

スマートフォンカメラによる指先検出を用いた 適応型ハンドジェスチャ認識

清田 優子[†] 若原 徹[†]
法政大学情報科学部[†]

1. まえがき

ハンドジェスチャによる情報入力、手を用いるためユーザが自然に使用することのできるインタフェースとして研究[1]されている。

本研究では Android プログラミングに着目し、スマートフォンなどに標準搭載されている単眼カメラを用いての、適応型ハンドジェスチャ認識を提案する。5名のユーザを対象として評価実験を行った。

2. 提案手法

本研究ではハンドジェスチャ認識を、(1)肌色検出、(2)指先検出、(3)指先位置の変化によるジェスチャ識別、の3段階で行う。また、肌色検出の適応処理として、アプリケーション起動直後に学習用画像の取得とパーセプトロンの学習を行う。認識対象とするハンドジェスチャは、指の上下左右4方向の移動、拡大・縮小にあたる動作、の合計6種類とする。上下左右4方向の移動は、画面の外から向かい合う辺の外に出るまで指を移動させる動作とする。拡大にあたる動作は、グーとパーを2回繰り返す動作とする。縮小にあたる動作は、手を画面内で1秒間静止させる動作とする。

2.1. 肌色検出の適応処理

本節では、個人適応のための処理を説明する。

2.1.1. 学習用画像の取得

本研究では、学習用画像は、アプリケーションを起動する際に取得する。肌色サンプルは、画面中央に枠を表示し、手で覆われた枠内の画素を使用する。非肌色サンプルは、手が写っていない状態の画像全体の画素からランダムに選択する。肌色サンプル、非肌色サンプル共に1250画素用いた。学習結果が偏らないように肌色サンプルと非肌色サンプルを交互に学習した。

2.1.2. 3層パーセプトロンの学習

本研究で使用した3層パーセプトロンは、入力層は8つのユニットを持ち、7つの入力値 (sinH, cosH, L, S, Y, Cb, Cr) とバイアス用の値 1 を保持する。中間層のユニット数は 20 としている。出力層は肌ユニットと非肌ユニットという二つのユニットを持つ。出力層のユニットは、二つとも 0 から 1 までの値を出力する。入力画素が肌色の場合は、肌ユニットが 1 に近い値を、非肌ユニットは 0 に近い値を出力し、入力画素が非肌色の場合はその逆となる。図 1 に本研究で使用した 3 層パーセプトロンを示す。

ンを示す。

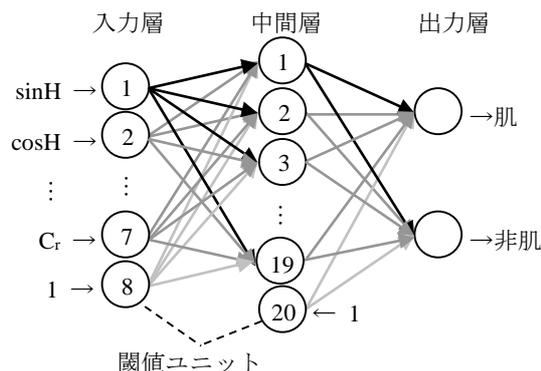


図 1 本研究で使用した 3 層パーセプトロン

活性化関数には sigmoid 関数を用いて、学習には誤差逆伝播法 (Back Propagation) を用いた。

2.2. 肌色検出

肌色検出には、2.1.で学習した 3 層パーセプトロンを用いる。入力画像は色空間変換を行い、7つの色情報 (sinH, cosH, L, S, Y, Cb, Cr) に変換された後、1画素ずつ色情報が学習済みのパーセプトロンへと渡される。パーセプトロンは入力された色情報から、二つの値 (肌色ユニット値と非肌色ユニット値) を出力する。この二つの出力値から肌色か否かを判断するが、本研究では式(1)の条件を満たすものを肌色とみなした。

$$O_{skin} \geq 0.6 \wedge O_{non-skin} \leq 0.5 \quad (1)$$

ここで O_{skin} は肌色ユニットの出力値を、 $O_{non-skin}$ は非肌色ユニットの出力値を表している。

その後、局所窓を用いてノイズ除去を行う。図 2 に肌色検出例を示す。



元画像 二値化処理画像
図 2 肌色検出例

2.3. 指先検出

本研究の条件では、文献[2]と同様にカメラ位置と手の

Adaptive Hand Gesture Recognition Using Fingertip Detection via Smartphone Camera

[†]Yuko KIYOTA, [†]Toru WAKAHARA

[†]Faculty of Computer and Information Sciences, Hosei University

位置が限定されることを利用する。指先の座標は肌色領域内で最も上に存在する点、つまり最も小さいy座標を持つ点として検出できる。この方法を利用することで高速な指先検出を実現することができる。画像の左上から右下に向かって全画素をスキャンし、初めて見つけた肌色画素の座標が、そのフレームの指先位置となる。

図3に指先の検出例を示す。



図3 指先の検出例

各フレームから検出した指先の座標は指先バッファに蓄える。指先バッファは、Circular FIFO で実装しており、最大70フレーム分の指先座標を保持する。

2.4. ジェスチャの識別

本節では、ジェスチャの識別について述べる。各ジェスチャについて、対象外、動作中、ジェスチャ成立を表すフラグを保持する。拡大にあたる動作については、対象外、拡大中、縮小中。ジェスチャ成立を表すフラグを保持する。指先の移動量と移動方向は指先バッファを遡ることで取得する。

2.4.1. 上下左右4方向の移動

上下左右4方向の移動が成立する条件は、指先位置が画面の外から向かい合う辺の外に出るまで進み続けることである。

2.4.2. 拡大にあたる動作

拡大にあたる動作が成立する条件は、指先位置が画面内で上、下、上の順に大きく移動することである。ここで指先位置の移動量の閾値は $height/4$ とする。heightは画面の高さを示す。

2.4.3. 縮小にあたる動作

縮小にあたる動作が成立する条件は、指先位置が画面内で1秒間静止し続けることである。広げた手を静止させる動作であるため、指先位置は画面の上部になると考えられる。

3. 実験

3.3. 開発環境と実験用画像データ

開発には、Android Studio 2.1.2, Android SDK, Android NDK[3]を使う。

本研究では、テストデータとして、5名のユーザによる6種類のハンドジェスチャの画像(横90×縦160)、1人あたり各50個、計1500個をAndroidスマートフォンのフロントカメラによって撮影した。

使用したスマートフォンはLG Electronics製 isai vivid LGV32, Android 6.0である。撮影条件として、端末を机の上に置いた状態で撮影を行った。秒間約15フレームの連番静止画像として取得した。

3.3. 実験結果

表1に、提案手法によるハンドジェスチャごとの認識率を掲げる。表2に、ユーザごとの認識率を示す。

なお、パーセプトロンの学習には60~250 sec かかった。1フレームあたりの処理時間は約150 msであった。

表1 各ハンドジェスチャの認識率

ハンドジェスチャ	認識率 (%)
上方向の移動	96.0
下方向の移動	99.2
左方向の移動	96.8
右方向の移動	96.0
拡大にあたる動作	94.4
縮小にあたる動作	96.0
全ジェスチャの平均	96.4

表2 各ユーザの認識率

ユーザ	A	B	C	D	E
認識率 (%)	94.7	92.3	99.7	95.3	100.0

拡大にあたる動作が他のジェスチャと比べて認識率が低かった原因として、ユーザごとにジェスチャを行う手とカメラの距離が異なり、グーとパーの間の指先位置の変化量を適切に設定できなかったことが挙げられる。また、拡大以外のジェスチャの誤認識の原因としては、肌色検出の不正確さが挙げられる。

4. むすび

本研究では、個人適応のハンドジェスチャ認識手法を提案し、6種類のジェスチャに対してスマートフォンカメラを用いてほぼ実時間で高精度認識を実現した。

本研究の課題として処理時間のさらなる削減が考えられる。提案手法ではアプリケーションを起動するごとに学習用の画素を取得してパーセプトロンの学習を行うことで個人適応を図っている。しかし、取得した学習用データによっては学習が収束するまでに時間がかかってしまうこともある。

また、肌色検出の精度向上も必要となる。提案手法では肌色領域内の最も上に存在する点として検出した指先の位置を基にジェスチャの識別を行うため、肌色検出の精度がハンドジェスチャの認識率に直結する。提案手法では画面の中央に表示した枠を手で覆うことで肌色サンプルを取得したため、指先は含まれていなかった。指先部分を肌色サンプルに加えることで指先の肌色検出の精度が向上し、指先検出の精度も向上すると考えられる。

文献

- [1] 村磯友子, 小川賀代, 小室孝, “携帯機器向け三次元ジェスチャインターフェースにおける主成分分析を用いた入力動作識別,” 映像情報メディア学会技術報告, Vol.36, No.21, pp. 9-12, Jun. 2012.
- [2] 入江英嗣, 森田光貴, 岩崎央, 千竈航平, 放地宏佳, 小木真人, 櫻原裕大, 芝星帆, 眞島一貴, 吉永努, “AirTarget: 光学シースルー方式 HMD とマーカレス画像認識による高可搬性実世界志向インターフェース,” 情報処理学会論文誌, Vol.55, No.4, pp. 1415-1427, Apr. 2014.
- [3] “Android Developers,” <https://developer.android.com/develop/index.html>