親指以外の2本指で操作することに

なった。これは、3.1(3)の処理で短い直線を除去する際に、親指となりうる直線も除去したためであると考えられる。

また、人だかりなどによって光がさえぎられる状況では、手形状認識に悪影響が出た。本手法では、明るさの影響を受けにくくするために、色相、彩度、輝度を使用する方法をとった。しかし、光がさえぎられると、あらかじめ設定した手の色の範囲では手を切り出すことができず、操作ができなくなることがあった。

## 5. おわりに

本報告で述べた手法は、完全な形で手を切り出せなくても認識が可能であり、指に異物をつけていても、認識することができる。今後は、さらなる処理速度の向上が課題である。また、他の操作への応用も考えていきたい。

## 参考文献

- [1] 葛貫, 荒井, 待井: “現実物をも取り込むことのできる直感的ユーザインタフェース”, 情処第51回全大, 6U-7, 1995.9.
- [2] 荒井, 葛貫: “実世界での操作に反応する机型の対話環境”, 日本ソフトウェア科学会 WISS '94 予稿集, pp.19-28, 1994.12.
- [3] 渡邊, 黒川: “コンピュータデスクトップへの身振り対話の導入”, 計測自動制御学会ヒューマンインタフェースシンポジウム講演論文集, Vol. 2, pp.65-72, 1993.
- [4] 岡村, 隅元: “非接触手形状認識とその応用”, 信学技報, HC93-6, pp.31-38, 1993.5.
- [5] 松山, 興水: “Hough変換とパターンマッチング”, 情報処理, Vol. 30, No. 9, pp.1035-1046, 1989.9.

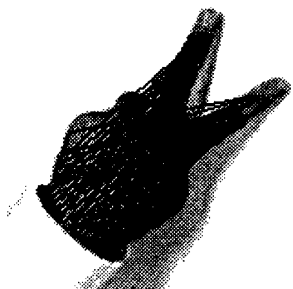


図3 Hough変換結果の例



図4 1本指の認識例

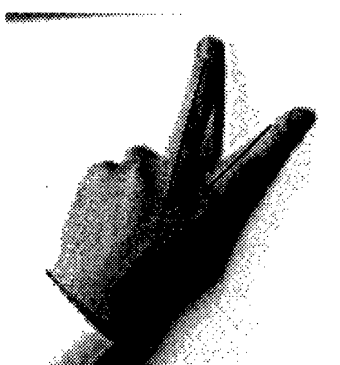


図5 2本指の認識例



図6 ポインティングの例

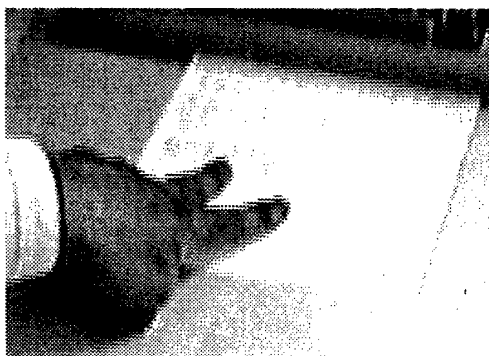


図7 シート移動の例