2007－AL－112 (7)
2007／5／11

安全な指紋認証のための、誤差を許容するゼロ知識証明

小島 晃司 †

† 東京大学理学部情報科学科

近年ゼロ知識証明を利用した安全な生体認証システムが提唱されているが、認証精度や利便性の面で課題が残されている。本研究では指紋認証に焦点を当てて、より良い性質をもつ安全な指紋認証を提案する。生体認証にゼロ知識証明を応用する際、誤差を含んだ形で同一性判定をするという困難が発生するが、本研究では誤差を許容する問題のゼロ知識証明を提案する事でその解決を図る。誤差を許容する問題としては、認証精度を向上させるため、マニューシャ法に特化した問題 (Δ-MA) を新たに提案、利用する。今回提案したプロトコルを計算機実験によって評価した結果、FAR (False Accept Rate),FRR (False Reject Rate) が共に 1%以下と、既存の同様の手法に比べて小さな値を得られることが分かった。

Zero-Knowledge Proofs Considering Error for Secure Fingerprint Authentication

KOJI KOJIMA †

† Dept. of Information Science, School of Science, the University of Tokyo

Recently, secure biometrics authentications using zero-knowledge are proposed, but they seems to have problems in terms of accuracy or utility. In our research, we focus on fingerprint authentication and propose the more secure fingerprint authentication protocol. When thinking about the application of zero-knowledge to biometrics authentication, the identification including errors makes problems. We solve this difficulty by proposing the zero-knowledge proofs for the problem considering error. As such a problem, we define and employ the novel problem Δ-MA, which is customized for Minutiae method, to improve the accuracy. The numerical verification of our protocol shows less than 1% of FAR (False Accept Rate) and FRR (False Reject Rate), which is less than the previous similar researches.

1 はじめに

ゼロ知識証明 [4] は GoldWasser らによって提唱された対話型証明の一種で、prover が verifier に、prover の秘密情報を漏らすことなしに証明を行うことが出来るという特長を持っている。通常証明を行う際は、相手を十分納得させるために (パスワードなどの) 秘密情報を提示することが要求されることを考えれば、この証明技術は非常に強力な能力を持っているといえ、実際にこの技術を応用したより安全な認証システムが提案されている [1]。このような流れの中、ゼロ知識証明をさらに認証技術に組み込むための研究が行われるようになり、その中のひとつとしてゼロ知識証明を生体認証技術に応用する研究がなされてきている。

生体認証は、近年になって特に注目を集めるようになってきている技術であり、通常の認証でパスワードや IC カードを記憶、所有していることによって認証を行うのに対して、個人の指紋や静脈パターン、あるいは筆跡などといった生体的な情報を保持していることで認証を行おうというものである。近年生体認証が注目を集めているひとつの理由として、生体認証特有の利便性を挙げることが出来る。パスワードや IC カードを記憶しておいたり、保持しておいたりすることは、利用者側に一種の負担を強いることになるが、自身の指紋や静脈パターンなどは、人間が通常無意識のうちに保持し続けている情報といえるため、その点で利用者側に負担を掛けなくてすむのである。しかしながら、生体認証に関しては幾つか問題点が掲げられており、その中のひとつとして情報の漏洩に

関する問題がある。通常の認証においてももちろん、情報の漏洩は大きな問題であるが、パスワードやICカードに関する情報が漏洩してしまった場合、利用者は新しいパスワードやICカードに切り替えることで比較的円滑に認証システムへ復帰することができる。しかし、指紋、静脈パターンといった生体情報は容易に変更が出来ないために一度漏洩してしまうと、利用者が再び認証システムを正常に利用していくことが非常に困難になってしまうのである。

ゼロ知識証明を生体認証へ応用するということは、この問題点を克服するためのひとつの手段であるといえる。前述のとおり、ゼロ知識証明は相手に秘密情報を漏らさずに証明を可能にするため、上手くこの技術を応用できれば理論的に相手に生体情報が漏れないことが保証された生体認証が構成できると考えられるからである。既存の研究によって、こういった能力を持った生体認証システムが幾つか提唱されているが、残念ながらそれらには若干の問題点が残っていると考えられる。

菊池らは多項式根の救解問題に関するゼロ知識証明を生体認証に利用する手法を提案している [7]が、verfier側に秘密情報を知らせなくてはならない、誤差をうまく取り扱うことが難しいなどの課題が残されていると考えられる。高橋らは、キャンセラブル生体認証 [5] にゼロ知識証明を利用する事で安全な生体認証を実現している [8]。この認証方式は安全性に関しては優れているものの、認証時に補助情報としてパスワードやICカードを利用する必要があるために、生体認証の特長の一つを失っているように見える。永井らは秘匿ニューラルネットワーク技術を利用して安全な生体認証を実現する方法を提案している [6]。これらの研究に関してはFAR=0.083、FRR=0.098との結果報告が出ており、認証精度として多少の難があると考えられる。

本研究では特に指紋認証技術に焦点を当てて、これらの既存研究とは別の手法を用いることによって安全な指紋認証システムを構成する。

2 研究の概要

本研究では、指紋認証アルゴリズムとして最も一般的と考えられるマニューシャ法をベースにして、その類似性をゼロ知識証明を利用して判定するプロトコルを提案する。マニューシャ法とは、人間の指紋の中に存在する特徴点[1] に注目し、二つの特徴点分布の類似性を調べることで同一性判定を行う方法である。

このようなプロトコルを実現するにあたって最も困難なのは特徴点分布のマッチングという、誤差を含む複雑な判定処理をどうやってゼロ知識証明するかということであり、これが通常のパスワード認証のゼロ知識証明を構築する場合と同様にプロトコルを実現できない理由といえる。本研究ではこの問題点を、特徴点分布を解に持つような、誤差を許容する問題に対するゼロ知識証明を行うことで解決を図る。ただし、ここで利用する問題は特徴点分布における類似性を保存するようなものを選択する必要がある。つまり、特徴点分布として互いに似ているものは、問題として互いに似ているものの答えとなる必要がある。そのために本研究では新たに Δ-MA という問題を提案し、この問題に対するゼロ知識証明を構成している。

このプロトコルが正常な認証を行えるかどうかは、特徴点分布から Δ-MA インスタンスへの変換が類似性をどの程度保存するのかに強く依存しており、本研究ではそれを計算機実験によって確かめた。その結果、FARとFRRが共に1%未満となり、従来の手法に比べて高い認証精度を達成している事がわかった。

3 問題の構成

この章では、本研究で提案するプロトコルで利用することになる種々の問題の定式化を行う。

はじめに以下のような問題 Δ-MA を定義する。直感的にいえばこの問題は、幾つかの制約条件を満たすように平面上に3種類の特徴点を配置する事ができるかを答える問題である。

定義 1 閾値を与える関数 $\Delta : (\mathcal{B}, \varphi, K) \mapsto d \in \mathbb{N}$ が与えられたとき、以下の問題を Δ-*Minutiae Arrangement Problem* (以下 Δ-*MA*) と呼ぶ。

- 入力
  三つ組 $(\mathcal{B}, \varphi, K)$。ただし、
  - $\mathcal{B}$ は $\mathbb{R}^2$ 上の有界集合からなる列を意味する。

[1] 特徴点とは指紋の隆線が特殊な形状を成している場所を指す言葉であり、一般的には隆線が分岐する点 (分岐点)、及び開始する点 (開始点) の事を指す。

- 関数 $\varphi : \mathcal{B} \to \mathbb{N}^3$ は、各有界集合 $B \in \mathcal{B}$ に関する制約条件 (より具体的には、B の内部に含まれる特徴点の総数を、種類ごとに纏めた後に降順に並べ替えたもの) を意味する。
- $K \in \mathbb{N}$ は、配置すべき特徴点の総数を意味する。

- 出力
  以下の条件が満たされたときに YES、そうでない時に NO を出力する。

$$^{\exists}\phi : \{1, 2, \ldots, K\} \to \{1, 2, 3\} \times \mathbb{R}^2 \quad s.t.,$$
$$^{\forall}B \in \mathcal{B}, \quad d(inners(B, \phi), \varphi(B)) \leq \Delta(\mathcal{B}, \varphi, K)$$

  関数 $inners : \mathcal{B} \times (\{1, 2, \ldots, K\} \to \{1, 2, 3\} \times \mathbb{R}^2) \to \mathbb{N}^3$ は、与えられた有界集合 B と特徴点配置関数 ϕ を受け取ったとき、ϕ によって B の内部に含まれることになる特徴点群を φ と同様の記法 (種類ごとに纏めた後、降順に並べなおした形) で返すものである。関数 $d : \mathbb{N}^3 \times \mathbb{N}^3 \to \mathbb{N}$ は特徴点情報間の「距離」[2] を定義する関数である (詳細は定義.4 を参照)。

この問題の witness となる関数 ϕ が特徴点分布に対応することは明らかであり、後で提案するプロトコルでは、与えられた特徴点分布を答えに持つような Δ-MA インスタンスを生成することで、特徴点分布の変換を行うことになる。

次に、2つの Δ-MA インスタンス同士の関係を評価する問題として、以下のような問題を定義する。

定義 2 以下の問題を *Minutiae Arrangement Subset Isomorphism Problem* (以下 *MASI*) として定義する。

- 入力
  $(\mathcal{B}_1, \varphi_1, K_1), (\mathcal{B}_2, \varphi_2, K_2) \in (\Delta\text{-})MA$
- 出力
  以下の条件が満たされたときに YES、そうでない時に NO を出力する。

$$^{\exists}(\tilde{\mathcal{B}_2}, \tilde{\varphi_2}, \tilde{K_2}) \in (\Delta\text{-})MA \quad s.t.$$
$$\begin{cases} \tilde{\mathcal{B}_2} \subseteq \mathcal{B}_2 \\ \tilde{\varphi_2} = \varphi_2|_{\tilde{\mathcal{B}_2}} \\ \tilde{K_2} = K_2 \end{cases}$$
$(\tilde{\mathcal{B}_2}, \tilde{\varphi_2}, \tilde{K_2})$ と $(\mathcal{B}_1, \varphi_1, K_1)$ は isomorphic

上記の条件が満たされているとき、以下の条件を満たすような関数 $\tilde{\pi} : \mathcal{B}_2 \to \mathcal{B}_1 \cup \{\perp\}$ を、$(\mathcal{B}_1, \varphi_1, K_1)$ と $(\mathcal{B}_2, \varphi_2, K_2)$ の間の *subset isomorphism* と呼ぶことにする。

$$\begin{cases} \tilde{\pi}(B^{(2)}) \neq \perp \Leftrightarrow B^{(2)} \in \tilde{\mathcal{B}_2} \\ \\ \pi := \tilde{\pi}|_{\tilde{\mathcal{B}_2}} \text{は } (\tilde{\mathcal{B}_2}, \tilde{\varphi_2}, \tilde{K_2}) \text{ と } (\mathcal{B}_1, \varphi_1, K_1) \\ \text{の間の isomorphism} \end{cases}$$

(「isomorphic」や「isomorphism」に関しては定義.5 を参照)

isomorphic(同型) という言葉からもわかるように、この問題は2つの Δ-MA インスタンス内の各制約条件が本質的に等価であるかどうかを評価する問題といえる。提案プロトコルの中では、秘匿性を保証するためにテンポラリインスタンスを利用することになるが、その生成の際にこの問題を利用することになる。

4 提案プロトコル

この章で、本研究で提案するプロトコルを定義する。このプロトコルは G3C(グラフ3彩色問題) に対するゼロ知識証明 [3] をベースにして構成した、Δ-MA に対するゼロ知識証明であるといえ、commitment scheme を利用している。

定義 3 有限集合 $\mathtt{N} \subseteq \mathbb{N}$ 及び $\mathtt{R} \subseteq \mathbb{R}^3$に対して、以下の様に Δ-MA に対するゼロ知識証明を定義する。

Common Input
Δ-MA インスタンス x

Auxiliary Input to the prover
x の適当な witness ϕ

[2] 対称性などを保持していないので、いわゆる距離関数ではない。

[3] N や R の例としては、`int` や `real(float, double)` などがあげられる。

Prover's first step(P1): randomization

prover は無作為に Δ-MA インスタンス x^* をひとつ、(x, x^*) が MASI インスタンスとして受理されるように生成する。この際、$|x^*|$ が適当な多項式 δ で関係付けることが出来る (つまり、$|x^*| := \delta(|x|)$ となる) 様に選ぶ。prover は x と x^* の間の subset isomorphism $\bar{\pi}$ を知っているので、これと ϕ から、新しく生成した x^* の witness ϕ^* を知ることが出来ることに注意。

Prover's second step(P2): commitment

prover は commitment scheme を利用して x^* と ϕ^* を verifier に commit する。具体的には、

$$C^{\varphi^*}(x^*) := (\mathcal{B}^*, C^{\varphi^*}(\varphi^*), K^*)$$
$$C^{\varphi^*}(\varphi^*) := \{C^{\varphi^*}_{r^{\varphi^*}_{B^*}}(\varphi^*(B^*)) \mid B^* \in \mathcal{B}^*\},$$

$$C^{\phi^*}(\phi^*) := \{C^{\phi^*}_{r^{\phi^*}_i}(\phi^*(i)) \mid i = 1, 2, \ldots, K^*\}.$$

を verifier に送信する。ここで

$$C^{\varphi^*}_{r^{\varphi^*}_{B^*}}(\varphi^*(B^*)) : \mathtt{N}^3 \to \{0,1\}^*$$

$$C^{\phi^*}_{r^{\phi^*}_i}(\phi^*(i)) : \{1,2,3\} \times \mathtt{R}^2 \to \{0,1\}^*$$

は、それぞれランダム鍵 $r^{\varphi^*}_{B^*}$ と $r^{\phi^*}_i$ を利用した commitment を意味する。

Verifier's first step(V1): question

verifier はメッセージを受け取った後以下の 3 種類の質問から一つを無作為に選び、prover に送る。

(V1-a): Subset Isomorphism
この質問は prover に (x, x^*) が本当に MASI のインスタンスとして受理されうるものかどうかを尋ねるものになる。もちろん、このプロトコルに従う prover はこの質問に簡単に答えることが出来る (P1 の際に $\bar{\pi}$ を入手しているので) が、prover が「ずる」をして自分にとって都合の良い x^* を生成しているとこの質問には答えることが出来ないことになる。

(V1-b): Minutiae Arrangement (lower)
このとき、verifier は無作為に $B^* \in \mathcal{B}^*$ を選択して、prover に条件

$$v(inners(B^*, \phi^*), \varphi^*(B^*)) \leq \Delta(\mathcal{B}^*, \varphi^*, K^*)$$

を達成させる為の特徴点集合を提示する事を要求する (v の意味については定義 4 を参照)。この質問は、直観的には prover に B^* の中にある特徴点集合で、もともとの制限 $\varphi^*(B^*)$ に比べて少なすぎない程度のものを、P2 で commit した特徴点集合から選択させるものといえる。

(V1-c): Minutiae Arrangement (upper)
この質問は、V1-b と同様に、無作為に選択した $B^* \in \mathcal{B}^*$ に対して、prover に

$$w(inners(B^*, \phi^*), \varphi^*(B^*)) \leq \Delta(\mathcal{B}^*, \varphi^*, K^*).$$

を達成する為に必要な特徴点集合を提示させるものとなる (w の意味については定義 4 を参照)。直観的には、質問 V1-b が制限 $\varphi^*(B^*)$ に比べて少なすぎないものを選ばせるのに対して、この質問は多すぎないものを選ばせるものといえる。

Prover's third step(P3): answer

ここで、porver は verifier からの質問に、その種類に応じて対処をすることになる。

(P3-a): Subset Isomorphism
このとき、prover は全制約 $\varphi^*(\mathcal{B}^*)$ を reveal (これは全ての $B^* \in \mathcal{B}^*$ に対して $\varphi^*(B^*)$ と $r^{\varphi^*}_{B^*}$ の組を渡す事で行われる) すると同時に、x と x^* の間の subset isomorphism $\bar{\pi}$ を教える事で verifier の質問に答える事になる。

(P3-b): Minutiae Arrangement (lower)
prover はこの種の質問に対しては、制限 $\varphi^*(B^*)$ を reveal し、B^* の内部にある特徴点集合 $I \subseteq \{1, 2, ..., K^*\}$ を渡して、さらにそれに対応する配置関数 $\phi^*(I)$ を reveal することで返答を行う。

(P3-c): Minutiae Arrangement (upper)
prover はこの種の質問に対しては、制限 $\varphi^*(B^*)$ を reveal し、B^* の外部 (内部でない点に注意) にある特徴点集合

$I \subseteq \{1, 2, ..., K^*\}$ を渡して、さらにそれに対応する配置関数 $\phi^*(I)$ を reveal することで返答を行う。

Verifier's second step(V2): verification
verifier は、最後に prover から送られてきた返答に対して、その妥当性を評価する (具体的な評価方法は自明なので省略する)。

この一連の処理を $|\mathcal{B}|^2$ 回繰り返し、その間に一度でも prover が不正な答えを返した場合は verifier は prover を拒否し、全て正しい答えを prover が返すことが出来たときに verifier は prover を受理する。

5 計算機による評価

この章で、今回提唱したプロトコルの妥当性を計算機実験によって評価する。今回の実験の際に利用した計算機環境は以下の通りである。

- OS : WindowsXP Professional Version 2002 Service Pack SP2
- CPU : Pentium(R) M 1.70GHz
- メモリ : 512MB RAM
- プログラム言語:
  - Java Version 1.5.0_06
  - Perl Version 5.8.7
- 利用したアプリケーション及びデータ
  - MegaMatcher algorithm demo program[4]
  - Sample fingerprint database[5] から、同一人物の指紋の画像 8 枚

今回の実験は以下のようにして行った。

1. 特徴点分布 ϕ_0 を指紋画像から抽出する。
2. 以下の処理を複数回 (今回の実験では各パラメータ毎に 10000 回ずつ) 繰り返す。
   (a) witness ϕ_0 を解とするような MA インスタンス[6] $x := (\mathcal{B}, \varphi, K)$ をランダムに生成し、適当な定数 d に対して $\Delta(\mathcal{B}, \varphi, K) := d$ と置く。
   (b) "challenging witness" ϕ を以下の 2 種の方法によって、それぞれ生成する。
      i. random generation
      単純にランダムに ϕ を生成する。これは、利用者が誤って他の利用者として認証を受けようとしてしまった場合を示唆している。
      ii. capture from the same person
      ϕ として、1 で利用したものと同一人物の別の指紋画像から抽出した特徴点配置を利用する。これは、利用者が正規の手順に則って認証を受けようとしている場合を示唆している。
   (c) ϕ が、$\Delta(\mathcal{B}, \varphi, K) := d$ とした Δ-MA インスタンス x の解になるかどうかを確かめる。
3. 認証確率を計算する。

この実験では、Δ-MA インスタンス x と各々の手法で生成した ϕ が問題とその witness の関係になっているかを確かめているだけで、プロトコルによって実際に受理されるかどうかを確かめていない。これは、提案プロトコルがゼロ知識性を有していることがわかっていれば生成した x と各 ϕ が問題とその witness の関係にあることと、ϕ を有した prover がこのプロトコルで受理されることとは、無視できるほどの小さな差を除けば同値になるからである。

本研究では、問題を難しくする都合上 3 種類の特徴点が指紋画像から抽出されることを前提にしているが、一般的なマニューシャ法で抽出する特徴点は 2 種類となる。そのため今回の実験では、特徴点のうちのひとつである分岐点を 2 種類に分類してこの問題を解決している。分類方法としては、指紋の隆線形状に対して何らかの向き付けを与え[7]、その向きに対して分岐方向が順方向か逆方向かで判断をする。これを計算機で処理すること

[4] http://www.neurotechnologija.com/download/MMDemo.zip
[5] http://www.neurotechnologija.com/download/VeriFinger_Sample_DB.zip

[6] MA とは、$\Delta(\mathcal{B}, \varphi, K) := 0$ としたときの Δ-MA 問題のこととする。
[7] たとえば渦巻状の指紋ならば、渦の中心を中心とした反時計上の向き付けを考える。

は可能と考えられるが、実装の困難さなどを加味して、今回は手動で行った。また、利用した指紋画像間の単純な位置合わせ (回転、平行移動、トリミング) も手動で行っている。

上で述べた手順の中で、特徴点分布からランダムにMAインスタンスを生成する、と記述したが、MAインスタンス全体の集合は非常に膨大であり[8]、その中には問題として非常に簡単なものも多く含まれていると考えられる[9]。そのため今回の実験ではMAインスタンスを生成方法する際、難しい問題が得られるように幾つかのheuristicsを適用した。

まず、生成する各有界集合 $B \in \mathcal{B}$ としては、適当な半径の円盤の集合を利用した。これは問題としての難しさをそれほど損ねることなく、それなりの情報量でMAインスタンスを評価するためである。尚、B を構成する円盤の総数及び半径は以下の様な範囲の中で無作為に選択を行っている。

$$
\begin{aligned}
B &:= \{C_i \mid i = 1, 2, \ldots, k\} \quad (1 \leq k \leq 4) \\
C_i &:= \{(x, y) \in \mathrm{R}^2 \mid (x - x_0)^2 + (y - y_0)^2 \leq r^2\} \\
&\quad (0 \leq x_0, y_0 \leq 100, 1 \leq r \leq 20)
\end{aligned}
$$

また、特定の B に含まれる特徴点が著しく多かったり少なかったりすると、問題の難しさが損なわれると考え、各 B に含まれる特徴点の総数に以下の様な制約を課している。

$$
\begin{aligned}
0.2K \leq max(inners(B, \phi_0)) \leq 0.6K \\
0 \leq sum(inners(B, \phi_0)) \leq 0.9K \\
max(a_1, a_2, a_3) := \max\{a_1, a_2, a_3\} \\
sum(a_1, a_2, a_3) := a_1 + a_2 + a_3
\end{aligned}
$$

この実験結果をFig.1、Fig.2及びTable.5に示す。Table.5から、パラメータ $\mathcal{B}$ を70から90、d を6から7辺りに設定すると確かにFARとFRRを1%未満にすることが出来ることがわかる。

6 実験結果考察

はじめにFig.1とFig.2から分かる特徴として、$\mathcal{B}$ を小さくすればするほど、また d を大きくすればするほど、受理率は大きくなることが分かる。こ

[8] R^2 上の有界集合としてとり得る場合の数を考えるだけでも $2^{|\mathrm{R}^2|}$ とおりあり、これを表現するには $|\mathrm{R}^2|$ ビットの記憶領域が必要になってしまう。

[9] 例えば、$\mathcal{B}$ の全ての有界集合が互いに disjoint だと、多項式時間で簡単に解の一つを得ることができる。

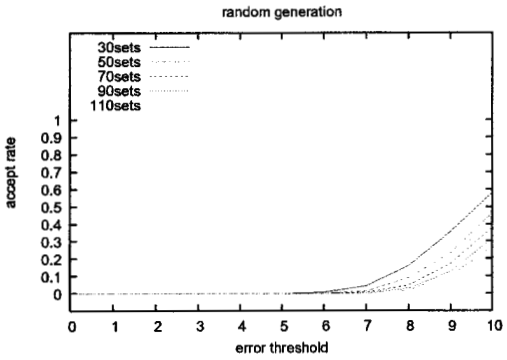

Fig. 1 ランダム生成した特徴点配置による受理率

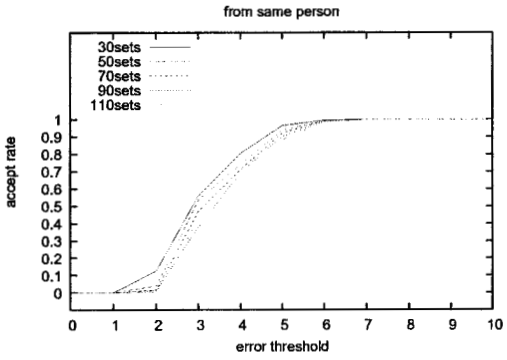

Fig. 2 同一人物の別の指紋画像から生成した特徴点配置による受理率

		Fig.1	Fig.2
$d = 5$	$\lvert\mathcal{B}\rvert = 70$	0.0000	0.9233
	$\lvert\mathcal{B}\rvert = 90$	0.0000	0.9038
	$\lvert\mathcal{B}\rvert = 110$	0.0000	0.8841
$d = 6$	$\lvert\mathcal{B}\rvert = 70$	0.0003	0.9920
	$\lvert\mathcal{B}\rvert = 90$	0.0000	0.9895
	$\lvert\mathcal{B}\rvert = 110$	0.0000	0.9875
$d = 7$	$\lvert\mathcal{B}\rvert = 70$	0.0070	1.0000
	$\lvert\mathcal{B}\rvert = 90$	0.0037	0.9988
	$\lvert\mathcal{B}\rvert = 110$	0.0022	0.9991

Table 1 各々の生成方法による受理率

れは $\mathcal{B}$ が制約条件の数を、d が同一とみなす許容範囲を、それぞれ直感的に意味することを考えれば容易に理解できることといえる。

今回得られた結果で注目すべき点はFig.2の方がFig.1のものよりも小さな d で高い受理率を達成している点である。このことから、Fig.2で生成された特徴点分布、つまり本人の指紋から得られた特徴点分布の方がFig.1での特徴点分布、つまり他人の指紋から得られた(ものを示唆する)特徴点分布よりも、総じて Δ-MA インスタンスとして ϕ_0 に似ていると判断されていることが分かり、意図したような結果が得られていることが分かる。また、$\mathcal{B}$ と受理率の関係を見てみると、$\mathcal{B}$ を大きくしたとき、Fig.1の方がFig.2よりも大きく受理率が落ちていることが分かる。このことは、$\mathcal{B}$ を大きくとることで、特徴点分布としての類似性をより正確に保持しつつ変換が可能になることを示唆しているといえ、従って認証精度をあげるのに有効であると考えられる。しかしながら、前述のとおり今回提唱しているプロトコルは実行に $O(|\mathcal{B}|^2)$ 回の通信を要するため、実際には認証精度と実行時間のトレードオフの関係が生まれると考えられる。

7 まとめ

以上の結果から、定義.3で提案したプロトコルによって1%未満のFAR/FRRを達成することが確認できた。もちろん、具体的な実験環境やサンプルデータの差異などがあるため直接比較することは出来ないが、少なくとも数値の上ではこの結果は従来の同様の手法に比べて高い精度での指紋認証を達成することが出来たと考えることが出来、また、認証のために必要とされる情報が利用者の指紋情報のみであるため、利用者がパスワード等を覚えておく必要はなく、利便性の高い認証システムであるといえる。この結果は生体認証技術の面からいえば、より安全で便利な指紋認証を実現するための一手法を提案したものであり、またゼロ知識証明の面からいえば、誤差を含むような問題をゼロ知識証明するための一つの手法を提案したということができる。今回は特徴点分布間の類似性を対象にして構成を行ったが、対象とする問題を適切に定義、利用することで誤差を含む問題全般に利用できるものを構成可能であると考えられる。

しかしながら、今回の研究には課題や問題点がいくつか存在する。まず第一に、前述のとおり実験の中で手動で行った部分が存在することである。具体的には、特徴点の抽出及び利用する指紋同士の位置あわせの二点で、実際にこれらの作業を機械的に実行させることを考えたときに、具体的にどのように実現すればよいのか、手動と比べどの程度の精度で作業が出来るのか、などを考える必要がある。また、実験で利用したサンプル量も問題といえる。実験で利用した指紋画像は1セット8枚のみであり、これは抽出作業を手作業で行うことに伴う困難さに起因している。抽出作業を機械化できれば大量のサンプルに対して実験を行うことが可能になると考えられる。こういった実験場の問題点に加えて、提案プロトコル自体にも問題点は存在すると考えられる。具体的には実行時間の問題で、定義.3で触れたように今回のプロトコルはもっとも基本的なゼロ知識証明として構成を行ったため、実行するのに $O(|\mathcal{B}|^2)$ だけの通信コストがかかってしまう。ゼロ知識証明の研究の中でこの問題は克服されているものの [2]、本研究ではその構成の困難さから利用していない。この点も本研究の課題であるといえる。

8 付録

定義 4 関数 $d : \mathbb{N}^3 \times \mathbb{N}^3 \to \mathbb{N}$ を、以下のようにして定義する。はじめに、自然数の三つ組 $m_1, m_2 \in \mathbb{N}^3$ が与えられたとき、以下の二種類の操作を考える。

1. $(a_1, a_2, a_3) \in \mathbb{N}^3$ からひとつの数 a_i を選んできて1足し合わせた後、降順にソートしなおす
2. $(a_1, a_2, a_3) \in \mathbb{N}^3$ からひとつの正数 a_i を選んできて1減じた後、降順にソートしなおす

一つ目の操作をコスト1、二つ目の操作をコスト0で行うことができるとして、m_1 から m_2 へと遷移する任意の操作列の中で、コストの総和が最小になるときの値を $v(m_1, m_2)$ として定義し、$d(m_1, m_2)$ を以下のようにして定義する。

$$
\begin{aligned}
d(m_1, m_2) &:= \max(v(m_1, m_2), w(m_1, m_2)) \\
w(m_1, m_2) &:= (a_1^{(1)} + a_2^{(1)} + a_3^{(1)}) \\
&\quad -(a_1^{(2)} + a_2^{(2)} + a_3^{(2)}) \\
m_i &:= (a_1^{(i)}, a_2^{(i)}, a_3^{(i)}) \quad (i = 1, 2)
\end{aligned}
$$

定義 5 二つの Δ-*MA* インスタンス $(\mathcal{B}_1, \varphi_1, K_1)$、$(\mathcal{B}_2, \varphi_2, K_2)$ が以下の条件を満たしているとき、この二つは *isomorphic* であると呼ぶ。

$$
\begin{aligned}
&K_1 = K_2 \\
&|\mathcal{B}_1| = |\mathcal{B}_2| \\
&{}^{\exists}\pi : \mathcal{B}_1 \to \mathcal{B}_2 \quad s.t. \\
&\begin{cases}
\pi \text{ は全単射} \\
{}^{\forall}B^{(1)} \in \mathcal{B}_1, \quad \varphi_1(B^{(1)}) = \varphi_2(\pi(B^{(1)})) \\
{}^{\forall}\mathcal{S} \subseteq \mathcal{B}_1, \\
\bigcap_{B^{(1)} \in \mathcal{S}} B^{(1)} \setminus \bigcup_{B^{(1)} \in \mathcal{B} \setminus \mathcal{S}} B^{(1)} = \emptyset \\
\Leftrightarrow \bigcap_{B^{(1)} \in \mathcal{S}} \pi(B^{(1)}) \setminus \bigcup_{B^{(1)} \in \mathcal{B} \setminus \mathcal{S}} \pi(B^{(1)}) = \emptyset
\end{cases}
\end{aligned}
$$

関数 $\pi : \mathcal{B}_1 \to \mathcal{B}_2$ のことを $(\mathcal{B}_1, \varphi_1, K_1)$ と $(\mathcal{B}_2, \varphi_2, K_2)$ の間の *isomorphism* と呼ぶ。

参考文献

1) A. Fiat and A. Shamir. How to prove yourself: practical solutions to identification and signature problems. In *Advances in Cryptology.Crypto '86*, pp. 186–194, New York, 1987. Springer–Verlag.

2) O. Goldreich and A. Kahan. How to construct constant-round zero-knowledge proof systems for np. *Journal of Cryptology*, Vol. 9, No. 3, pp. 167–189, 1996. Preliminary versions date to 1988.

3) O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems. *Journal of the ACM*, Vol. 38, No. 1, pp. 691–729, 1991. Preliminary versions in 27th FOCS, 1986.

4) S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal of Computing*, Vol. 18, No. 1, pp. 186–208, 1989. Preliminary version in 17th STOC, 1985.

5) N. K. Ratha, J. H. Connell, and R. M. Bolle. Enhancing security and privacy in biometric-based authentication systems. *IBM System Journal*, Vol. 40, No. 3, 2001.

6) 永井慧, 菊池浩明, 尾形わかは, 西垣正勝. "zerobio-秘匿ニューラルネットワーク評価を用いた指紋認証システム". In *Proceedings of Computer Security Symposium 2006*, pp. 633–638, 2006.

7) 菊池浩明, 尾形わかは, 西垣正勝. 多項式の根のゼロ知識証明とリモートバイオメトリクスへの応用. In *Proceedings of Symposium on Cryptography and Information Security 2007*, p. 42, 2007.

8) 高橋健太, 比良田真史. セキュアなリモート生体認証プロトコルの提案. In *Proceedings of Symposium on Cryptography and Information Security 2007*, p. 301, 2007.