虹彩コードを秘匿する虹彩認証方式の提案

太田陽基[†] 清本晋作[†] 田中俊昭[†]

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，各人固有の身体的特徴などを用いて本人確認を行う生体認証技術が提案されている．生体認証に使用する生体情報は変更できないため，一度他人に盗まれると二度と認証に使用できなくなる．したがって，生体認証では特に生体情報の管理を厳重にする必要がある．しかし，管理者側で本人確認を行うサーバ認証モデルでは，生体情報をサーバに送信するため，生体情報が外部に露呈する．よって，生体情報は暗号化されることが望ましい．通信ネットワーク上は暗号化などにより生体情報を秘匿可能であるが，認証時には復号しなくてはならない．センサなどによって読みとられる生体情報には，周囲の環境状況や利用者の動作の違いなどによる微少な誤差が生じる．それゆえ，登録情報と入力情報の類似度の高さにより認証される生体認証では，生体情報を暗号化したまま認証できない．そのため，従来の方式ではサーバの管理者には利用者の生体情報を知られてしまう．そこで，生体認証の中でも虹彩認証に主眼を置き，虹彩コードに変換関数を施すことにより，虹彩コードを秘匿する手法を提案する．そして，変換関数の利用により，虹彩コードを秘匿したままサーバにおける虹彩認証が可能なこと，およびその変換関数が虹彩コードを秘匿するのに十分な安全性を有することを示す．

Proposal of an Iris Identification Scheme Hiding Iris Codes

Haruki Ota,[†] Shinsaku Kiyomoto[†] and Toshiaki Tanaka[†]

Biometric authentication is one of the personal identification schemes based on individual physical or behavioral characteristics. Biometric data extracted from biometrics used in biometric authentication should be hidden, not to be stolen by others. Nonetheless, server administrators could obtain biometric data, since it is not encrypted on the server in the authentication process. In this case, a cryptographic scheme cannot be simply applied to hide biometric data, because of the following reasons. The enrolled biometric data and the newly inputted biometric data are not exactly the same in biometric authentication, even if they are captured from the same person. If the matching score between both data is equal to or greater than the stated threshold, the person with the newly inputted data is regarded as the person with the enrolled data. In this paper, we propose an iris identification scheme on the server, hiding the user's raw iris data, an "iris code." We first consider the method to transform it into another code. It is essential for the method to conserve the normalized Hamming distance between the enrolled iris code and the newly inputted iris code. We then show that the method is sufficiently secure to hide the iris codes.

1. まえがき

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次の 3 種類に分けられる[1)]．

(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．

(1) の認証はパスワードや暗証番号などを用いた認証である．この認証は現在最も広く利用されており，容易に実現可能な手段であるが，利用者がパスワードなどを忘れやすい，忘れないよう安易にすると他人に推測されやすい，といった欠点がある．(2) の認証は IC カードや磁気カードなどを用いた認証である．この認証は携帯性に優れ，操作も容易な手段であるが，IC カードなどの紛失や盗難の危険性がある．(3) の認証は本稿で扱う虹彩などの各人固有の生体情報を用いた認証である．この認証は特別な装置や高度な処理ソフトウェアを必要とするなどの今後解決すべき課題は存在するが，(1) の認証のように記憶する必要も (2) の認証のように紛失するおそれもないという長所がある．

しかし，(3) の生体認証においても他人による盗難

† 株式会社 KDDI 研究所
KDDI R&D Laboratories, Inc.

の危険性が存在する．生体認証は特に，他の認証手段より秘密情報の管理を厳重にする必要がある．パスワードやICカード内の秘密情報はたとえ盗まれても変更可能であるが，生体情報は変更できないため，一度盗まれると二度と認証に使用できなくなるからである．生体認証は，利用者側において本人確認を行うクライアント認証モデルと管理者側において本人確認を行うサーバ認証モデルの2つに大別される[1),2)]．クライアント認証モデルにおける生体情報を秘匿する方法として，耐タンパ性を有する処理機能組み込み型のICカードを利用する方法が提案されている[1),3)]．この方法では，ICカードに生体情報を保管しカード内で認証処理を行うため，生体情報がICカードの外部に露呈しない．それに対し，サーバ認証モデルでは，生体情報をサーバに送信する必要があるため，生体情報が外部に露呈する．よって，生体情報は暗号化により秘匿されることが望ましい．通信ネットワーク上は暗号化により生体情報を秘匿することができるとしても，認証時には復号しなくてはならない．センサなどによって読みとられる生体情報には，周囲の環境状況や利用者の動作の違いなどによる微少な誤差が生じる．それゆえ，登録情報と入力情報の類似度の高さにより認証される生体認証では，生体情報を暗号化したまま認証することができない．そのため，従来の方式ではサーバの管理者には利用者の生体情報を知られてしまい，この利用者が他のサーバにおいても同じアルゴリズムの生体認証を利用している場合，サーバの管理者によってなりすまされるおそれが出てくる．この問題を解決する生体情報を秘匿する手法が，指紋認証に関していくつか提案されている[4)~6)]．しかし，生体認証の中でも指紋認証と同程度の実用性と高精度を誇る虹彩認証に関しては，筆者らによる手法[7)]以外にはほとんど提案されていない．そのうえ，文献4)〜6)の指紋認証に対する手法は，次の2つの理由から虹彩認証に適用することができない．

- 生体の特徴抽出の方法と対応する照合・判定の方法に関し，文献4)〜6)で使用されている各指紋認証アルゴリズムと本稿で用いる虹彩認証アルゴリズムが異なるため．
- 文献4)，5)の手法が正当性の観点から，文献6)の手法が実用性の観点からそれぞれ不十分であるため．

そこで，虹彩認証に関して，上記の問題を解決する新たな手法が必要である．

本稿では，生体認証の中でも虹彩認証に主眼を置き，虹彩画像から特徴量を抽出して得られる「虹彩コード」に変換処理（以下，変換関数）を施すことにより，虹彩コードを秘匿する手法を提案する．そして，変換関数を利用することにより，虹彩コードを秘匿したままサーバにおける虹彩認証が可能なこと，およびその変換関数が虹彩コードを秘匿するのに十分な安全性を有することを示す．

本稿は以下のように構成されている．2章では，生体認証を概説し，特に本稿で扱う虹彩認証について述べている．3章が本稿の核となる章である．この章では，虹彩コードを秘匿する変換関数について述べ，変換関数を利用したサーバにおける虹彩認証方式を提案している．4章では，3章で述べた変換関数の正当性と安全性について評価し，提案方式の有効性を示している．

2. 生体認証

本章では，生体認証について概説し，特に本稿で扱う虹彩認証について詳説する．

2.1 概　　要

本節では，生体認証の概要を説明する．

生体認証は，各人固有の生体情報を用いて本人確認を行う認証のことであり，指紋や顔，虹彩などの身体的特徴を利用する方法と声紋や動的署名などの行動的特徴を利用する方法の2つに大別される．主な生体認証技術の比較を**表1**に示す[1),8)]．ただし，記号H，M，Lはそれぞれ High，Medium，Low レベルの性能を表している．文献1)，8)には，表1の技術以外にも，身体的特徴を利用する方法として，顔の赤外画像，網膜，静脈，掌形，耳，匂い，DNA（deoxyribonucleic acid），行動的特徴を利用する方法として，キーストローク，歩行があげられている．表1から，身体的特徴による認証の方が行動的特徴による認証より全体的に高性能であることが分かる．身体的特徴による認証の中でも，特に指紋認証と虹彩認証が高性能かつ高精度であり，すでに実用化されている．指紋認証につい

表1 生体認証技術の比較
Table 1 Comparison of biometric authentication technologies.

分類	身体的特徴			行動的特徴	
生体情報	指紋	顔	虹彩	声紋	動的署名
一般性	M	H	H	M	L
唯一性	H	L	H	L	L
永続性	H	M	H	L	L
収集性	M	H	M	M	H
精度	H	L	H	L	L
受容性	M	H	L	H	H
脅威耐性	H	L	H	L	L

ては，生体情報を秘匿する手法がいくつか提案されている[4)~6)]．しかし，虹彩認証については，筆者らによる手法が提案されている[7)]程度で，十分に議論されていない．

2.2 虹彩認証

本節では，本稿で扱う虹彩認証について述べる．

虹彩とは，黒目の内側で瞳孔より外側のドーナツ状の筋肉質部分のことである．人の目は妊娠6カ月の胎児の頃までに形成され，その時点でつくられる瞳孔から外側に向かってカオス状の皺が発生する．この皺は生後2年ほどで成長が止まり，それ以降変化しない．虹彩の模様は各人固有のパターンであり，同一人の左右の目でも一卵性双生児の目でも異なるパターンになる．眼球内部の疾病や目の充血，また目の不自由な人に対しても，虹彩は影響を受けず，虹彩認証精度の劣化にはならない．このように，虹彩は生体認証の「万人不同」「終生不変」という性質を有している．

そこで，一般的な虹彩認証アルゴリズムにおける登録手順と認証手順を説明する[1),9),10)]．以降，$\mathcal{X} = \{0, 1\}$とする．

虹彩認証アルゴリズムにおける登録手順は次のようにして行われる．

(1) カメラを用いて撮影することにより利用者の目の画像を取得する．その際，生体検知を行い，生きた人間の目であることを確認する．

(2) 手順(1)において取得した目の画像から，画像輝度の変化を利用して強膜側境界，瞳孔側境界，上下瞼側境界を決定して虹彩領域を特定し，虹彩画像のみを切り出す．

(3) 手順(2)において特定した虹彩領域に8つの環状解析ゾーンを割り当て，その環状解析ゾーンの走査により，虹彩コードを取得する．

(4) 手順(3)において得られたn [bit]の虹彩コード$\boldsymbol{A} = (A_1 \cdots A_n)$ $(\in \mathcal{X}^n)$をサーバに送信し，サーバのデータベースにテンプレートデータとして登録する．

ただし，虹彩コード長 n は主に 2,048 [bit], 4,096 [bit] が使われている．続いて，虹彩認証アルゴリズムにおける認証手順は次のようにして行われる．

(1) 登録手順(1)〜(3)を繰り返すことにより得られた虹彩コードをサーバに送信する．

(2) 手順(1)において得られたn [bit]の入力虹彩コード$\boldsymbol{B} = (B_1 \cdots B_n)$ $(\in \mathcal{X}^n)$と登録手順(4)における登録虹彩コード$\boldsymbol{A}$間の正規化ハミング距離 HD_{org}

$$HD_{\mathrm{org}} = \frac{1}{n}\sum_{j=1}^{n}(A_j \oplus B_j)$$

をサーバにおいて計算する（ただし，$\oplus$ 記号は排他的論理和を表している）．

(3) 手順(2)において得られた正規化ハミング距離 HD_{org} がある閾値以下であるならば利用者を本人であると判定し，ある閾値より大きければ利用者を他人であると判定する．

通常，閾値は統計的に得られた本人拒否率（本人を誤って他人と見なす割合）と他人受け入れ率（他人を誤って本人と見なす割合）が一致する値に設定される．ただ，サービスの利用形態によっては，トレードオフの関係にある両者を考慮して，サーバの管理者が任意に閾値を設定することができる．また，登録時や認証時に虹彩コードをサーバに送信する場合には，通信ネットワーク上における他人の盗聴などを防止するため，虹彩コードに暗号化などの処理を施す必要がある．しかし，上記の従来方式では，登録虹彩コードと入力虹彩コード間の正規化ハミング距離を計算することから，暗号化された虹彩コードを認証時に復号しなくてはならない．そのため，サーバの管理者は利用者の虹彩コードを容易に入手でき，この利用者が他のサーバにおいても同じアルゴリズムの虹彩認証を利用している場合，サーバの管理者はその利用者になりすますことが可能になる．したがって，認証時も虹彩コードを秘匿する必要がある．

そこで，3章において，サーバの管理者にも虹彩コードを知られることなく，虹彩認証を可能にする方式を提案する．

3. 提案方式

本章では，虹彩コードを秘匿する手法について述べ，その手法を用いたサーバにおける虹彩認証方式を提案する．

3.1 虹彩コードの秘匿手法の提案

本節では，虹彩コードを秘匿する手法について説明する．

2.2節の虹彩認証アルゴリズムにおいて示したように，虹彩認証は登録虹彩コードと入力虹彩コードの類似度の高さにより行う認証である．つまり，登録虹彩コードと入力虹彩コード間の正規化ハミング距離が設定された閾値以下であるときに本人と見なす認証である．そのため，虹彩コードを秘匿する手段に暗号化関数やパスワード認証の際に利用される MD5（Message Digest 5）などの一方向性ハッシュ関数を利用するこ

とができない．なぜなら，暗号化関数や一方向性ハッシュ関数では，登録虹彩コードと入力虹彩コード間の（正規化）ハミング距離が保存されないからである．

そこで，虹彩コードを秘匿するために施される変換手法の要件は次の 2 つであると考えられる．

(1) 変換前後において登録虹彩コードと入力虹彩コード間の（正規化）ハミング距離が保存される．

(2) 変換後の虹彩コードから変換前の虹彩コードを容易に推測できない．

これら 2 要件を満足する変換手法の概要を以下に述べる．変換手法は次の 3 つの処理から構成される．

(1) 拡大処理．

(2) 並べ替え処理．

(3) 回転処理．

処理 (1) の拡大処理は，n [bit] の虹彩コードに n [bit] の乱数を結合して，$2n$ [bit] の拡大虹彩コードに変換する処理である．処理 (2) の並べ替え処理は，処理 (1) で得られた $2n$ [bit] の拡大虹彩コードをある規則に従って並べ替えを行い，$2n$ [bit] の並べ替え虹彩コードに変換する処理である．処理 (3) の回転処理は，処理 (2) で得られた $2n$ [bit] の並べ替え虹彩コードを以下のようにして，$2n$ [bit] の変換虹彩コードに変換する処理である．まず，$2n$ [bit] の並べ替え虹彩コードを $2n$ 次元ベクトルに変換する．ただし，変換規則は

$$\begin{cases} 1 & \longleftrightarrow & \dfrac{1}{\sqrt{2}} \\ 0 & \longleftrightarrow & -\dfrac{1}{\sqrt{2}} \end{cases} \tag{1}$$

と定める．次に，$2n$ 次元ベクトル中の任意の 2 要素を抜き出し，2 次元ベクトルを構成する．このとき，この 2 次元ベクトルは原点 $(0,0)$ 中心，半径 1 の 2 次元円周上の 45 [deg]，135 [deg]，225 [deg]，315 [deg]（横軸正の方向を 0 [deg] とした場合）に位置する座標のいずれかになる．そして，この円周上の点を 0 [deg]，90 [deg]，180 [deg]，270 [deg] のいずれかの角度だけ原点中心の（反時計回りの）回転変換を行い，得られた値を $2n$ 次元ベクトルの要素として抜き出したもとの位置に戻す．2 要素の抜き出し・回転変換・もとの位置に戻す，という処理を $(3n/2)$ 回繰り返し，最後に得られた $2n$ 次元ベクトルを式 (1) の変換規則を用いて 2 進変換する．このようにして得られたものが $2n$ [bit] の変換虹彩コードとなる．

以下に処理 (3) の回転処理の例を示す（**図 1** 参照）．ここでは，並べ替え虹彩コード中の 2 [bit] の要素 “11” を変換することを考える．このとき，式 (1) の変換規則より原点中心，半径 1 の 2 次元円周上の 45 [deg] に位置する座標 $(1/\sqrt{2}, 1/\sqrt{2})$ に変換される．そして，点 $(1/\sqrt{2}, 1/\sqrt{2})$ に原点中心の 180 [deg] 回転変換を施すと仮定すると，得られる座標は $(-1/\sqrt{2}, -1/\sqrt{2})$ となる．したがって，両座標はともに $(-1/\sqrt{2})$ であるから，式 (1) の変換規則より最終的に “00” に変換される．すなわち，この例では “11” が “00” に変換されたことになる．

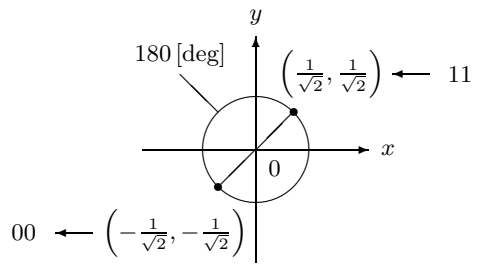

図 1 回転処理の例
Fig. 1 An example for rotation processing.

具体的には，次の手順で虹彩コード $\boldsymbol{X} = (X_1 \cdots X_n)$ $(\in \mathcal{X}^n)$ に変換を施す（**図 2** 参照）．ただし，虹彩コード $\boldsymbol{X}$ は登録虹彩コード $\boldsymbol{A}$，入力虹彩コード $\boldsymbol{B}$ の両方を表している．

(1) 安全性の保証されている乱数生成アルゴリズムを用いて，

$$\left\{ n + \log_2 2n + (2\log_2 2n + 2) \cdot \frac{3n}{2} \right\} \text{ [bit]}$$

の乱数 $\boldsymbol{R} = (\boldsymbol{R}_e \boldsymbol{R}_p \boldsymbol{R}_1 \cdots \boldsymbol{R}_j \cdots \boldsymbol{R}_{3n/2})$ $(\boldsymbol{R}_e \in \mathcal{X}^n, \boldsymbol{R}_p \in \mathcal{X}^{\log_2 2n}, \boldsymbol{R}_j \in \mathcal{X}^{(2\log_2 2n+2)}$ $(j = 1, \cdots, 3n/2))$ を生成する．ただし，

$$\boldsymbol{R}_j = (\boldsymbol{u}_j \boldsymbol{v}_j \boldsymbol{\theta}_j)$$
$$(\boldsymbol{u}_j, \boldsymbol{v}_j \in \mathcal{X}^{\log_2 2n}, \boldsymbol{\theta}_j \in \mathcal{X}^2)$$

である．また，生成された乱数は他人に知られないよう安全に保管され，虹彩コードを登録するサーバごとに異なるものとする．

(2) 手順 (1) において得られた乱数 $\boldsymbol{R}$ の要素 $\boldsymbol{R}_e$ を n [bit] の虹彩コード $\boldsymbol{X}$ に結合して，$2n$ [bit] の拡大虹彩コード $\boldsymbol{Y} = (Y_1 \cdots Y_{2n})$ $(\in \mathcal{X}^{2n})$

$$\boldsymbol{Y} = (\boldsymbol{X} \,||\, \boldsymbol{R}_e)$$

に変換する．ただし，“||” 記号は前後の結合を表している．

(3) 手順 (2) において得られた $2n$ [bit] の拡大虹彩コード $\boldsymbol{Y}$ に対し，手順 (1) において得られた乱数 $\boldsymbol{R}$ の要素 $\boldsymbol{R}_p$ によって選択される並べ替え規則に従い，$2n$ [bit] の並べ替え虹彩コード $\boldsymbol{Z} = (Z_1 \cdots Z_{2n})$ $(\in \mathcal{X}^{2n})$ に変換を施す．

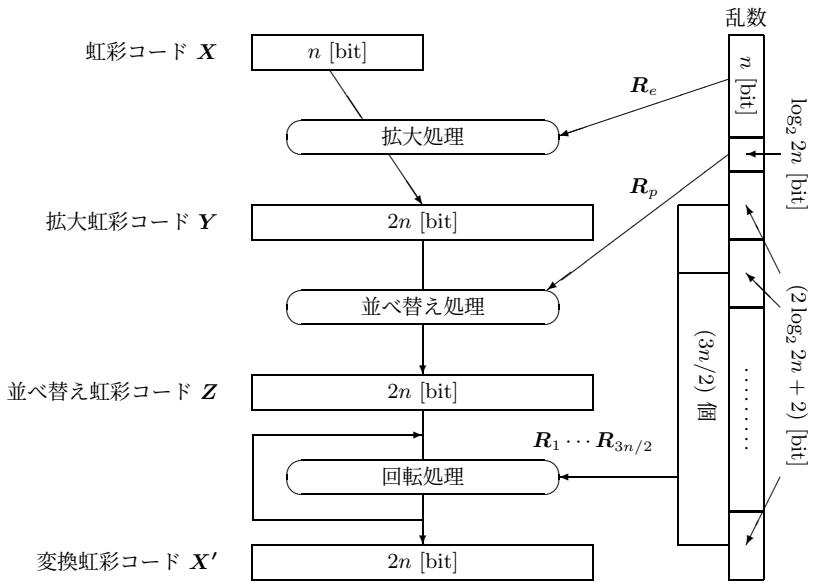

図 2 変換手法のブロック図
Fig. 2 Block diagram for transformation method.

ただし，並べ替え規則は $2n$ 個あり，1〜$2n$ までの $2n$ 個の数字をランダムに並べた行が，$2n$ 行あるような変換テーブルである．乱数 $\boldsymbol{R}_p$ により変換テーブルの使用する行を決め，その行に記載された番号に $2n$ [bit] の拡大虹彩コード $\boldsymbol{Y}$ のインデックスを対応させて並べ替える．本来，$2n$ [bit] の拡大虹彩コード $\boldsymbol{Y}$ の並べ替えは，ビットの重複を許した場合，全部で $(2n)!$ 通り存在する．しかし，必要な乱数の長さは短い方が望ましい．そこで，並べ替え規則は，$2n$ [bit] の拡大虹彩コード $\boldsymbol{Y}$ がオール 0（またはオール 1）であるときを除いたすべての並べ替え数の最小値（ビットの重複を許さない場合）である $2n$ 個に設定した．また，変換テーブルは ${}_{(2n)!}\mathrm{C}_{2n}$ 通り作成可能であるため，ユーザごと別個に設定し，安全に保管されているものとする．

(4) 手順 (3) において得られた $2n$ [bit] の並べ替え虹彩コード $\boldsymbol{Z}$ を

$$\begin{matrix} W_i \\ W'_i \end{matrix} = \begin{cases} \dfrac{1}{\sqrt{2}} & \text{for} \quad Z_i = 1 \\ -\dfrac{1}{\sqrt{2}} & \text{for} \quad Z_i = 0 \end{cases} \qquad (i = 1, \cdots, 2n)$$

に従い，$2n$ 次元ベクトル $\boldsymbol{W} = (W_1, \cdots, W_{2n})$ $(\in \mathcal{R}^{2n})$ と変換 $2n$ 次元ベクトル $\boldsymbol{W'} = (W'_1, \cdots, W'_{2n})$ $(\in \mathcal{R}^{2n})$ にそれぞれ変換する．ただし，$\mathcal{R}$ は実数の集合を表している．

(5) $j = 1$ とする．

(6) 手順 (1) において得られた乱数 $\boldsymbol{R}$ の要素 $\boldsymbol{R}_j$ に対し，$2n$ 次元ベクトル $\boldsymbol{W}$ からそれぞれ $([\boldsymbol{u}_j]_{10} + 1)$，$([\boldsymbol{v}_j]_{10} + 1)$ 番目の要素 $W_{[\boldsymbol{u}_j]_{10}+1}$，$W_{[\boldsymbol{v}_j]_{10}+1}$ を抜き出し，2 次元ベクトル $(W_{[\boldsymbol{u}_j]_{10}+1}, W_{[\boldsymbol{v}_j]_{10}+1})$ を構成する．ただし，$[a]_{10}$ は a の 10 進表現を表している．

(7) 手順 (6) において得られた 2 次元ベクトル $(W_{[\boldsymbol{u}_j]_{10}+1}, W_{[\boldsymbol{v}_j]_{10}+1})$ を原点中心に $([\boldsymbol{\theta}_j]_{10}+1)$ 番目の角度（以下，θ_j [deg]）だけ回転して，2 次元ベクトル $(W'_{[\boldsymbol{u}_j]_{10}+1}, W'_{[\boldsymbol{v}_j]_{10}+1})$ に移す変換を行う．すなわち，

$$\begin{aligned} &\begin{pmatrix} W'_{[\boldsymbol{u}_j]_{10}+1} \\ W'_{[\boldsymbol{v}_j]_{10}+1} \end{pmatrix} \\ &= \begin{pmatrix} \cos\theta_j & -\sin\theta_j \\ \sin\theta_j & \cos\theta_j \end{pmatrix} \begin{pmatrix} W_{[\boldsymbol{u}_j]_{10}+1} \\ W_{[\boldsymbol{v}_j]_{10}+1} \end{pmatrix} \\ &= \begin{pmatrix} W_{[\boldsymbol{u}_j]_{10}+1} \cdot \cos\theta_j - W_{[\boldsymbol{v}_j]_{10}+1} \cdot \sin\theta_j \\ W_{[\boldsymbol{u}_j]_{10}+1} \cdot \sin\theta_j + W_{[\boldsymbol{v}_j]_{10}+1} \cdot \cos\theta_j \end{pmatrix} \end{aligned}$$

となる．ただし，θ_j は

$$\theta_j = 90 \cdot [\boldsymbol{\theta}_j]_{10} \qquad (2)$$

と定める．そして，得られた $W'_{[\boldsymbol{u}_j]_{10}+1}$，$W'_{[\boldsymbol{v}_j]_{10}+1}$ を変換 $2n$ 次元ベクトル $\boldsymbol{W'}$ の要素として抜き出したもとの位置に戻す（すでに要素が存在している場合は置き換える）．

(8) $j = j + 1$ として，手順 (6) と (7) を $j \leq$

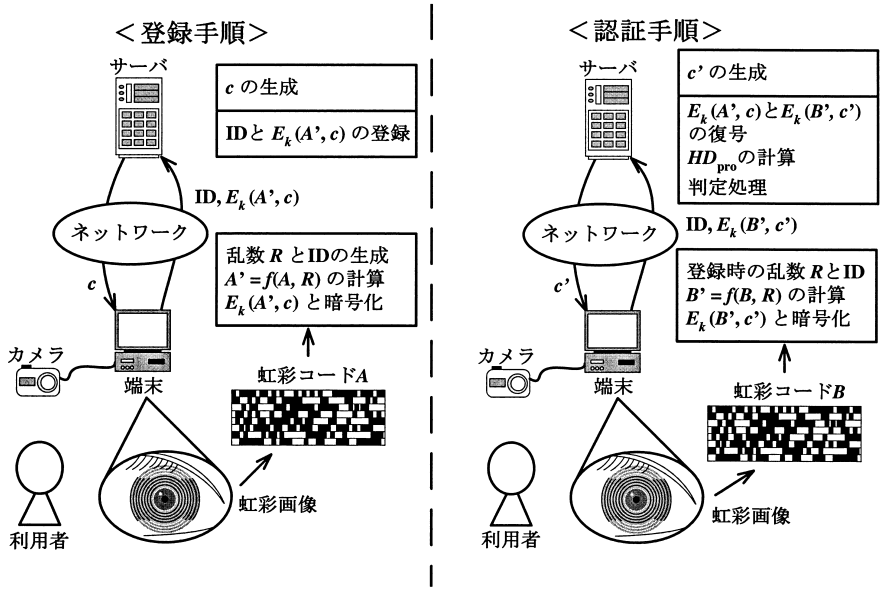

図 3 提案方式の手順
Fig. 3 Two procedures of the proposed scheme.

$(3n/2)$ の成立している間繰り返す．ただし，$2 \leq j \leq (3n/2)$ では，手順 (6) の「$2n$ 次元ベクトル $\boldsymbol{W}$」を「変換 $2n$ 次元ベクトル $\boldsymbol{W'}$」に置き換え，それ以降の手順では，変換後の $2n$ 次元ベクトルや 2 次元ベクトル，各要素を用いることとする．これは，$2n$ 次元ベクトル $\boldsymbol{W}$ の要素は複数回選択される可能性があり，2 回目以降はそれ以前までの回転変換後の要素を用いるためである．また，$W_{[\boldsymbol{u}_j]_{10}+1}$ と $W_{[\boldsymbol{v}_j]_{10}+1}$ が等しい場合も，同様の手順で回転処理を行うことができる．これらの等しい要素を用いて，2 次元ベクトル $(W_{[\boldsymbol{u}_j]_{10}+1}, W_{[\boldsymbol{v}_j]_{10}+1})$ を構成し，手順どおりの回転変換により，2 次元ベクトル $(W'_{[\boldsymbol{u}_j]_{10}+1}, W'_{[\boldsymbol{v}_j]_{10}+1})$ を得る．そして，得られた $W'_{[\boldsymbol{u}_j]_{10}+1}$，$W'_{[\boldsymbol{v}_j]_{10}+1}$ を変換 $2n$ 次元ベクトル $\boldsymbol{W'}$ の要素としてもとに位置に戻す際に，$W'_{[\boldsymbol{u}_j]_{10}+1}$，$W'_{[\boldsymbol{v}_j]_{10}+1}$ の順に戻し，前者の要素を後者の要素に置き換える．

(9) 手順 (8) において得られた変換 $2n$ 次元ベクトル $\boldsymbol{W'}$ を

$$X'_i = \begin{cases} 1 & \text{for} \quad W'_i = \dfrac{1}{\sqrt{2}} \\ 0 & \text{for} \quad W'_i = -\dfrac{1}{\sqrt{2}} \end{cases} \quad (i = 1, \cdots, 2n)$$

に従い，変換虹彩コード $\boldsymbol{X'} = (X'_1 \cdots X'_{2n})$ $(\in \mathcal{X}^{2n})$ に変換する．

以上の手順にて得られた $\boldsymbol{X'}$ が変換虹彩コードとなる．上記の変換を次のような関数として定義する．

定義 1 n [bit] の虹彩コードと $\{n + \log_2 2n + (2\log_2 2n + 2)(3n/2)\}$ [bit] の乱数に対し，上記の変換により $2n$ [bit] の変換虹彩コードを出力する関数を

$$f : \mathcal{X}^n \times \mathcal{X}^{\{n+\log_2 2n+(2\log_2 2n+2)(3n/2)\}} \to \mathcal{X}^{2n}$$

と定める．ただし，関数 f の計算時に虹彩コードと乱数はともに他人に奪われることがないものとする．□

3.2 節で，定義 1 の関数を利用した虹彩認証方式を提案する．

3.2 虹彩認証方式の提案

本節では，3.1 節で定義した関数を利用することにより，虹彩コードを秘匿したままサーバにおける虹彩認証が可能な方式を提案する．

定義 1 の関数を利用した提案方式の登録手順と認証手順を説明する（**図 3** 参照）．また，関数 E_k を次のように定義する．

定義 2 サーバの公開鍵を k とすると，$2n$ [bit] の変換虹彩コードと $2n$ [bit] のチャレンジコードに対し，$2n$ [bit] 出力の公開鍵暗号化関数 E_k を

$$E_k : \mathcal{X}^{2n} \times \mathcal{X}^{2n} \to \mathcal{X}^{2n}$$

と定める． □

提案方式の登録手順は次のようにして行われる．ただし，虹彩コードを取得する方法は文献 9), 11) に従い，この安全性については本稿の議論の対象外とする．

(1) 利用者はサーバに登録のリクエストをし，サー

バが生成した $2n$ [bit] のチャレンジコード $\boldsymbol{c} = (c_1 \cdots c_{2n})$ $(\in \mathcal{X}^{2n})$ を受信する．

(2) カメラを用いて撮影することにより利用者の目の画像を取得する．その際，生体検知を行い，生きた人間の目であることを確認する．

(3) 手順 (2) において取得した目の画像から，画像輝度の変化を利用して強膜側境界，瞳孔側境界，上下瞼側境界を決定して虹彩領域を特定し，虹彩画像のみを切り出す．

(4) 手順 (3) において特定した虹彩領域に 8 つの環状解析ゾーンを割り当て，その環状解析ゾーンの走査により，虹彩コードを取得する．

(5) 手順 (4) において得られた n [bit] の虹彩コード $\boldsymbol{A}$ に対し，$\{n + \log_2 2n + (2\log_2 2n + 2)(3n/2)\}$ [bit] の乱数 $\boldsymbol{R}$ を生成し，変換虹彩コード $\boldsymbol{A'} = (A'_1 \cdots A'_{2n})$ $(\in \mathcal{X}^{2n})$
$\boldsymbol{A'} = (A'_1 \cdots A'_{2n}) = f(\boldsymbol{A}, \boldsymbol{R})$
を計算する．

(6) 手順 (5) において得られた変換虹彩コード $\boldsymbol{A'}$ に対し，$E_k(\boldsymbol{A'}, \boldsymbol{c})$ と暗号化して，自らが作成した ID とともにサーバに送信する．その後，虹彩コード $\boldsymbol{A}$，変換虹彩コード $\boldsymbol{A'}$，暗号化された変換虹彩コード $E_k(\boldsymbol{A'}, \boldsymbol{c})$，チャレンジコード $\boldsymbol{c}$ をすべて消去する．

(7) 手順 (6) において利用者から送られてきた $E_k(\boldsymbol{A'}, \boldsymbol{c})$ と ID をサーバ上のデータベースにテンプレートデータとして登録する．

続いて，提案方式の認証手順は次のようにして行われる．

(1) 利用者はサーバに認証のリクエストをし，サーバが生成した $2n$ [bit] のチャレンジコード $\boldsymbol{c'} = (c'_1 \cdots c'_{2n})$ $(\in \mathcal{X}^{2n})$ を受信する．

(2) 登録手順 (2)〜(4) を繰り返すことにより，虹彩コードを取得する．

(3) 手順 (2) において得られた n [bit] の虹彩コード $\boldsymbol{B}$ に対し，登録時に使用した乱数 $\boldsymbol{R}$ を用いて，変換虹彩コード $\boldsymbol{B'} = (B'_1 \cdots B'_{2n})$ $(\in \mathcal{X}^{2n})$
$\boldsymbol{B'} = (B'_1 \cdots B'_{2n}) = f(\boldsymbol{B}, \boldsymbol{R})$
を計算する．

(4) 手順 (3) において得られた変換虹彩コード $\boldsymbol{B'}$ に対し，$E_k(\boldsymbol{B'}, \boldsymbol{c'})$ と暗号化して，登録手順 (6) で作成した ID とともにサーバに送信する．その後，虹彩コード $\boldsymbol{B}$，変換虹彩コード $\boldsymbol{B'}$，暗号化された変換虹彩コード $E_k(\boldsymbol{B'}, \boldsymbol{c'})$，チャレンジコード $\boldsymbol{c'}$ をすべて消去する．

(5) 手順 (4) において利用者から送られてきた ID に基づく暗号化された登録変換虹彩コード $E_k(\boldsymbol{A'}, \boldsymbol{c})$ をデータベースから取得して復号し，得られた登録変換虹彩コード $\boldsymbol{A'}$ と，同じく手順 (4) において利用者から送られてきた暗号化された入力変換虹彩コード $E_k(\boldsymbol{B'}, \boldsymbol{c'})$ をサーバ上で復号し，得られた入力変換虹彩コード $\boldsymbol{B'}$ との間の正規化ハミング距離 HD_{pro}

$$HD_{\mathrm{pro}} = \frac{1}{n}\sum_{j=1}^{2n}(A'_j \oplus B'_j)$$

をサーバ上で計算する．その後，登録変換虹彩コード $\boldsymbol{A'}$，入力変換虹彩コード $\boldsymbol{B'}$，暗号化された入力変換虹彩コード $E_k(\boldsymbol{B'}, \boldsymbol{c'})$ をすべて消去する．

(6) 手順 (5) において得られた正規化ハミング距離 HD_{pro} が設定された閾値以下であるならば利用者を本人であると判定し，設定された閾値より大きければ利用者を他人であると判定する．

提案方式では，虹彩コードをサーバに送信する際もサーバで認証する際も，関数 f により虹彩コードを変換しているため，他人に虹彩コードを知られることはない．しかし，虹彩コードを変換関数 f により秘匿しても，通信ネットワーク上で変換虹彩コードを他人に不正入手されると，リプレイ攻撃をされる可能性がある．そこで，通信ネットワーク上におけるリプレイ攻撃を防止するため，提案方式ではサーバと利用者間においてチャレンジレスポンス方式を採用した．また，カメラを用いて利用者の目の画像を取得する際，生きた人間の目であることを確認するために，生体検知を行っている．虹彩認証における生体検知は瞳孔の動きや収縮などにより行われる[9]が，生体検知情報にはサーバの管理者が行う可能性のあるなりすましなどの不正に利用される情報を含んでいないとして，秘匿すべき対象としないことにする．

そこで，4 章において，変換関数 f を用いて虹彩コードを秘匿しても，サーバにおける虹彩認証が可能なこと，およびその変換関数が虹彩コードを秘匿するのに十分な安全性を有することを示す．

4. 提案方式の評価

本章では，正当性と安全性の観点から提案方式の評価を行い，提案方式の有効性を示す．

4.1 提案方式の正当性

本節では，3.1 節で述べた変換関数 f により虹彩コードを秘匿しても，サーバにおける虹彩認証が可能なことを示す．

提案方式の正当性を変換関数 f の 3 つの処理それぞれについて考察する．考察する点は，変換関数の要件 (1) の「変換前後において登録虹彩コードと入力虹彩コード間の（正規化）ハミング距離が保存される」という点である．

処理 (1) の拡大処理において，n [bit] の虹彩コードから $2n$ [bit] の拡大虹彩コードへの変換には，n [bit] の乱数の結合を行っている．登録虹彩コード，入力虹彩コードともに同じ乱数を使用しているため，乱数部分における両虹彩コード間のハミング距離は 0 である．よって，拡大処理前後において，両虹彩コード間の正規化ハミング距離は保存される．

処理 (2) の並べ替え処理において，$2n$ [bit] の拡大虹彩コードから $2n$ [bit] の並べ替え虹彩コードへの変換には，乱数により決定される並べ替え規則に従い，並べ替えを行っている．登録拡大虹彩コード，入力拡大虹彩コードともに同じ並べ替え規則を使用しているため，両虹彩コードにおける要素の位置関係は不変であり，両虹彩コード間のハミング距離も不変である．よって，並べ替え処理前後において，両虹彩コード間の正規化ハミング距離は保存される．

処理 (3) の回転処理において，$2n$ [bit] の並べ替え虹彩コードから $2n$ [bit] の変換虹彩コードへの変換には，2 要素ごとに原点中心の 2 次元回転変換を行っている．登録並べ替え虹彩コード，入力並べ替え虹彩コードに対し，回転処理では次の 3 つのポイントにより提案方式の正当性を示す．

(1) 回転変換に対する座標間の弧長の不変性．
(2) 並べ替え虹彩コードから円周上の座標への変換方法．
(3) 回転変換の回転角の設定．

ポイント (1) は，原点中心，半径 1 の円周上における任意の 2 点に対し，同じ角度だけ原点中心の回転変換を施しても，座標間の弧長は変化しない，という性質を利用したポイントである．つまり，原点中心，半径 1 の円周上の 2 点 C_1，C_2 が原点中心の角度 θ 回転によりそれぞれ C_1'，C_2' に変換されるとき，

$$\overset{\frown}{C_1C_1'} = \overset{\frown}{C_2C_2'}$$

が成立するという性質である．ポイント (2) と (3) は，並べ替え虹彩コードから円周上の座標への変換と回転変換後の座標から変換虹彩コードへの 2 進変換に対し，並べ替え虹彩コード間の（正規化）ハミング距離を保存するためのポイントである．並べ替え虹彩コード中の任意の 2 [bit] を円周上の座標へ変換することから，ポイント (2) は 90 [deg] ごとに 4 つの座標に割り当てる方法が最適である．また，ポイント (1) における座標間の弧長が不変であること，およびポイント (2) の座標が 90 [deg] ごとに位置することより，ポイント (3) は 90 [deg] の整数倍に設定することが最適である．以上の 3 つのポイントから，回転処理前後において，両虹彩コード間の正規化ハミング距離は保存される．

したがって，変換関数 f を用いて，登録虹彩コード $\boldsymbol{A}$ と入力虹彩コード $\boldsymbol{B}$ をそれぞれ，登録変換虹彩コード $\boldsymbol{A'}$ と入力変換虹彩コード $\boldsymbol{B'}$ に変換しても，提案方式における正規化ハミング距離 HD_{pro} と従来方式における正規化ハミング距離 HD_{org} に関して，

$$\begin{aligned} HD_{\mathrm{pro}} &= \frac{1}{n}\sum_{j=1}^{2n}(A'_j \oplus B'_j) \\ &= \frac{1}{n}\sum_{j=1}^{n}(A_j \oplus B_j) \\ &= HD_{\mathrm{org}} \end{aligned}$$

が成立するので，サーバにおける虹彩認証が可能になる．

4.2 提案方式の安全性

本節では，関数 f が虹彩コードを秘匿するのに十分な安全性を有することを示す．

4.2.1 攻撃モデルに関する考察

本項では，本稿において想定される攻撃モデルを明確にし，それらのモデルに関連する変換関数の性質について考察する．

本稿において想定される攻撃者（サーバの管理者）の目的は，利用者の虹彩コードを用いてその利用者になりすますことである．そのため，攻撃者は次のいずれかの攻撃を行うと考えられる．

(1) 攻撃者は利用者の虹彩コードを入手することにより，その利用者になりすます．
(2) 攻撃者は利用者の虹彩コードを知らずに，その利用者になりすます．

そのとき，各攻撃は次の攻撃モデルにそれぞれ対応する．

攻撃モデル (1)：変換虹彩コードから虹彩コードを推測する．
攻撃モデル (2)：変換虹彩コードを直接利用する．

提案方式の仮定より，生成された乱数は安全に保管されているため，攻撃者は乱数を入手することができない．よって，攻撃者は任意の虹彩コードに対する変換虹彩コードも入手できないし，サーバで得られる変換虹彩コードを虹彩コードに戻すこともできない．

そこで，攻撃モデル (1) と (2) に関連して，変換

関数の性質について考察する．処理 (1) の拡大処理において，n [bit] の虹彩コードにダミーとなる n [bit] の乱数を結合しており，処理 (2) の並べ替え処理において，$2n$ [bit] の拡大虹彩コードを安全に保管されている並べ換え規則により $2n$ [bit] の並べ替え虹彩コードに変換している．これより，n [bit] の虹彩コードが $2n$ [bit] の並べ替え虹彩コード内に混在しているため，攻撃者が虹彩コードを推測することは困難である．

処理 (3) の回転処理において，並べ替え虹彩コード中の 2 [bit] の要素から変換された 2 次元ベクトルに原点中心の回転変換を施すため，各回転変換に関しては線形変換である．この線形性は，回転変換前後において（正規化）ハミング距離を保存するために必要である．回転処理は $2n$ [bit] の並べ替え虹彩コード中から任意に 2 カ所選択する処理を $(3n/2)$ 回行っているため，要素はのべ $3n$ [bit] 分選択されることになる．この回数に設定した理由は，$2n$ [bit] の並べ替え虹彩コードの要素が一様に選択されると仮定した場合に，選択されない要素をなくし，かつ 1 回選択される要素と 2 回選択される要素の比率が等しくなるからである．実際には，同じ要素が重複して 2 回以上選択される場合と重複せずに 1 回以下選択される場合が必ず存在するので，回転処理は非線形性を有する．したがって，関数全体では非線形変換である．また，回転処理は，$2n$ [bit] の並べ替え虹彩コード中から選択された 2 [bit] と回転角の組合せによって，1 [bit] ごとの振舞いが異なるように設定されている．そのため，変換虹彩コードに対して，誤差が生じることのある虹彩コード部分であってもつねに等しい乱数部分であっても，各ビットはともに変化することも変化しないこともある．よって，攻撃者が多数の変換虹彩コードを観測したとしても，虹彩コード部分と乱数部分とを確実に識別することは困難であると考えられる．

式 (2) の θ_j は，0 [deg]，90 [deg]，180 [deg]，270 [deg] のいずれかの角度が乱数に従って選択される．そのため，並べ替え虹彩コードから変換された 2 次元ベクトルは原点中心，半径 1 の円周上の 45 [deg]，135 [deg]，225 [deg]，315 [deg] に位置する座標へほぼ一様に変換され，ある座標にだけ特に多く変換されることはない．

以上の考察と変換関数の各処理が乱数によって決定されることから，変換関数の安全性は生成される乱数の乱数性に依存する部分が大きい．ただ，乱数生成アルゴリズムの安全性は保証されていると仮定しているので，虹彩コードはほぼ一様に変換虹彩コードに変換され，攻撃者が変換虹彩コードの分析により虹彩コードを推測することは困難であると考えられる．

4.2.2 全数探索に対する安全性

本項では，各攻撃対象における全数探索の成功確率がきわめて低いことを示す．

4.2.1 項において考察したように，生成される乱数の安全性により，攻撃者が変換虹彩コードから虹彩コードを推測することは困難である．そのため，攻撃者が利用者のなりすましに成功するために行うべき攻撃は全数探索である．

そこで，各攻撃対象における全数探索の成功確率について考察する．まず，攻撃モデル (1) に対し，攻撃対象と考えられるのは，n [bit] の虹彩コードと $\{n+\log_2 2n+(2\log_2 2n+2)(3n/2)\}$ [bit] の乱数である．n [bit] の虹彩コードにおける全数探索の成功確率 $P_1(n)$ は

$$P_1(n)=2^{-n}$$

で与えられる．また，文献 9) によると，同じ虹彩から得られた 2,048 [bit] の虹彩コード間における正規化ハミング距離の平均は 0.084 と測定されている．つまり，同じ虹彩から得られた n [bit] の虹彩コード間でも，$\lceil 0.084n \rceil$ [bit] 程度の誤差が生じる．この誤差まで許容した場合，n [bit] の虹彩コードにおける全数探索の成功確率 $P_1'(n)$ は

$$P_1'(n)=\frac{{}_n\mathrm{C}_{\lceil 0.084n \rceil}}{2^n}$$

で与えられる．ただし，$\lceil a \rceil$ は a 以上の最小の整数を表している．また，$\{n+\log_2 2n+(2\log_2 2n+2)(3n/2)\}$ [bit] の乱数における全数探索の成功確率 $P_2(n)$ は

$$P_2(n)=2^{-\{n+\log_2 2n+(2\log_2 2n+2)(3n/2)\}}$$

で与えられる．

次に，攻撃モデル (2) に対し，攻撃対象と考えられるのは，$2n$ [bit] の変換虹彩コードである．$2n$ [bit] の変換虹彩コードにおける全数探索の成功確率 $P_3(n)$ は

$$P_3(n)=2^{-2n}$$

で与えられる．また，$P_1'(n)$ と同様にして，同じ虹彩から得られた $2n$ [bit] の変換虹彩コード間でも，$\lceil 0.084n \rceil$ [bit] 程度の誤差が生じるとし，この誤差まで許容した場合，$2n$ [bit] の変換虹彩コードにおける全数探索の成功確率 $P_3'(n)$ は

$$P_3'(n)=\frac{{}_{2n}\mathrm{C}_{\lceil 0.084n \rceil}}{2^{2n}}$$

で与えられる．

そこで，$n=2048, 4096$ のとき，各確率の概算を**表 2**

表 2 各確率の概算
Table 2 The approximate estimates of all probabilities.

確率	$n = 2048$	$n = 4096$
$P_1(n)$	3.09×10^{-617}	9.57×10^{-1234}
$P_1'(n)$	3.43×10^{-361}	2.43×10^{-721}
$P_2(n)$	1.02×10^{-24664}	$< 5.05 \times 10^{-43430}$
$P_3(n)$	9.57×10^{-1234}	9.17×10^{-2467}
$P_3'(n)$	5.65×10^{-924}	1.45×10^{-1831}

に示す．参考として現在安全とされている 1,024 [bit] の RSA 暗号において，素因数分解よりさらに効率の悪い秘密鍵の全数探索の成功確率 P_{RSA} を計算すると，

$$P_{\mathrm{RSA}} = 2^{-1024} \approx 5.56 \times 10^{-309}$$

となる．したがって，$n = 2048, 4096$ に対し，

$$P_1(n), P_1'(n), P_2(n), P_3(n), P_3'(n) < P_{\mathrm{RSA}}$$

が成立する．

ここで，本稿における脅威に関し，サーバの管理者が自由に乱数を選択できることまでを許容し，自らの虹彩コードを用いて他のサーバで利用者になりすます場合を考察する．この場合，サーバの管理者は利用者が他のサーバに送る変換虹彩コードを得る必要がある．しかし，通信ネットワーク上では変換虹彩コードを他のサーバの公開鍵を用いて暗号化しており，乱数は登録するサーバごとに異なることを仮定しているため，サーバの管理者は他のサーバに送られる利用者の変換虹彩コードを得ることはできない．したがって，サーバの管理者が乱数を自由に選ぶことができたとしても，利用者になりすまして他のサーバに認証される確率は $P_3'(n)$ と等価になり，その危険性は無視できるくらい低いと結論づけられる．また，サーバの管理者が利用者自身と結託することを仮定した場合には，他のサーバで利用者になりすますことも可能である．しかし，サーバの管理者と利用者の結託は，利用者の虹彩コードを秘匿する本稿の目的に反するため，仮定していない．

虹彩コードを秘匿したまま虹彩認証を可能にするためには，変換関数の計算前後において，登録虹彩コードと入力虹彩コード間の（正規化）ハミング距離を保存する必要がある．そのため，本提案手法では通常の公開鍵暗号や共通鍵暗号ほどの強度は達成されないが，実用的には十分な強度を有するといえる．したがって，提案方式における変換関数が虹彩コードを秘匿するのに十分な安全性を有することが示された．

5. むすび

本稿では，生体認証の中でも虹彩認証に主眼を置き，虹彩コードに変換関数を施すことにより，虹彩コードを秘匿する手法を提案した．そして，その変換関数を利用することにより，虹彩コードを秘匿したままサーバにおける虹彩認証が可能な方式も提案した．その変換関数とは，拡大処理，並べ替え処理，回転処理の 3 つの処理から構成され，関数全体では非線形性を有する．そのため，変換関数の計算前後において登録虹彩コードと入力虹彩コード間の正規化ハミング距離が保存され，サーバにおける虹彩認証が可能であるにもかかわらず，虹彩コードを秘匿するのに十分な安全性を有することを示した．また，虹彩認証以外の生体認証についても同様にして，本提案方式の変換関数が利用可能であると考えられる．今後の課題としては，本提案方式において，他の攻撃に対するさらなる安全性を評価することなどがあげられる．

参考文献

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，p.174, 共立出版 (2002).
2) 瀬戸洋一：バイオメトリクスを用いた本人認証技術，計測と制御，Vol.37, No.6, pp.395–401 (1998).
3) 瀬戸洋一：IC カードを用いた個人認証技術とその将来展望，映像情報メディア学会誌，Vol.55, No.2, pp.194–198 (2001).
4) Ratha, N., Connell, J. and Bolle, R.: Enhancing security and privacy in biometrics-based authentication systems, *IBM Systems Journal*, Vol.40, No.3, pp.614–634 (2001).
5) 若山公威，出路裕介，冷 基立，岩田 彰：指紋照合によるリモートユーザ認証方式，情報処理学会論文誌，Vol.44, No.2, pp.401–404 (2003).
6) 中村智博，吉浦紀晃，小野里好邦：ハッシュ関数を用いた生体情報による認証の実験，情報処理学会研究報告，Vol.2003, No.18, pp.245–250 (2003).
7) 太田陽基，笹野義二，菅谷史昭：プライバシーを保護する虹彩認証方式の提案，コンピュータセキュリティシンポジウム 2003，Vol.2003, No.15, pp.163–168 (2003).
8) Jain, A., Bolle, R. and Pankanti, S.: Introduction to Biometrics, *BIOMETRICS: Personal Identification in Networked Society*, Jain, A., Bolle, R. and Pankanti, S. (Eds.), pp.1–41, Kluwer Academic Publishers (2002).
9) Daugman, J.: High Confidence Visual Recognition of Persons by a Test of Statistical Independence, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.15, No.11, pp.1148–1161 (1993).
10) Zhang, D.: *AUTOMATED BIOMETRICS: Technologies and Systems*, p.331, Kluwer Aca-

demic Publishers (2000).
11) Daugman, J.: Recognizing Persons by their Iris Patterns, *BIOMETRICS: Personal Identification in Networked Society*, Jain, A., Bolle, R. and Pankanti, S. (Eds.), pp.103–121, Kluwer Academic Publishers (2002).

(平成 15 年 12 月 4 日受付)
(平成 16 年 6 月 8 日採録)

太田 陽基（正会員）

1976 年生．2002 年東京工業大学大学院理工学研究科集積システム専攻修士課程修了．同年 KDDI（株）に入社．現在，（株）KDDI 研究所セキュリティグループ研究員．セキュリティ技術の研究に従事．電子情報通信学会会員．

清本 晋作（正会員）

1975 年生．2000 年筑波大学大学院工学研究科物質工学専攻前期博士課程修了．同年 KDD（株）入社．現在，（株）KDDI 研究所セキュリティグループ研究員．ストリーム暗号，暗号プロトコル，モバイルセキュリティの研究に従事．日本物理学会，電子情報通信学会各会員．2004 年電子情報通信学会学術奨励賞受賞．

田中 俊昭（正会員）

1960 年生．1986 年大阪大学大学院工学研究科通信工学専攻前期博士課程修了．同年 KDD（株）入社．現在，（株）KDDI 研究所セキュリティグループリーダー．暗号プロトコル，著作権保護，モバイルセキュリティ，次世代 IDS の研究に従事．電子情報通信学会会員．