

携帯電話を用いた掌紋認証システムにおける掌位置検知方式の実装

太田 陽基[†] 渡辺 龍[†] 田中 俊昭[†]

[†] (株) KDDI 研究所 〒356-8502 埼玉県ふじみ野市大原 2-1-15

E-mail: †{haruki,ryu,toshi}@kddilabs.jp

あらまし 携帯電話を盗難・紛失した場合の第三者による不正使用などを防止するために、生体認証機能を搭載した端末も増えている。一方、生体認証の中でも、多くの掌紋認証方式が Zhang らにより提案されてきた。しかし、これらの方式はユーザに掌の位置合わせを行わせる特別な装置を用いているため、携帯電話に搭載されているカメラにはそのまま適用することができない。そこで本稿では、カメラ付携帯電話を用いた掌紋認証システムにおける掌位置検知方式の実装について述べる。本掌位置検知方式について評価実験を行った結果、約 96 % の高い検知率が得られ、本方式の有効性が示された。また、正しく検知されなかった掌画像を考察することにより、掌画像の撮影時における注意点が得られた。

キーワード 掌位置検知方式、掌紋認証、携帯電話、カメラ

Implementation of Palm Position Alignment Method in Palmprint Verification Using Cellular Phone

Haruki OTA[†], Ryu WATANABE[†], and Toshiaki TANAKA[†]

[†] KDDI R&D Laboratories, Inc. 2-1-15 Ohara, Fujimino-shi, Saitama, 356-8502 Japan

E-mail: †{haruki,ryu,toshi}@kddilabs.jp

Abstract When a cellular phone is lost or stolen, it may be unjustly used by a malicious user. There are many cellular phones with biometrics functions in order to prevent such an unjust use. On the other hand, many palmprint verification schemes were proposed by Zhang et al. However, particular machines in which a user can align his/her palm are used in these schemes. Therefore, these schemes cannot be applied to camera-equipped cellular phones. In this paper, we describe the implementation of palm position alignment method in the palmprint verification using the camera-equipped cellular phone. As a result of the evaluation experiment about this scheme, we obtained about 96 % position alignment rate.

Key words palm position alignment method, palmprint verification, cellular phone, equipped camera

1. ま え が き

1.1 背 景

本人認証技術とは、認証する必要のある個人に対し、事前に行った登録情報を用いて本人であることを確認する技術である。本人認証技術としては、①本人が持つ知識による認証、②本人の所有物による認証、③本人の身体的特徴などによる認証の 3 種類に分けられる [1]。そのうち、③の認証が本稿において扱う掌紋などの各人固有の生体情報を用いた生体認証である。生体認証には、指紋や顔、虹彩、掌紋などの身体的特徴を利用する方法と声や手書き、キーストロークなどの行動的特徴を利用する方法の 2 つに大別される。

近年の急速な普及にともない、多くの人が携帯電話を所有し、

名前のおり常に持ち歩くようになった。携帯電話の高性能化・高機能化により、多くの個人情報が保管され、オンラインショッピングやオンラインバンキングなどの決済が行われることも一般的になった。そのため、携帯電話を盗難・紛失した場合に備えて、悪意のある第三者による不正使用や個人情報の漏洩を防止するために、さまざまなセキュリティ機能が搭載されている。そのセキュリティ機能の中でも、①の知識による認証や②の所有物による認証に比べてなりすましされにくいという点において、上記の生体認証が注目され、指紋認証、顔認証、声認証、手書き認証などの機能を搭載した端末も増えてきている。

一方、生体認証の中でも、多くの掌紋認証方式が Zhang らにより提案されてきた [2]。掌紋認証は、顔認証、声認証、手書き認証より精度が高く、指紋認証と同程度の精度を有すること

が一般的に知られている。

著者らは、Zhang らの既存方式をもとに、掌紋画像の取得に携帯電話に搭載されているカメラを用いた掌紋認証プロトタイプシステムを試作した [3]。しかしながら、Zhang らの方式や著者らのシステムでは、ユーザが自分で掌の位置合わせを行う必要があった。そこで本稿では、掌紋画像の取得に携帯電話に搭載されているカメラを用いた掌紋認証を想定し、掌の位置を正しく検知することのできる掌位置検知方式の実装について述べる。

1.2 関連研究

Zhang らの方式とは異なる方式として、Ito らは画像の位相情報により照合を行う位相限定相関法 (Phase-Only Correlation: POC) を用いた掌紋認証方式を提案した [4], [5]。POC 関数は、類似の画像間では鋭いピークを示すが、異なる画像間では全くピークが現れないという特徴を有するため、このピークの高さが画像間の類似度を調べる指標となる。また、POC 関数は画像間の位置のずれや明るさの変化、加算ノイズに対してロバストであるため、生体認証において非常に有効である。ただし、本稿では、掌位置検知機能のない Zhang らの既存方式に対し、掌の位置を正しく検知することを目的としている。

1.3 構成

本稿は以下のように構成されている。2. 節では、Zhang らの掌紋認証方式について説明する。3. 節では、既存方式の課題を解決するための掌位置検知方式を提案する。4. 節では、3. 節の提案方式を適用した掌紋認証方式の手順を説明し、BREW エミュレータへの実装について述べる。5. 節では、3. 節の提案方式の評価結果を示し、その結果に対する考察を行う。6. 節で、本稿のまとめを述べる。

2. 既存技術

2.1 掌紋

本小節では、本稿の掌紋認証方式において使用される掌紋について述べる。

厳密には、掌紋といわゆる「手相」は異なる。掌紋とは、掌全体に見られる一定間隔に並んだ紋様の皮膚隆起線である。それに対し、手相とは、占いなどに用いられる「生命線」「頭脳線」「感情線」などと呼ばれる掌にできるしわである。本稿の掌紋認証方式において利用しているのは主に「手相」であるが、Zhang らの既存方式 [2] が “palmprint” (掌紋) という単語を使用していることから、特に断りが無い限り、以降は掌紋と手相をすべて含めて「掌紋」と呼ぶことにする。また、「厳密な意味での掌紋」は指紋同様「万人不同」「終生不変」の性質を有するが、手相は変化するものであることに注意する。

2.2 既存方式

本小節では、Zhang らにより提案された掌紋認証方式を概説する。

Zhang らはこれまで多くの掌紋認証方式を提案しているが、ここでは画像の解像度が低い場合にも利用可能な掌紋認証方式 [6], [7] の概略を説明する。この理由は、本稿の方式が携帯電

話における使用を目的とし、低解像度の画像でも認証可能であることが必要だと考えられるためである。

Zhang らの方式の概略は以下のとおりである。

(1) CCD カメラの搭載された撮影装置を用いて、掌を開いた状態で手の全体の画像を撮影する。その際、装置には各指先を置く目安となる印があり、ユーザはその印に合わせて掌を置く。

(2) 手順 (1) で得られた画像に処理を施してバイナリ画像に変換した後、境界線追跡アルゴリズムを用いて、手の部分とそれ以外の部分の境界線を定め、人差し指と中指、薬指と小指の間の各点から、基準となる座標の中心点と軸を決定する。

(3) 手順 (2) で定めた中心点と軸から、掌紋の主線が含まれるように必要な掌紋の範囲を正方形にとる。

(4) 手順 (3) で定めた範囲の掌紋のアナログ情報に補正ガボールフィルタを適用して「掌紋コード」と呼ばれるビット行列に変換する。

(5) 手順 (4) で得られた掌紋コードと、同方式であらかじめ登録されていた掌紋コードとの正規化ハミング距離を計算し、認証するか否かを決定する。

上記の手順 (1) は、撮影された手の画像から掌の位置を検知する方式である。しかしながら、上記の方式は、ユーザに掌の位置合わせを行わせる特別な装置を用いているため、掌の位置や方向、大きさは常にほぼ同一になる。一方、携帯電話などに搭載されたカメラで掌を撮影する場合、掌の位置や方向、大きさは必ずしも同じにならない。そのため、上記の方式では、掌紋認証に使用する同一の掌紋画像を得ることができず、本人同士の掌紋コードを高精度で照合することができない。そこで、携帯電話などに搭載されたカメラで撮影する場合でも、掌の位置や方向、大きさがほぼ同一になり、正しく検知することが可能な方式が必要となる。

3. 掌位置検知方式

本節では、前節において述べた課題を解決するための掌位置検知方式を提案する。

提案する掌位置検知方式では、まず掌方向を認識し、次に掌の上端を決め、その掌の上端の情報から掌の左右端を定め、最後にこれらの位置から掌の下端を決定する。提案方式の具体的な手順は以下のとおりである。

(1) 1次微分画像の作成

撮影された 320×240 ピクセルの画像を x 方向、 y 方向でそれぞれ微分し、 x 方向微分値と y 方向微分値を成分とする 1 次微分ベクトル (濃度勾配ベクトル) (図 1 参照) に関する結果画像を保存する。微分は微分前テーブルとフィルタの積和演算により行われる。フィルタサイズは 7 とし、フィルタの値は、 x 方向微分の場合には最左列を -1 、最右列を 1 、残りを 0 とし、 y 方向微分の場合には最上行を -1 、最下行を 1 、残りを 0 とする。

(2) 掌方向の算出

撮影された画像の x 成分 $3 \sim 236$ 及び y 成分 $3 \sim 316$ の領域にあ

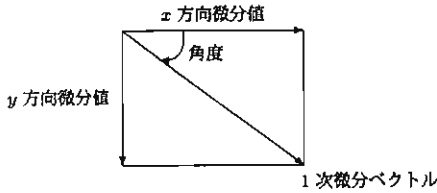


図1 1次微分ベクトル

Fig.1 Gray-level differential vector.

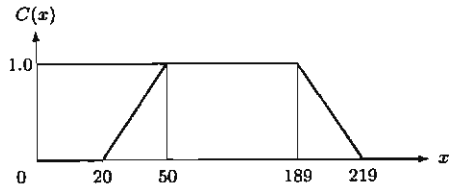


図2 係数 $C(x)$ の変化

Fig.2 Transition of a coefficient $C(x)$.

る全画素に対して、1次微分ベクトル長をその角度ごとに累積する。ただし、その角度 θ の逆方向の角度 ($0^\circ \leq \theta \pm 180^\circ \leq 359^\circ$ の成立する角度) についても累積する。得られた累積テーブルを平滑化する。平滑化は、平滑前の累積テーブルとフィルタの積和演算により行われる。フィルタサイズは 31 とし、フィルタの値はすべて 1 とする。平滑化された累積テーブルの最大値を $135^\circ \sim 225^\circ$ の範囲で探索し、その角度を掌方向に垂直な方向とする。次の手順 (3) において画像を回転する際には、上記の角度から 180° 引いた値を回転角度とする。

(3) 画像の回転

撮影された画像を手順 (2) にて得られた回転角度 φ だけ回転させ、回転画像を作成する。回転前画像の座標を (x, y) とすると、回転後画像の座標 (x', y') は

$$\begin{cases} x' = x \cos \varphi + y \sin \varphi + x_0(1 - \cos \varphi) - y_0 \sin \varphi \\ y' = -x \sin \varphi + y \cos \varphi + x_0 \sin \varphi + y_0(1 - \cos \varphi) \end{cases}$$

にて表される。ただし、 (x_0, y_0) は回転中心である画像中央の座標とする。この回転処理のため、回転画像は指が画像の上方向を向くことになる。回転画像を処理範囲内で作成されるように、画像の拡大率を 0.85 に調節する。後処理のために、回転前後の座標を保存しておく。

(4) 1次微分画像の作成 (回転画像対応)

手順 (3) にて得られた回転画像に対し、手順 (1) と同じ処理を行う。

(5) 1次微分ベクトル成分画像マスク

手順 (4) にて得られた回転1次微分ベクトル成分画像に対して、そのベクトル角度 ψ を以下を条件としてマスクを行い、マスク画像を作成する。

$$-45^\circ \leq \psi \leq 45^\circ, 135^\circ \leq \psi \leq 225^\circ$$

を満たさない場合には x 方向微分値を、

$$45^\circ \leq \psi \leq 135^\circ, 225^\circ \leq \psi \leq 315^\circ$$

を満たさない場合には y 方向微分値をそれぞれ 0 クリアする。この処理は、画像において斜め方向の1次微分ベクトルをマスクする効果がある。

(6) 上端の検知

撮影された画像領域にある、手順 (4) にて得られた回転1次微分ベクトル x 成分画像の全画素に対して、その濃度値 $g(x)$ を $\sum_x C(x)|g(x)|$ により y 座標ごとに累積する。ただし、係数 $C(x)$ は対象画素の x 座標に依存し、図2にて表される。得ら

れた累積テーブルを平滑化する。ただし、フィルタサイズは 21 とする。平滑化された累積テーブル $AT[y]$ において、

$$AT[y-1] - AT[y]$$

が最大となる位置、つまり最大減少位置を $10 \sim 192$ の範囲で探索し、その位置を掌の上端 tp とする。

(7) 左右端の検知

左右端の検知には、2とおりの手法を用いる。まず、手法1により検知を行い、失敗した場合のみ手法2により検知を行う。

(a) 手法1

手順 (6) にて得られた掌の上端 tp に対して、 $(tp-30) \sim (tp+10)$ を上方領域、 $(tp+20) \sim (tp+60)$ を下方領域とし、各領域における x 座標ごとの x 方向正及び負の微分値の累積グラフを作成し、極大点を求める。また、同じく掌の上端 tp に対して、 $(tp-20) \sim (tp+20)$ を中央領域とし、その領域における x 座標ごとの y 方向微分値の絶対値の累積グラフを作成し、極大点を求める (図3参照)。ここで、意味のある極大点を得るために、平滑化処理及び極大点の削除処理を行う。平滑化処理のフィルタサイズは 11 とする。次の2つのいずれかの条件を満たす場合に、極大点の削除を行う (図4参照)。

○ 極小値と極大値の差分 A と、最大値 B の間に、

$$A < B \times 0.2$$

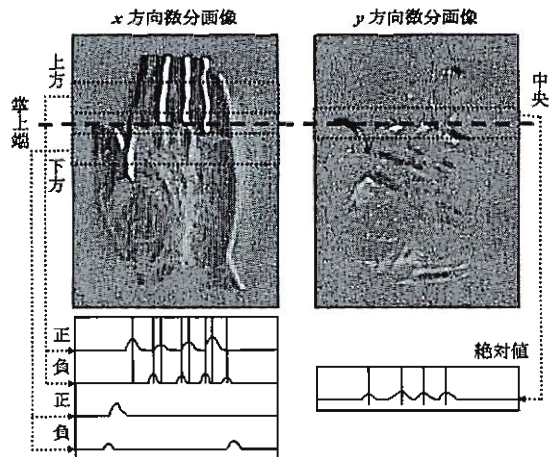


図3 手法1における累積グラフ

Fig.3 Accumulation graphs in the method 1.

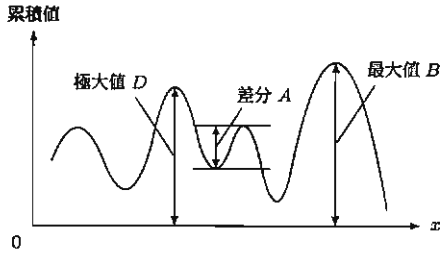


図 4 極大点の削除処理
Fig. 4 Delete process of maximal points.

の関係がある場合、この極大点を削除する。

- 極大値 D と最大値 B の間に、

$$D < B \times 0.3$$

の関係がある場合、この極大点を削除する。

掌上端上方における x 方向正の微分値の累積グラフと掌上端上方における x 方向負の微分値の累積グラフ及び掌上端中央における y 方向微分値の絶対値の累積グラフを使用して、極大点と付け根の対応づけを行い、その整合性を判定する。整合性ありと判定された場合のみ、次の左端探索処理及び右端探索処理に移る。極大点と付け根の対応づけは以下のとおりに行う。ただし、「付け根 0」を人差し指と中指の間にある付け根、「付け根 1」を中指と薬指の間にある付け根、「付け根 2」を薬指と小指の間にある付け根とする。また、 $PT[i]$ 、 $AV_x[i]$ 、 $AV_y[i]$ をそれぞれ、 i の位置、 i における x 方向微分値の累積値、 i における y 方向微分値の累積値を表すことにする。

(i) 正極大点 P に最近の負極大点 M を探し、負極大点 M に最近の正極大点 P' を探す。そして、両方の正極大点 P と P' の位置が一致するかどうかを確認する。 $PT[P] \neq PT[P']$ ならば、失敗として終了する。 $PT[P] = PT[P']$ ならば、次の手順 (ii) に進む。

- (ii) 手順 (i) の各極大点について、

$$PT[P] < PT[M]$$

ならば、

$$\begin{cases} PT[lm] = PT[P - 1], PT[r_{l0}] = PT[P] \\ PT[r_{l1}] = PT[P + 1], PT[r_{l2}] = PT[P + 2] \\ PT[r_{r0}] = PT[M], PT[r_{r1}] = PT[M + 1] \\ flg = 1 \end{cases}$$

と仮の対応づけを行う。ただし、 lm を左側極大点、 r_{l0} 、 r_{l1} 、 r_{l2} をそれぞれ付け根 0、付け根 1、付け根 2 の各左端、 r_{r0} 、 r_{r1} をそれぞれ付け根 0、付け根 1 の各右端、 flg をフラグとする。また、

$$PT[P] > PT[M]$$

ならば、

$$\begin{cases} PT[lm] = PT[M - 1], PT[r_{l0}] = PT[M] \\ PT[r_{l1}] = PT[M + 1], PT[r_{l2}] = PT[M + 2] \\ PT[r_{r0}] = PT[P], PT[r_{r1}] = PT[P + 1] \\ flg = 0 \end{cases}$$

と仮の対応づけを行う。

(iii) 手順 (ii) の各点について、以下の 4 つの整合性判定を行う。

- 付け根位置の間隔について

$$\begin{cases} PT[r_{l2}] - PT[r_{l1}] < 0.8(PT[r_{l1}] - PT[r_{l0}]) \\ PT[r_{l2}] - PT[r_{l1}] > 1.2(PT[r_{l1}] - PT[r_{l0}]) \\ PT[r_{r1}] - PT[r_{r0}] < 0.8(PT[r_{l1}] - PT[r_{l0}]) \\ PT[r_{r1}] - PT[r_{r0}] > 1.2(PT[r_{l1}] - PT[r_{l0}]) \end{cases}$$

のいずれかを満たす場合、整合性なしとする。

- 付け根の累積値について

$$\begin{cases} AV_x[r_{l1}] < 0.5AV_x[r_{l0}], AV_x[r_{l1}] > 1.5AV_x[r_{l0}] \\ AV_x[r_{l2}] < 0.5AV_x[r_{l0}], AV_x[r_{l2}] > 1.5AV_x[r_{l0}] \\ AV_x[r_{r0}] < 0.5AV_x[r_{l0}], AV_x[r_{r0}] > 1.5AV_x[r_{l0}] \end{cases}$$

のいずれかを満たす場合、整合性なしとする。

- 左側極大点について

$$\begin{cases} 0.6(PT[r_{l1}] - PT[r_{l0}]) \leq PT[r_{l0}] - PT[lm] \\ \leq 1.4(PT[r_{l1}] - PT[r_{l0}]) \\ 0.5AV_x[r_{l0}] \leq AV_x[lm] \leq 1.5AV_x[r_{l0}] \end{cases}$$

のすべてを満たす場合、整合性なしとする。人差し指の左側（つまり親指の上側）に指のような特徴が発生している場合、ここで整合性なしとなる。

- y 方向微分累積値について

$$\begin{cases} AV_y[r_{-1}] > 0.7AV_y[r_0], AV_y[r_{-1}] > 0.7AV_y[r_1] \\ AV_y[r_0] < 0.5AV_y[r_1] \end{cases}$$

のいずれかを満たす場合、整合性なしとする。ただし、 r_{-1} 、 r_0 、 r_1 をそれぞれ付け根 -1 、付け根 0 、付け根 1 とし、付け根 -1 とは、付け根 -1 と付け根 1 の中点が付け根 0 となるような位置とする。

すべての整合性判定に合格したときのみ、以下の左端及び右端の探索を行う。

- 左端探索 (図 5 参照)

まず、極大点と付け根の対応づけの手順 (ii) の flg の値により、次のとおり使用する累積グラフを選択する。

$$\begin{cases} \text{掌上端下方における } x \text{ 方向負の微分値の累積グラフ} & \text{if } flg = 1 \\ \text{掌上端下方における } x \text{ 方向正の微分値の累積グラフ} & \text{if } flg = 0 \end{cases}$$

ここで、付け根幅 rw_i を

$$rw_i = \frac{(PT[r_{l2}] - PT[r_{l0}]) + (PT[r_{r1}] - PT[r_{r0}])}{3}$$

とし、付け根累積値 ra_i を

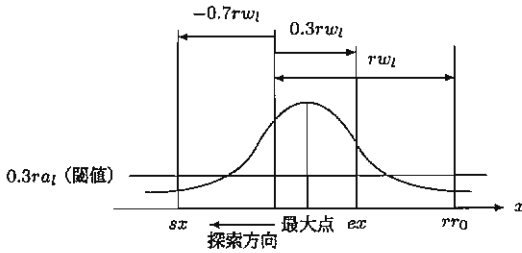


図5 手法1における左端探索
Fig.5 Left end search in the method 1.

$$ra_l = \frac{AV_x[rr_0] + AV_x[rr_1]}{2}$$

とする。ただし、使用される累積値は

$$\begin{cases} \text{掌上端上方における } x \text{ 方向負の微分値の累積グラフ} & \text{if } flg = 1 \\ \text{掌上端上方における } x \text{ 方向正の微分値の累積グラフ} & \text{if } flg = 0 \end{cases}$$

における累積値とする。まず、 $sx \sim ex$ 間の最大点を探す。ここで、閾値未満の場合、左端 lft の位置を

$$PT[lft] = PT[rr_0] - rw_l$$

とする。閾値以上の場合は、負方向に $0.6max$ 未満になるまで探索し、その直前を左端 lft の位置とする。ただし、 max は最大値を表している。

○ 右端探索 (図6参照)

まず、極大点と付け根の対応づけの手順 (ii) の flg の値により、次のとおり使用する累積グラフを選択する。

$$\begin{cases} \text{掌上端下方における } x \text{ 方向正の微分値の累積グラフ} & \text{if } flg = 1 \\ \text{掌上端下方における } x \text{ 方向負の微分値の累積グラフ} & \text{if } flg = 0 \end{cases}$$

ここで、付け根幅 rw_r を左端探索時の付け根幅 rw_l と等しくし、付け根累積値 ra_r を

$$ra_r = \frac{AV_x[r_l_0] + AV_x[r_l_1] + AV_x[r_l_2]}{3}$$

とする。ただし、使用される累積値は

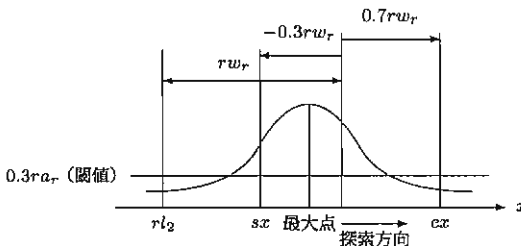


図6 手法1における右端探索
Fig.6 Right end search in the method 1.

$$\begin{cases} \text{掌上端上方における } x \text{ 方向正の微分値の累積グラフ} & \text{if } flg = 1 \\ \text{掌上端上方における } x \text{ 方向負の微分値の累積グラフ} & \text{if } flg = 0 \end{cases}$$

における累積値とする。まず、 $sx \sim ex$ 間の最大点を探す。ここで、閾値未満の場合、右端 rgt の位置を

$$PT[rgt] = PT[r_l_2] + rw_r$$

とする。閾値以上の場合は、正方向に $0.6max$ 未満になるまで探索し、その直前を右端 rgt の位置とする。

(b) 手法2

以下では、左端、右端を独立に処理する。

○ 左端探索

(i) $(0, tp + 5) \sim (99, tp + 40)$ の領域にある、手順 (4) にて得られた回転1次微分ベクトル x 成分画像の全画素に対して、濃度値の絶対値を x 座標ごとに累積する。

(ii) 手順 (i) にて得られた累積テーブルを平滑化する。ただし、フィルタサイズは11とする。

(iii) 手順 (ii) にて得られた累積テーブルから、極大点、最大点 (最大累積値の極大点) を探索する。ここで、

$$\max_i AV_x[i] < 7500$$

の場合、5を左端 lft の位置として検知を終了する (手順 (iv) は行わない)。また、隣接する極大点の間隔が10.0未満の場合は、累積値の小さい方を削除する (手順 (iv) において極大点として扱わない)。

(iv) 次の条件

- 最大点より右側に位置すること。
- $AV_x[pl] > 0.75\{\max_i AV_x[i]\}$

を満たす極大点 pl を仮の左端とする。複数存在する場合は、最も右に位置するものとする。一つも存在しない場合は、最大点を仮の左端とする。次に、この仮の左端から、左方向へ累積テーブルを探索し、

$$AV_x[j] < 0.6AV_x[pl]$$

となった場合、 $(PT[j] + 1)$ を左端 lft の位置とする。探索範囲30を超えても、上記の不等式が成立しない場合は探索を終了し、その終了位置 le に対して、 $(PT[le] + 1)$ を左端 lft の位置とする。

○ 右端探索

(i) $(140, tp + 20) \sim (239, tp + 100)$ の領域にある、手順 (4) にて得られた回転1次微分ベクトル x 成分画像の全画素に対して、濃度値の絶対値を x 座標ごとに累積する。

(ii) 手順 (i) にて得られた累積テーブルを平滑化する。ただし、フィルタサイズは11とする。

(iii) 手順 (ii) にて得られた累積テーブルから、極大点、最大点 (最大累積値の極大点) を探索する。ここで、

$$\max_i AV_x[i] < 7500$$

の場合、234 を右端 *rgt* の位置として検知を終了する（手順 (iv) は行わない）。また、隣接する極大点の間隔が 10.0 未満の場合は、累積値の小さい方を削除する（手順 (iv) において極大点として扱わない）。

(iv) 次の条件

- 最大点より左側に位置すること。
- $AV_x[pr] > 0.75\{\max_i AV_x[i]\}$

を満たす極大点 *pr* を仮の右端とする。複数存在する場合は、最も左に位置するものとする。一つも存在しない場合は、最大点を仮の右端とする。次に、この仮の右端から、右方向へ累積テーブルを探索し、

$$AV_x[j] < 0.8AV_x[pr]$$

となった場合、 $(PT[j] - 1)$ を右端 *rgt* の位置とする。探索範囲 30 を超えても、上記の不等式が成立しない場合は探索を終了し、その終了位置 *re* に対して、 $(PT[re] - 1)$ を右端 *rgt* の位置とする。

(8) 下端の検知

掌の縦横比（縦長／横長）を 1.1 とし、下端 *btm* の位置を

$$PT[btm] = PT[tp] + (PT[rgt] - PT[lft]) \times 1.1$$

で計算する。

(9) 掌位置座標の計算

掌の端 *tp*, *lft*, *rgt*, *btm* は回転した画像に対する座標であるため、手順 (3) において保存された回転前後の位置情報により、本来の座標に戻し、回転前における掌の 4 隅の座標を計算する。つまり、上記の座標を手順 (2) にて得られた回転角度 φ だけ逆回転させ、回転前の座標に戻す。回転後画像の座標を (x', y') 、回転中心である画像中央の座標を (x_0, y_0) とすると、回転前画像の座標 (x, y) は

$$\begin{cases} x = x' \cos \varphi - y' \sin \varphi + x_0(1 - \cos \varphi) + y_0 \sin \varphi \\ y = x' \sin \varphi + y' \cos \varphi - x_0 \sin \varphi + y_0(1 - \cos \varphi) \end{cases}$$

にて表される。

4. 掌紋認証方式

本節では、前節において提案した掌位置検知方式を適用した掌紋認証方式及びその実装について述べる。

前節において提案した掌位置検知方式を適用した掌紋認証方式の手順は以下のとおりである。ただし、掌位置検知方式以外の手順は、Zhang らの既存方式 [6], [7] の手順を参考にす。また、ユーザの掌紋コードが、同じ方式を用いてあらかじめ登録されているとする。

(1) 携帯電話を使用する際に認証が必要な場合、携帯電話に搭載されているカメラを用いて、掌のカラー画像を手動で撮影する。

(2) 手順 (1) にて得られたカラー画像をグレースケール化する。

(3) 手順 (2) にて得られたグレースケール画像に、前節

において提案した掌位置検知方式を適用し、掌位置を検知し、掌紋範囲の画像を得る。

(4) 手順 (3) にて得られた掌紋範囲の画像をフーリエ変換した後、低域通過フィルタを適用し、フーリエ逆変換によりグレースケール画像に復元する。

(5) 手順 (4) にて得られた掌紋の範囲の復元画像に対して、ガボールフィルタを適用して、実数部分の掌紋コード Q_R と虚数部分の掌紋コード Q_I をそれぞれ得る。

(6) 手順 (5) にて得られた掌紋コード $Q = (Q_R, Q_I)$ と、同方式であらかじめ登録されていた掌紋コード $P = (P_R, P_I)$ との正規化ハミング距離

$$HD = \frac{1}{2} (HD_R + HD_I)$$

を計算し、あらかじめ設定された閾値により認証するか否かを決定する。ただし、 HD_R と HD_I は実数部分の掌紋コード P_R, Q_R と虚数部分の掌紋コード P_I, Q_I に関する正規化ハミング距離

$$HD_R = \frac{\sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \{P_R(i, j) \oplus Q_R(i, j)\}}{N_i N_j}$$

$$HD_I = \frac{\sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \{P_I(i, j) \oplus Q_I(i, j)\}}{N_i N_j}$$

をそれぞれ表し、 N_i と N_j は掌紋コードのビット行列における行数と列数をそれぞれ表している。

ここで、本掌紋認証方式の実装について簡単に述べる。BREW 3.1 のプラットフォーム [8] が搭載されているカメラ付携帯電話を想定し、BREW SDK 3.1.2 の BREW エミュレータに対し、本掌紋認証方式の手順 (2) ~ (5) に対応するモジュールを実装した。図 7 に、BREW エミュレータにおけるモジュール構成を示す。BREW エミュレータには、掌画像入力から掌紋コードの生成までに必要な 9 つの機能を BREW アプレットとして実装した。掌のカラー画像の撮影にはカメラ付携帯電話を使用し、手動で BREW エミュレータに移動する。画像サイズは 320×240 ピクセルであり、生成される掌紋コードは 36738 ビットである。

5. 評価・考察

5.1 評価

本小節では、3. 節において提案した掌位置検知方式の評価結果を示す。

提案した掌位置検知方式を評価するために、以下の手順により実験を行った。

(1) 24 名の被験者に次の条件にて左手の掌画像を合計 14 枚撮影してもらう。このとき、画像枚数の合計は 336 枚である。表 1 に、撮影した掌画像の例を示す。本実験では、ユーザが自分の掌の画像を自然に撮影する動作を想定している。そのため、カメラと掌との間の距離について 2 とおりの距離、指が画像の上を向いた方向に対して左右 30° 回転までの 7 とおりの角度を

表 1 掌画像の例
Table 1 Examples of palm images.

	左 30°	左 20°	左 10°	0°	右 10°	右 20°	右 30°
22cm							
25cm							

BREW アプレット

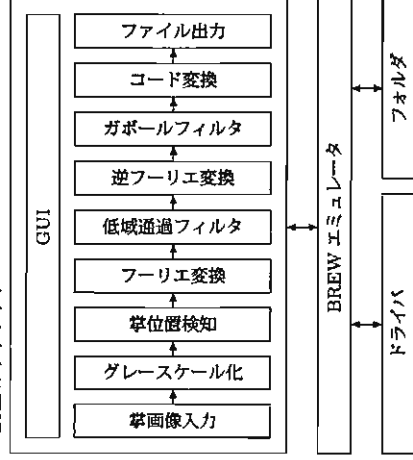


図 7 BREW エミュレータにおけるモジュール構成
Fig. 7 Module composition of BREW emulator.

対象とする。ここでは、周辺を含む掌全体が撮影されていない場合、指が画像の横方向や下方向を向いている場合などは対象外である。

- カメラと掌との距離: 22cm と 25cm
- 指が画像の上を向いた方向に対する角度:
 - 左方向に 30°, 20°, 10°
 - 0°
 - 右方向に 10°, 20°, 30°
- 枚数: 各 1 枚

(2) 手順 (1) において撮影してもらった画像に対し、3. 節において提案した掌位置検知方式を用いて、掌の 4 隅の検知座標を求め。

(3) 手順 (1) において撮影してもらった画像に対し、掌の左端 (頭指線の左端)、掌の右端 (感情線の右端)、人差し指と中指の付け根、中指と薬指の付け根、薬指と小指の付け根の 5 点から、掌の 4 隅の正解座標を計算する。

(4) 手順 (3) において得られた掌の 4 隅の正解座標に対し、手順 (2) において得られた掌の 4 隅の検知座標が一致し

ているかどうかを判定する。なお、誤差として、次の 2 つの条件をともに満足するならば、掌の 4 隅の検知座標と正解座標が一致していると判定する。

$$\begin{cases} -3.0 \leq \alpha \leq 7.0 \\ \beta \leq 30.0 \end{cases}$$

ただし、 α 、 β は各座標における上辺の角度差、各座標における左上座標間及び右上座標間の距離の和をそれぞれ表している。図 8 に、掌の 4 隅の検知座標と正解座標の例を示す。ただし、実線が検知座標、点線が正解座標をそれぞれ表している。

提案した掌位置検知方式について、以上の手順により実験を行った評価結果を表 2 に示す。386 枚の掌画像に対し、正しく検知された掌画像は 323 枚であり、約 96 % の高い検知率が得られている。

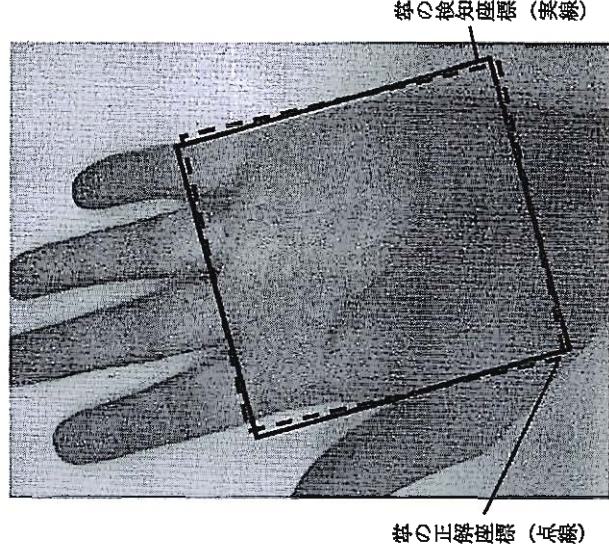


図 8 検知座標と正解座標の例

Fig. 8 An example of alignment and correct coordinates.

表 2 提案方式の評価結果

Table 2 Evaluation results of the proposed scheme.

画像数 [枚]	検知数 [枚]	検知率 [%]
336	323	96.1

5.2 考 察

本小節では、前小節における提案方式の評価結果に関する考察を行う。

図 9 に、正しく検知されなかった掌画像の例を示す。ただし、図 8 と同様、実線が掌の 4 隅の検知座標、点線が掌の 4 隅の正解座標を表している。

以下に、各画像について考察を行う。ただし、【 】内に掌画像の特徴、●の箇条書きにその際の問題点をそれぞれ挙げる。

- 左上【人差し指が中指から離れて斜めになっている】
- 人差し指が中指から離れているため、左端の位置が正解の位置より左側にずれている。
- 人差し指が斜めになっているため、掌の 4 隅の位置が正解の位置より右方向に回転している。
- 右上【小指が薬指から離れて斜めになっている】
- 小指が薬指から離れているため、右端の位置が正解の位置より右側にずれている。
- 小指が斜めになっているため、掌の 4 隅の位置が正解の位置より左方向に回転している。

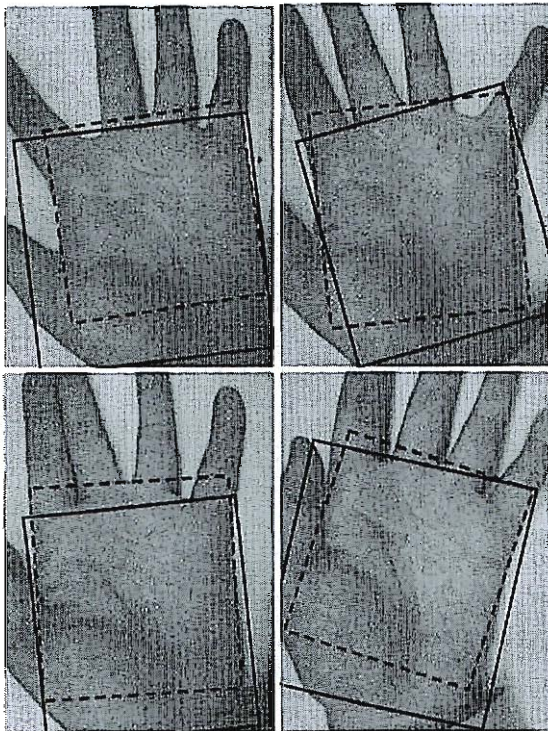


図 9 検知されなかった掌画像の例

Fig. 9 Examples of palm images that were not aligned.

- 左下【人差し指と中指がくっついている】
- 人差し指と中指がくっついているため、上端の位置が正解の位置より下側にずれている。
- 右下【親指が掌にくっついている】
- 親指が掌にくっついているため、左端の位置が正解の位置より左側にずれている。
- 親指が掌にくっついているため、掌の 4 隅の位置が正解の位置より左方向に回転している。

上記の考察から、携帯電話に搭載されているカメラで掌画像を撮影する際、以下の点に注意することが必要である。

- 人差し指と小指を他の指から離れすぎないように、かつ斜めになりすぎないようにする。
- 指と指の間はくっつけずに適度にあげる。
- 親指は掌にくっつけずに適度にあげる。

本実験では、カメラと掌との間の距離及び掌の方向を指定したため、正しく検知されないことがあったが、上記の注意点は、ユーザが適度にリラックスした状態で撮影することにより、容易に実現することができる。そのため、以上の点に注意して、携帯電話に搭載されているカメラで掌画像を撮影することにより、提案方式の検知率を上げることが可能になる。

6. あとがき

本稿では、掌紋画像の取得に携帯電話に搭載されているカメラを用いた掌紋認証を想定し、掌位置を正しく検知することのできる掌位置検知方式の実装について述べた。本掌位置検知方式について評価実験を行った結果、約 96 % の高い検知率が得られ、本方式の有効性が示された。また、正しく検知されなかった掌画像を考察することにより、掌画像の撮影時における注意点が得られた。

文 献

- [1] 瀬戸洋一, サイバーセキュリティにおける生体認証技術, 共立出版(株), 東京, 2002.
- [2] D.D. Zhang, Palmprint authentication, Kluwer Academic Publishers, Massachusetts, 2004.
- [3] 太田陽基, 沼本晋作, 田中俊昭, “カメラ付携帯電話を用いた掌紋認証プロトタイプシステムの試作,” 2007 信学ソ大(通信), B-18-3, Sept. 2007.
- [4] K. Ito, T. Aoki, H. Nakajima, K. Kobayashi, and T. Higuchi, “A palmprint recognition algorithm using phase-based image matching,” Proc. 2006 IEEE International Conference on Image Processing, pp.2669-2672, Atlanta, GA, USA, Oct. 2006.
- [5] K. Ito, T. Aoki, H. Nakajima, K. Kobayashi, and T. Higuchi, “A palmprint recognition algorithm using phase-only correlation,” IEICE Trans. Fundamentals, vol.E91-A, no.4, pp.1023-1030, Apr. 2008.
- [6] W.-K. Kong and D.D. Zhang, “Palmprint texture analysis based on low-resolution images for personal authentication,” Proc. International Conference on Pattern Recognition (ICPR 2002), IEEE Computer Society, pp.807-810, Quebec, Canada, Aug. 2002.
- [7] D.D. Zhang, W.-K. Kong, J. You, and M. Wong, “Online palmprint identification,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol.25, no.9, pp.1041-1150, Sept. 2003.
- [8] Qualcomm BREW Home, “<http://brew.qualcomm.com/brew/en/>.”