(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2018-207433
(P2018-207433A)
(43)公開日　平成30年12月27日(2018.12.27)

(51)Int.Cl.			F I			テーマコード（参考）
H04L	9/32	(2006.01)	H04L	9/00	673D	5J104
H04L	9/08	(2006.01)	H04L	9/00	601A	
G06F	21/32	(2013.01)	H04L	9/00	601E	
			G06F	21/32		

審査請求 未請求　請求項の数 12　OL　（全 22 頁）

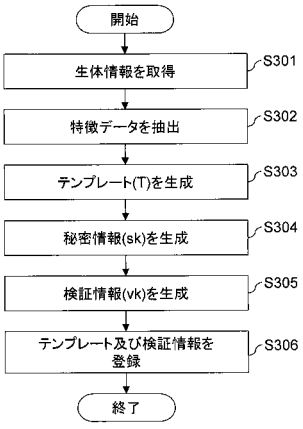
(21)出願番号　特願2017-114026 (P2017-114026)
(22)出願日　平成29年6月9日 (2017.6.9)

(71)出願人　000005108
株式会社日立製作所
東京都千代田区丸の内一丁目６番６号
(74)代理人　110001678
特許業務法人藤央特許事務所
(72)発明者　高橋　健太
東京都千代田区丸の内一丁目６番６号　株式会社日立製作所内

Fターム(参考)　5J104 AA16　EA20　KA01　KA16　NA02
NA37　PA07

(54)【発明の名称】計算機システム、秘密情報の検証方法、及び計算機

(57)【要約】　（修正有）
【課題】効率性及び安全性を両立したバイオメトリック暗号の技術を実現するシステムを提供する。
【解決手段】計算機を含む計算機システムであって、計算機は、ユーザから取得した生体情報に基づいて、特徴データを生成し、特徴データに基づいて、生体情報に発生する誤差を示す誤差特徴データ及び生体情報に発生する誤差以外の部分を示す定常特徴データを生成し、生体情報の誤差特徴データに基づいてテンプレートを生成し、生体情報の定常特徴データに基づいて、暗号学的処理に用いる第１秘密情報を生成し、前記第１秘密情報に基づいて第１検証情報を生成する。
【選択図】図３



【特許請求の範囲】
【請求項１】
少なくとも一つの計算機を含む計算機システムであって、
前記少なくとも一つの計算機は、演算装置及び前記演算装置に接続される記憶装置を有し、
前記演算装置は、
ユーザから取得された第１生体情報に基づいて第１特徴データを生成し、
前記第１特徴データに基づいて、前記第１生体情報に発生する誤差を示す誤差特徴データ及び前記第１生体情報に発生する誤差以外の部分を示す定常特徴データを生成し、
前記第１生体情報の誤差特徴データに基づいてテンプレートを生成し、前記ユーザの識別情報及び前記テンプレートを対応付けて前記記憶装置に格納し、
前記第１生体情報の定常特徴データに基づいて、暗号学的処理に用いる第１秘密情報を生成し、前記第１秘密情報に基づいて第１検証情報を生成し、前記ユーザの識別情報及び前記第１検証情報を対応付けて前記記憶装置に格納し、
前記暗号学的処理の実行要求を受け付けた場合、ユーザから取得された第２生体情報に基づいて第２特徴データを生成し、
前記テンプレート及び前記第２特徴データに基づいて、第２秘密情報を生成し、
前記第２秘密情報に基づいて、第２検証情報を生成し、
前記第１検証情報及び前記第２検証情報を比較することによって、前記第２秘密情報の検証を行い、
前記第２秘密情報の検証の結果に基づいて、前記暗号学的処理を実行することを特徴とする計算機システム。
【請求項２】
請求項１に記載の計算機システムであって、
前記第１特徴データ及び前記第２特徴データは、実数を要素とする特徴ベクトルであり、
前記誤差特徴データは、前記特徴ベクトルの各要素の小数部分を要素とするベクトルであり、
前記定常特徴データは、前記特徴ベクトルの各要素の整数部分を要素とするベクトルであることを特徴とする計算機システム。
【請求項３】
請求項２に記載の計算機システムであって、
前記演算装置は、
前記誤差特徴データの各要素の値を丸める丸め処理を実行し、
前記丸め処理が実行された前記第１生体情報の誤差特徴データを前記テンプレートとして生成することを特徴とする計算機システム。
【請求項４】
請求項２に記載の計算機システムであって、
前記演算装置は、
任意の長さのデータ列であるソルトを生成し、
前記丸め処理が実行された第１生体情報の誤差特徴データ及び前記ソルトの組を前記テンプレートとして生成することを特徴とする計算機システム。
【請求項５】
請求項２に記載の計算機システムであって、
前記演算装置は、
ハッシュ関数、鍵付きハッシュ関数、及び暗号化関数の少なくともいずれかに、前記第１生体情報の定常特徴データを入力することによって前記第１秘密情報を生成し、
前記第２特徴データ及び前記テンプレートに含まれる前記第１生体情報の誤差特徴データに基づいて、前記第２生体情報の定常特徴データを生成し、
前記ハッシュ関数、前記鍵付きハッシュ関数、及び前記暗号化関数の少なくともいずれ

かに、前記第２生体情報の定常特徴データを入力することによって前記第２秘密情報を生成することを特徴とする計算機システム。
【請求項６】
秘密情報を用いて暗号学的処理を実行する計算機システムにおける秘密情報の検証方法であって、
前記計算機システムは、演算装置及び前記演算装置に接続される記憶装置を有する少なくとも一つの計算機を含み、
前記秘密情報の検証方法は、
前記演算装置が、ユーザから取得された第１生体情報に基づいて第１特徴データを生成する第１のステップと、
前記演算装置が、前記第１特徴データに基づいて、前記第１生体情報に発生する誤差を示す誤差特徴データ及び前記第１生体情報に発生する誤差以外の部分を示す定常特徴データを生成する第２のステップと、
前記演算装置が、前記第１生体情報の誤差特徴データに基づいてテンプレートを生成し、前記ユーザの識別情報及び前記テンプレートを対応付けて前記記憶装置に格納する第３のステップと、
前記演算装置が、前記第１生体情報の定常特徴データに基づいて、前記暗号学的処理に用いる第１秘密情報を生成し、前記第１秘密情報に基づいて第１検証情報を生成し、前記ユーザの識別情報及び前記第１検証情報を対応付けて前記記憶装置に格納する第４のステップと、
前記演算装置が、前記暗号学的処理の実行要求を受け付けた場合、ユーザから取得された第２生体情報に基づいて第２特徴データを生成する第５のステップと、
前記演算装置が、前記テンプレート及び前記第２特徴データに基づいて、第２秘密情報を生成する第６のステップと、
前記演算装置が、前記第２秘密情報に基づいて、第２検証情報を生成する第７のステップと、
前記演算装置が、前記第１検証情報及び前記第２検証情報を比較することによって、前記第２秘密情報の検証を行う第８のステップと、
前記演算装置が、前記第２秘密情報の検証の結果に基づいて、前記暗号学的処理を実行する第９のステップとことを特徴とする秘密情報の検証方法。
【請求項７】
請求項６に記載の秘密情報の検証方法であって、
前記第１特徴データ及び前記第２特徴データは、実数を要素とする特徴ベクトルであり、
前記誤差特徴データは、前記特徴ベクトルの各要素の小数部分を要素とするベクトルであり、
前記定常特徴データは、前記特徴ベクトルの各要素の整数部分を要素とするベクトルであることを特徴とする秘密情報の検証方法。
【請求項８】
請求項７に記載の秘密情報の検証方法であって、
前記第３のステップは、
前記演算装置が、前記誤差特徴データの各要素の値を丸める丸め処理を実行するステップと、
前記演算装置が、前記丸め処理が実行された前記第１生体情報の誤差特徴データを前記テンプレートとして生成するステップと、を含むことを特徴とする秘密情報の検証方法。
【請求項９】
請求項７に記載の秘密情報の検証方法であって、
前記第３のステップは、
前記演算装置が、任意の長さのデータ列であるソルトを生成するステップと、
前記演算装置が、前記丸め処理が実行された第１生体情報の誤差特徴データ及び前記ソ

ルトの組を前記テンプレートとして生成するステップと、を含むことを特徴とする秘密情報の検証方法。
【請求項１０】
請求項７に記載の秘密情報の検証方法であって、
前記第４のステップは、前記演算装置が、ハッシュ関数、鍵付きハッシュ関数、及び暗号化関数の少なくともいずれかに、前記第１生体情報の定常特徴データを入力することによって前記第１秘密情報を生成するステップを含み、
前記第６のステップは、
前記演算装置が、前記第２特徴データ及び前記テンプレートに含まれる前記第１生体情報の誤差特徴データに基づいて、前記第２生体情報の定常特徴データを生成するステップと、
前記演算装置が、前記ハッシュ関数、前記鍵付きハッシュ関数、及び前記暗号化関数の少なくともいずれかに、前記第２生体情報の定常特徴データを入力することによって前記第２秘密情報を生成するステップと、を含むことを特徴とする秘密情報の検証方法。
【請求項１１】
データの秘匿に用いる秘密情報の生成に用いるテンプレートを生成する計算機であって、
前記計算機は、演算装置及び前記演算装置に接続される記憶装置を有し、
前記演算装置は、
ユーザから取得された第１生体情報に基づいて第１特徴データを生成し、
前記第１特徴データに基づいて、前記第１生体情報に発生する誤差を示す誤差特徴データ及び前記第１生体情報に発生する誤差以外の部分を示す定常特徴データを生成し、
前記誤差特徴データの各要素の値を丸める丸め処理を実行し、
任意の長さのデータ列であるソルトを生成し、
前記丸め処理が実行された前記第１生体情報の誤差特徴データ及び前記ソルトに基づいて、検証対象の第２秘密情報の生成に用いるテンプレートを生成し、前記ユーザの識別情報及び前記テンプレートを対応付けて前記記憶装置に格納し、
前記第１生体情報の定常特徴データに基づいて第１秘密情報を生成し、前記第１秘密情報に基づいて、前記第２秘密情報の検証に用いる検証情報を生成し、前記ユーザの識別情報及び前記検証情報を対応付けて前記記憶装置に格納することを特徴とする計算機。
【請求項１２】
請求項１１に記載の計算機であって、
前記第１特徴データは、実数を要素とする特徴ベクトルであり、
前記誤差特徴データは、前記特徴ベクトルの各要素の小数部分を要素とするベクトルであり、
前記定常特徴データは、前記特徴ベクトルの各要素の整数部分を要素とするベクトルであることを特徴とする計算機。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、ユーザの生体情報に基づいて、認証、暗号、及び署名などの処理を行うシステムに関する。
【背景技術】
【０００２】
指紋、静脈、顔、及び虹彩などの生体情報に基づいて個人の認証などを行う生体認証技術が広く利用されている。従来の生体認証技術では、以下のような処理が行われる。まず、ユーザ登録時には、端末は、ユーザの生体情報から抽出した特徴データをテンプレートとしてシステムに登録する。ユーザ認証時には、端末は、テンプレートと、再度ユーザの生体情報から抽出した特徴データとを比較して、類似度が十分大きい場合、すなわち、二つの特徴データの距離が十分近い場合、認証の成功と判定し、類似度が小さい場合には認

証の失敗と判定する。

【０００３】

生体情報は、一般には取り替えることのできない情報である。そのため、生体情報の漏えいは大きな問題となる。この問題に対し、生体情報を秘匿したまま認証する、テンプレート保護型の生体認証技術が研究開発されている。その中で、生体情報から鍵データを生成し、暗号学的な認証処理、暗号化処理、復号化処理、及び署名生成処理などの処理を行う、バイオメトリック暗号と呼ばれる技術が注目されている。

【０００４】

バイオメトリック暗号では、端末は、生体情報の登録時に、生体情報の特徴データＸを変換し、秘密鍵Ｋを埋め込むことによって、保護テンプレートＴを生成する。その後、端末は、新たに取得した生体情報の特徴データＸ’及び保護テンプレートＴを用いて秘密鍵Ｋを復元する。秘密鍵の復元が成功した場合、端末は、当該秘密鍵Ｋを用いて暗号学的な認証処理、暗号化処理、復号化処理、及び電子署名生成処理を実行できる。

【０００５】

安全性の要件から、バイオメトリック暗号において、保護テンプレートＴを用いた特徴データＸの復元又は推定は十分困難でなくてはならない。一方、特徴データＸ’が特徴データＸに十分類似している場合、秘密鍵の復元処理が成功するようにしなければならない。

【０００６】

バイオメトリック暗号の具体的な実現方法として、例えば、非特許文献１に記載されている方法が提案されている。非特許文献１では、生体情報から抽出された特徴データは、式（１）に示すようなｎ次元実数ベクトルＸであって、各要素とも±δまでの誤差が許される。すなわち、値の誤差が±δの範囲であれば本人として受理される。また、秘密鍵は式（２）に示すような整数ベクトルとして表現できるものとする。秘密鍵の各要素ｓ＿ｉはそれぞれｑビット整数とする。ここで、添字ｉは１からｎまでの値であり、ｑは任意の整数である。したがって、要素ｓ＿ｉは０以上、かつ、２^{q}－１以下の整数である。

【０００７】

【数１】

$$X = (x\_1, \ldots, x\_n) \ldots (1)$$

【０００８】

【数２】

$$S = (s\_1, \ldots, s\_n) \ldots (2)$$

【０００９】

生体情報の登録時のテンプレートＴは式（３）に示すようなベクトルとして表現でき、要素ｔ＿ｉは式（４）に基づいて算出される。ここで、添字ｉは１からｎまでの値である。

【００１０】

【数３】

$$T = (t\_1, \ldots, t\_n) \ldots (3)$$

【００１１】

【数４】

$$t\_i = x\_i - \delta - 2\delta s\_i \ldots (4)$$

【００１２】

新たに取得した生体情報の特徴データＸ’は式（５）に示すようなベクトルとして表現できる。

【００１３】

【数５】

$$X' = (x'\_1, \ldots, x'\_n) \quad \ldots (5)$$

【００１４】

秘密鍵の検証時には、端末は、保存されたテンプレートＴ及び特徴データＸ’を用いて、式（７）に示す演算を実行することによって式（６）で表現される秘密鍵Ｓ’の各要素の値を算出する。これによって、秘密鍵Ｓ’が復元される。ここで、添字ｉは１からｎまでの値である。また、記号「［　］」は、括弧内の値の小数部分を切り捨てて、整数部分を取り出す演算を表す。

【００１５】

【数６】

$$S = (s'\_1, \ldots, s'\_n) \quad \ldots (6)$$

【００１６】

【数７】

$$s'\_i = \left[\frac{(x'\_i - t\_i)}{2\delta} \right] \quad \ldots (7)$$

【００１７】

ここで、式（８）を満たす場合、復元された秘密鍵Ｓ’は秘密鍵Ｓと一致するため、端末は、秘密鍵が正しく復元されたものと判定し、秘密鍵の検証を行ったユーザを、システムに登録されたユーザ本人として受理する。

【００１８】

【数８】

$$|x'\_i - x\_i| < \delta \quad \ldots (8)$$

【００１９】

非特許文献１で用いられるテンプレートＴを用いて特徴データＸ及び秘密鍵Ｓを一意に求めることはできないため、非特許文献１の方法を用いることによって、一定の効果を有するテンプレートの保護を実現できる。

【先行技術文献】

【非特許文献】

【００２０】

【非特許文献１】Gang Zheng, et.al., “Cryptographic Key Generation from Biometric Data Using Lattice Mapping”, In 18th International Conference on Pattern Recognition (ICPR'06), 2006.

【発明の概要】

【発明が解決しようとする課題】

【００２１】

しかし、非特許文献１の方法では、安全性及び効率性に課題がある。

【００２２】

安全性に関して、テンプレートＴを用いて特徴データＸ及び秘密鍵Ｓの候補を絞り込むことができるという問題がある。具体的には、秘密鍵Ｓの要素ｓ＿ｉがｑビットの整数であるという制約から、ｔ＿ｉを知っている攻撃者は、式（４）に基づいて特徴データＸの要素ｘ＿ｉの候補を式（９）に示す範囲に絞り込むことができる。

【００２３】

【数９】

$$t\_i + \delta \leq x\_i \leq t\_i + 2(2^q - 1) \quad \ldots (9)$$

【００２４】

さらに、特徴データの要素ｘ＿ｉが取り得る値は式（１０）に示す範囲であり、攻撃者が当該範囲を知っている場合、要素ｘ＿ｉの候補は式（９）及び式（１０）の共通範囲に絞り込むことができる。

【００２５】

【数１０】

$$x\_min \leq x\_i \leq x\_max \ldots (10)$$

【００２６】

例えば、要素ｘ＿ｉ＝ｘ＿ｍａｘ、かつ、要素ｓ＿ｉ＝０である場合、式（４）より要素ｔ＿ｉは式（１１）に示すような値となる。そのため、テンプレートＴを知っている攻撃者は、式（９）から要素ｘ＿ｉが式（１２）を満たすことが分かる。

【００２７】

【数１１】

$$t\_i = x\_i - \delta \ldots (11)$$

【００２８】

【数１２】

$$x\_max \leq x\_i \ldots (12)$$

【００２９】

したがって、式（１０）及び式（１２）から、要素ｘ＿ｉの範囲は式（１３）に示すようになるため、攻撃者は、要素ｘ＿ｉ＝ｘ＿ｍａｘとなることが分かる。また、攻撃者は、式（４）から要素ｓ＿ｉ＝０も分かる。

【００３０】

【数１３】

$$x\_max \leq x\_i \leq x\_max \ldots (13)$$

【００３１】

前述の仮定以外の場合においても、要素ｘ＿ｉの範囲が確率的に絞り込まれる可能性がある。したがって、十分な安全性を確保するためには、｜ｘ＿ｍｉｎ｜及び｜ｘ＿ｍａｘ｜に対して２^{q}が十分大きくなるように、ｑの値を大きくし、また、秘密鍵Ｓの要素ｓ＿ｉを、０以上かつ２^{q}－１以下の範囲で一様ランダムに生成する必要である。

【００３２】

しかし、ｑを大きくした場合、テンプレートＴのサイズも増大し、効率性の課題が生じる。具体的には、特徴データＸの要素ｘ＿ｉの小数部分の表現桁数が２進数表示でｒ桁とした場合、テンプレートＴの要素ｔ＿ｉは、整数部分がｑビットとなり、小数部分がｒビットとなる。したがって、テンプレートＴのデータサイズは、ｎ（ｑ＋ｒ）ビットとなる。

【００３３】

ｑ＝２５６、ｒ＝５２（ｄｏｕｂｌｅの仮数部桁数）、ｎ＝１０００とした場合、テンプレートＴのデータサイズは３８．５ＫＢとなる。一本の指の指紋から一つのテンプレートＴを生成する場合、一ユーザに対して１０個のテンプレートＴが生成される。１億人分のテンプレートＴがシステムに登録された場合、データサイズは合計で３８．５ＴＢとなる。認証用のデータと、登録されたＮ個の全てのテンプレートとの照合を行う認証方法である１：ｎ認証を実現しようとした場合、３８．５ＴＢのデータにアクセスする必要があるため、認証処理において、当該データを格納する記憶領域へのアクセス時間が支配的となる。

【００３４】

本発明は、従来のバイオメトリック暗号技術の課題である安全性及び効率性を向上することを目的とする。具体的には、特徴データの推定が困難であり、かつ、サイズが小さい

テンプレートを生成することを目的とする。
【課題を解決するための手段】
【００３５】
本願において開示される発明の代表的な一例を示せば以下の通りである。すなわち、少なくとも一つの計算機を含む計算機システムであって、前記少なくとも一つの計算機は、演算装置及び前記演算装置に接続される記憶装置を有し、前記演算装置は、ユーザから取得された第１生体情報に基づいて第１特徴データを生成し、前記第１特徴データに基づいて、前記第１生体情報に発生する誤差を示す誤差特徴データ及び前記第１生体情報に発生する誤差以外の部分を示す定常特徴データを生成し、前記第１生体情報の誤差特徴データに基づいてテンプレートを生成し、前記ユーザの識別情報及び前記テンプレートを対応付けて前記記憶装置に格納し、前記第１生体情報の定常特徴データに基づいて、暗号学的処理に用いる第１秘密情報を生成し、前記第１秘密情報に基づいて第１検証情報を生成し、前記ユーザの識別情報及び前記第１検証情報を対応付けて前記記憶装置に格納し、前記暗号学的処理の実行要求を受け付けた場合、ユーザから取得された第２生体情報に基づいて第２特徴データを生成し、前記テンプレート及び前記第２特徴データに基づいて、第２秘密情報を生成し、前記第２秘密情報に基づいて、第２検証情報を生成し、前記第１検証情報及び前記第２検証情報を比較することによって、前記第２秘密情報の検証を行い、前記第２秘密情報の検証の結果に基づいて、前記暗号学的処理を実行することを特徴とする。
【発明の効果】
【００３６】
本発明によれば、テンプレートのサイズを大幅に削減し、かつ、特徴データの推定が困難なバイオメトリック暗号技術を実現できる。上記した以外の課題、構成及び効果は、以下の実施例の説明により明らかにされる。
【図面の簡単な説明】
【００３７】
【図１】実施例１の生体認証システムの構成例を示す図である。
【図２】実施例１の生体認証システムを構成する計算機のハードウェア構成の一例を示す図である。
【図３】実施例１の登録端末が実行する登録処理を説明するフローチャートである。
【図４】実施例１の認証端末が実行する検証処理を説明するフローチャートである。
【図５】実施例１のテンプレートＴの生成処理及び秘密情報ｓｋの生成処理を説明するフローチャートである。
【図６】実施例１の秘密情報ｓｋ’の復元処理を説明するフローチャートである。
【発明を実施するための形態】
【００３８】
以下、本発明に係る実施例を添付図面を用いて説明する。各図において共通の構成については同一の参照符号が付されている。
【実施例１】
【００３９】
実施例１の生体認証システムでは、まず、生体認証システムに含まれる計算機が、ユーザの生体情報からテンプレート及び復元された秘密情報（秘密鍵）を検証するための検証情報を生成し、データベースに登録する。計算機は、暗号学的処理の実行に伴う秘密情報の検証を行う場合、登録されたテンプレート及び新たに取得したユーザの生体情報を用いて秘密情報を復元する。また、計算機は、復元された秘密情報及び検証情報に基づいて、秘密情報の検証を行い、検証結果に基づいてユーザ認証処理、暗号化処理、復号化処理、及び電子署名の生成処理などの暗号学的処理を実行する。
【００４０】
図１は、実施例１の生体認証システムの構成例を示す図である。図２は、実施例１の生体認証システムを構成する計算機のハードウェア構成の一例を示す図である。
【００４１】

生体認証システムは、登録端末１００、認証端末１１０、ＤＢサーバ１２０、及びネットワーク１３０から構成される。登録端末１００、認証端末１１０、及びＤＢサーバ１２０は、ネットワーク１３０を介して互いに接続される。ネットワーク１３０は、ＬＡＮ（Ｌｏｃａｌ　Ａｒｅａ　Ｎｅｔｗｏｒｋ）及びＷＡＮ（Ｗｉｄｅ　Ａｒｅａ　Ｎｅｔｗｏｒｋ）等が考えられる。なお、本実施例はネットワーク１３０の種別に限定されない。また、ネットワーク１３０の接続方式は有線及び無線のいずれでもよい。

【００４２】

登録端末１００は、ユーザから生体情報を取得し、生体情報を用いてテンプレート及び検証情報を生成し、また、テンプレート及び検証情報をＤＢサーバ１２０に登録する。

【００４３】

ここで、登録端末１００のハードウェア構成について説明する。図２に示すように登録端末１００は、ＣＰＵ２００、メモリ２０１、記憶装置２０２、入力装置２０３、出力装置２０４、及び通信装置２０５を有する。なお、後述する認証端末１１０及びＤＢサーバ１２０も同一のハードウェア構成である。

【００４４】

ＣＰＵ２００は、登録端末１００の演算装置であり、メモリ２０１に格納されるプログラムを実行する。ＣＰＵ２００がプログラムにしたがって処理を実行することによって、特定の機能を実現するモジュールとして動作する。以下の説明では、モジュールを主語に処理を説明する場合、ＣＰＵ２００が当該モジュールを実現するプログラムを実行していることを示す。

【００４５】

メモリ２０１は、登録端末１００の主記憶装置であり、ＣＰＵ２００が実行するプログラム及びプログラムが使用するデータを格納する。また、メモリ２０１は、プログラムが一時的に使用する一時領域を含む。

【００４６】

記憶装置２０２は、登録端末１００の副記憶装置であり、データを永続的に格納する。記憶装置２０２は、例えば、ＨＤＤ（Ｈａｒｄ　Ｄｉｓｋ　Ｄｒｉｖｅ）及びＳＳＤ（Ｓｏｌｉｄ　Ｓｔａｔｅ　Ｄｒｉｖｅ）等が考えられる。

【００４７】

入力装置２０３は、登録端末１００に各種データを入力するための装置であり、キーボード、マウス、タッチパネル、及びセンサなどを含む。

【００４８】

出力装置２０４は、各種情報を出力するための装置であり、タッチパネル及びディスプレイなどを含む。

【００４９】

通信装置２０５は、ネットワークを介して多の装置と通信するためのインタフェースである。

【００５０】

ここで、登録端末１００の機能構成について説明する。登録端末１００のメモリ２０１は、データ取得モジュール１０１、特徴データ抽出モジュール１０２、テンプレート生成モジュール１０３、秘密情報生成モジュール１０４、及び検証情報生成モジュール１０５を実現するプログラムを格納する。

【００５１】

データ取得モジュール１０１は、ユーザから指紋及び静脈などの登録用の生体情報を取得し、取得した登録用の生体情報を特徴データ抽出モジュール１０２に出力する。

【００５２】

特徴データ抽出モジュール１０２は、登録用の生体情報から登録用の特徴データを抽出し、抽出した登録用の特徴データをテンプレート生成モジュール１０３及び秘密情報生成モジュール１０４に出力する。

【００５３】

テンプレート生成モジュール１０３は、登録用の特徴データに基づいてテンプレートを生成する。テンプレート生成モジュール１０３は、テンプレートをＤＢサーバ１２０に送信することによって、当該テンプレートをテンプレートＤＢ１２２に登録する。

【００５４】

秘密情報生成モジュール１０４は、登録用の特徴データに基づいて秘密情報を生成する。秘密情報生成モジュール１０４は、秘密情報を検証情報生成モジュール１０５に出力する。

【００５５】

検証情報生成モジュール１０５は、秘密情報に基づいて認証端末１１０が復元した秘密情報を検証するための検証情報を生成する。検証情報生成モジュール１０５は、検証情報をＤＢサーバ１２０に送信することによって、当該検証情報を検証情報ＤＢ１２３に登録する。

【００５６】

認証端末１１０は、ユーザから新たに生体情報を取得し、新たに取得した生体情報とテンプレートに基づいて秘密情報を復元し、復元された秘密情報の検証を行う。また、認証端末１１０は、復元された秘密情報の検証結果に基づいて、ユーザ認証処理、暗号化処理、復号化処理、及び電子署名の生成処理などの暗号学的処理を実行する。

【００５７】

ここで、認証端末１１０の機能構成について説明する。認証端末１１０のメモリ２０１は、データ取得モジュール１１１、特徴データ抽出モジュール１１２、秘密情報復元モジュール１１３、秘密情報検証モジュール１１４、及びデータ処理モジュール１１５を実現するプログラムを格納する。

【００５８】

データ取得モジュール１１１は、ユーザから秘密情報を生成するための生体情報を取得し、取得した生体情報を特徴データ抽出モジュール１１２に出力する。

【００５９】

特徴データ抽出モジュール１１２は、生体情報から特徴データを抽出し、抽出した特徴データを秘密情報復元モジュール１１３に出力する。

【００６０】

秘密情報復元モジュール１１３は、特徴データ及びテンプレートに基づいて秘密情報を復元し、復元された秘密情報を秘密情報検証モジュール１１４に出力する。

【００６１】

秘密情報検証モジュール１１４は、検証情報に基づいて復元された秘密情報の正しさを検証する。すなわち、登録端末１００によって生成された秘密情報と、復元された秘密情報とが一致するか否かが判定される。

【００６２】

データ処理モジュール１１５は、秘密情報を用いて、認証処理、暗号化処理、復号化処理、及び電子署名生成処理などの暗号学的処理を実行する。

【００６３】

ＤＢサーバ１２０は、生体認証システムにおいて使用される各種データを管理する。また、ＤＢサーバ１２０は、データの登録処理及びデータの検索処理などを実行する。

【００６４】

ここで、ＤＢサーバ１２０の機能構成について説明する。ＤＢサーバ１２０のメモリ２０１は、データベース管理モジュール１２１を実現するプログラムを格納し、また、テンプレートＤＢ１２２及び検証情報ＤＢ１２３を格納する。なお、テンプレートＤＢ１２２及び検証情報ＤＢ１２３は、ＤＢサーバ１２０の記憶装置２０２に格納されてもよい。

【００６５】

データベース管理モジュール１２１は、データの登録、更新、及び検索を行う。テンプレートＤＢ１２２は、テンプレートを格納するデータベースである。テンプレートＤＢ１２２には、ユーザの識別情報及びテンプレートを対応付けたデータが一つ以上格納される

。例えば、ユーザの識別情報を格納するフィールド及びテンプレートを格納するフィールドから構成されるエントリを一つ以上含むテーブル形式のデータベースが考えられる。

【００６６】

検証情報ＤＢ１２３は、検証情報を格納するデータベースである。検証情報ＤＢ１２３には、ユーザの識別情報及び検証情報を対応付けたデータが一つ以上格納される。例えば、ユーザの識別情報を格納するフィールド及び検証情報を格納するフィールドから構成されるエントリを一つ以上含むテーブル形式のデータベースが考えられる。

【００６７】

本実施例では、一つのＤＢサーバ１２０がテンプレートＤＢ１２２及び検証情報ＤＢ１２３を保持するが、複数のＤＢサーバ１２０から構成される分散データベースを用いて管理してもよい。この場合、各ＤＢサーバ１２０にテンプレート及び検証情報が分散して格納される。テンプレート及び検証情報を分散して管理することによって、情報の漏えいリスクが減少するため、安全性を高めることができる。

【００６８】

なお、テンプレートＤＢ１２２及び検証情報ＤＢ１２３の少なくともいずれかを認証端末１１０が保持してもよい。また、ＩＣカード、ＵＳＢメモリ、データをＱＲコード（登録商標）に変換した印刷物などの可搬媒体、又は、スマートフォンなどの個人端末に、テンプレートＤＢ１２２及び検証情報ＤＢ１２３の少なくともいずれかを格納してもよい。

【００６９】

本実施例では、登録端末１００、認証端末１１０、及びＤＢサーバ１２０を物理的に独立した計算機として記載しているが、本発明はこれに限定されない。一つの計算機に複数の機能を統合してもよい。例えば、認証端末１１０及びＤＢサーバ１２０を一つの計算機を用いて実現してもよい。

【００７０】

なお、登録端末１００、認証端末１１０、及びＤＢサーバ１２０の各々が有する各モジュールについては、二つ以上のモジュールを一つのモジュールにまとめてもよいし、一つのモジュールを機能毎に複数のモジュールに分けてもよい。

【００７１】

次に、実施例１の登録処理の詳細を図３を用いて説明する。図３は、実施例１の登録端末１００が実行する登録処理を説明するフローチャートである。

【００７２】

登録端末１００は、ユーザ又はオペレータの操作を受け付けた場合、以下で説明する登録処理を開始する。まず、登録端末１００のデータ取得モジュール１０１は、入力装置２０３を用いてユーザから登録用の生体情報を取得する（ステップＳ３０１）。なお、データ取得モジュール１０１は、登録処理の開始時又は登録用の生体情報の取得時に、ユーザのＩＤ及び氏名など、ユーザの識別情報も取得する。

【００７３】

次に、登録端末１００の特徴データ抽出モジュール１０２は、登録用の生体情報から登録用の特徴データを抽出する（ステップＳ３０２）。例えば、画像及び特徴ベクトルが特徴データとして抽出される。

【００７４】

次に、登録端末１００のテンプレート生成モジュール１０３は、登録用の特徴データに基づいてテンプレートＴを生成する（ステップＳ３０３）。テンプレートＴの生成処理の詳細については後述する。

【００７５】

次に、登録端末１００の秘密情報生成モジュール１０４は、登録用の特徴データに基づいて秘密情報ｓｋを生成する（ステップＳ３０４）。秘密情報ｓｋの生成処理の詳細については後述する。

【００７６】

次に、登録端末１００の検証情報生成モジュール１０５は、秘密情報ｓｋに基づいて検

証情報ｖｋを生成する（ステップＳ３０５）。ここで、検証情報ｖｋの生成方法の一例について説明する。秘密情報ｓｋに基づいて検証情報ｖｋを生成する方法として以下の三つの方法が考えられる。

【００７７】

（生成方法１）検証情報生成モジュール１０５は、式（１４）に示すように、任意の一方向性関数Ｈａｓｈ（）を用いて検証情報ｖｋを生成する。一方向性関数Ｈａｓｈ（）は、例えば、ＳＨＡ２５６及びＳＨＡ３等の暗号学的ハッシュ関数が考えられる。

【００７８】

【数１４】

$$vk = Hash(sk) \ldots (14)$$

【００７９】

（生成方法２）検証情報生成モジュール１０５は、式（１５）に示すように、巡回群Ｇ＝＜ｇ＞及び秘密情報ｓｋの集合から整数集合への写像φを用いて検証情報ｖｋを生成する。ここでｇは、Ｇの生成元である。

【００８０】

【数１５】

$$vk = g^{\wedge}\varphi(sk) \ldots (15)$$

【００８１】

生成された組（ｓｋ，ｖｋ）は、ＥｌＧａｍａｌ暗号／署名、ＤＳＡ、Ｓｃｈｎｏｒｒ署名、又はそれらの楕円曲線版アルゴリズムなど、多くの公開鍵暗号／電子署名アルゴリズムにおける秘密鍵及び公開鍵の組として扱うことができる。

【００８２】

（生成方法３）検証情報生成モジュール１０５は、式（１６）に示すように、検証情報を生成するための秘密鍵又はパラメータであるｐを変数とする関数Ｅｎｃ（）を用いて検証情報ｖｋを生成する。なお、ｐは、ユーザ単位又は登録用の生体情報単位に設定してもよいし、認証端末１１０単位又は生体認証システム単位に設定してもよい。

【００８３】

【数１６】

$$vk = Enc(sk, p) \ldots (16)$$

【００８４】

関数Ｅｎｃ（）は、ＡＥＳ又はＲＳＡなどにおける暗号化関数、及び鍵付きハッシュ関数などである。以上がステップＳ３０５の処理の説明である。

【００８５】

次に、登録端末１００のテンプレート生成モジュール１０３及び検証情報生成モジュール１０５は、それぞれ、テンプレートＴ及び検証情報ｖｋをＤＢサーバ１２０に登録する（ステップＳ３０６）。その後、登録端末１００は、登録処理を終了する。

【００８６】

具体的には、テンプレート生成モジュール１０３は、ユーザの識別情報及びテンプレートＴを含む登録要求をＤＢサーバ１２０に送信し、検証情報生成モジュール１０５は、ユーザの識別情報及び検証情報ｖｋを含む登録要求をＤＢサーバ１２０に送信する。

【００８７】

ＤＢサーバ１２０は、テンプレート生成モジュール１０３から登録要求を受信した場合、ユーザの識別情報及びテンプレートＴを対応付けたデータをテンプレートＤＢ１２２に登録する。また、ＤＢサーバ１２０は、検証情報生成モジュール１０５から登録要求を受信した場合、ユーザの識別情報及び検証情報ｖｋを対応付けたデータを検証情報ＤＢ１２３に登録する。

【００８８】

次に、実施例１における検証処理の詳細を図４を用いて説明する。図４は、実施例１の認証端末１１０が実行する検証処理を説明するフローチャートである。

【００８９】

認証端末１１０は、ユーザ又はオペレータの操作を受け付けた場合、以下で説明する認証処理を開始する。まず、認証端末１１０のデータ取得モジュール１１１は、入力装置２０３を用いてユーザから検証用の生体情報を取得する（ステップＳ４０１）。なお、データ取得モジュール１１１は、検証処理の開始時又は検証用の生体情報の取得時に、ユーザのＩＤ及び氏名など、ユーザの識別情報も取得する。

【００９０】

次に、認証端末１１０の特徴データ抽出モジュール１１２は、検証用の生体情報から検証用の特徴データを抽出する（ステップＳ４０２）。

【００９１】

次に、認証端末１１０の秘密情報復元モジュール１１３は、ＤＢサーバ１２０のテンプレートＤＢ１２２からテンプレートＴを取得し、テンプレートＴ及び検証用の特徴データに基づいて秘密情報ｓｋ’を復元する（ステップＳ４０３）。秘密情報ｓｋ’の復元処理（生成処理）の詳細については後述する。

【００９２】

なお、秘密情報復元モジュール１１３は、テンプレートＴを取得する場合、ユーザの識別情報を含むテンプレートの取得要求をＤＢサーバ１２０に送信する。ＤＢサーバ１２０は、テンプレートＤＢ１２２を参照して、ユーザの識別情報が対応付けられたテンプレートを検索し、検索結果を認証端末１１０に送信する。

【００９３】

次に、認証端末１１０の秘密情報検証モジュール１１４は、ＤＢサーバ１２０の検証情報ＤＢ１２３から検証情報ｖｋを取得し、検証情報ｖｋに基づいて秘密情報ｓｋ’の正しさを検証する（ステップＳ４０４）。すなわち、ｓｋ’＝ｓｋが成り立つか否かが判定される。

【００９４】

具体的には、秘密情報検証モジュール１１４は、秘密情報ｓｋ’から検証情報ｖｋ’を生成し、検証情報ｖｋ’とＤＢサーバ１２０から取得した検証情報ｖｋとが一致するか否かを判定する。なお、検証情報ｖｋ’の生成方法は、検証情報ｖｋの生成方法に応じて以下のような方法が考えられる。

【００９５】

（生成方法１）を採用した場合、検証情報ｖｋ’は式（１７）で与えられ、（生成方法２）を採用した場合、検証情報ｖｋ’は式（１８）で与えられ、（生成方法３）を採用した場合、検証情報ｖｋ’は式（１９）で与えられる。なお、式（１９）においてｐは検証情報ｖｋの生成時に用いた値と同一の値である。以上がステップＳ４０４の処理の説明である。

【００９６】

【数１７】

$$vk' = \mathrm{Hash}(sk') \ldots (17)$$

【００９７】

【数１８】

$$vk' = g\hat{}\varphi(sk') \ldots (18)$$

【００９８】

【数１９】

$$vk' = \mathrm{Enc}(sk', p) \ldots (19)$$

【００９９】

次に、認証端末１１０は、秘密情報ｓｋ’の検証結果に基づいて、暗号学的処理を実行する（ステップＳ４０５）。その後、認証端末１１０は、認証処理を終了する。

【０１００】

次に、テンプレートＴの生成処理及び秘密情報ｓｋの生成処理の詳細を図５を用いて説明する。図５は、実施例１のテンプレートＴの生成処理及び秘密情報ｓｋの生成処理を説明するフローチャートである。以下の説明ではベクトルの各要素の値は２進法又は１０進法で表現されるものとする。

【０１０１】

テンプレート生成モジュール１０３は、登録用の特徴データに対して変換処理及び正規化処理の少なくともいずれかを実行することによって、式（２０）に示す登録用の正規化特徴ベクトルＸを生成する（ステップＳ５０１）。登録用の正規化特徴ベクトルＸの各要素Ｘ＿ｉは実数である。なお、特徴データが既に正規化されている場合には、正規化処理は実行されなくてもよい。

【０１０２】

【数２０】

$$X = (X\_1, \ldots, X\_n) \quad \ldots (20)$$

【０１０３】

なお、後述する検証用の正規化特徴ベクトルＸ’は式（２１）のように表す。検証用の正規化特徴ベクトルＸ’の各要素Ｘ’＿ｉは実数である。

【０１０４】

【数２１】

$$X' = (X'\_1, \ldots, X'\_n) \quad \ldots (21)$$

【０１０５】

また、二つの特徴ベクトルＸ、Ｘ’の距離を式（２２）のように定義する。本実施例では、認証端末１１０は、式（２３）を満たす場合、二つの特徴ベクトルＸ、Ｘ’が同一ユーザから取得した生体情報の特徴ベクトルであると判定する。

【０１０６】

【数２２】

$$d(X, X') = \max\_i |X\_i - X'\_i| \quad \ldots (22)$$

【０１０７】

【数２３】

$$d(X, X') < 0.5 \quad \ldots (23)$$

【０１０８】

なお、ステップＳ５０１では、テンプレート生成モジュール１０３は、同一のユーザから取得した生体情報については高い確率で式（２３）が成立し、また、異なるユーザから取得した生体情報については高い確率で式（２３）が確立しないように、登録用の特徴データに対して適切な変換処理又はスケーリングを実行する。

【０１０９】

次に、テンプレート生成モジュール１０３は、登録用の正規化特徴ベクトルＸに基づいて、各要素Ｘ＿ｉの整数部分を要素とする整数ベクトルＸＩ、及び、各要素Ｘ＿ｉの小数部分を要素とする小数ベクトルＸＤを生成する（ステップＳ５０２）。整数ベクトルＸＩ及び小数ベクトルＸＤはそれぞれ式（２４）及び式（２５）のように表される。

【０１１０】

【数２４】

$$XI = (XI\_1, \ldots, XI\_n) \quad \ldots (24)$$

【０１１１】
【数２５】

$$XD = (XD\_1, \ldots, XD\_n) \quad \ldots (25)$$

【０１１２】
具体的には、テンプレート生成モジュール１０３は、要素Ｘ＿ｉの小数点以下を切り捨てることによって、整数ベクトルＸＩを生成する。また、テンプレート生成モジュール１０３は、式（２６）の演算を実行することによって小数ベクトルＸＤを生成する。
【０１１３】
【数２６】

$$XD\_i = X\_i - XI\_i \quad \ldots (26)$$

【０１１４】
次に、テンプレート生成モジュール１０３は、小数ベクトルＸＤの各要素ＸＤ＿ｉを、小数点以下ｋ桁に丸める丸め処理を実行し、式（２７）に示すような丸め小数ベクトルＸＤｒを生成する（ステップＳ５０３）。
【０１１５】
【数２７】

$$XDr = (XDr\_1, \ldots, XDr\_n) \quad \ldots (27)$$

【０１１６】
具体的には、テンプレート生成モジュール１０３は、小数点第ｋ＋１位以下を丸める。これによって、小数点第ｋ位までの値を要素とする丸め小数ベクトルＸＤｒが生成される。例えば、２進数で表現された要素ＸＤｒ＿ｉがｋ桁、すなわち、ｋビットである場合、丸め小数ベクトルＸＤｒのサイズはｎｋビットである。
【０１１７】
なお、丸め処理の方法としては、切り捨て、切り上げ、最近接丸め、及び偶数丸めなどが考えられる。
【０１１８】
このように要素の桁数を小さくすることによって、テンプレートＴのサイズをさらに小さくできる。ただし、丸めるサイズが小さい場合、丸め処理による誤差によって秘密情報の復元処理が失敗する確率が高くなる。当該確率と丸めるサイズを指定するｋとの関係については後述する。以下の説明では、丸め処理による誤差によって秘密情報の復元処理が失敗する確率を、エラー確率とも記載する。
【０１１９】
次に、テンプレート生成モジュール１０３は、ソルトを生成する（ステップＳ５０４）。
【０１２０】
具体的には、テンプレート生成モジュール１０３は、ランダムに生成された所定の長さ（例えば、ｓビット）の文字列若しくはビット列、又は、任意の数値範囲からランダムに生成された所定の長さの数値をソルトとして生成する。また、テンプレート生成モジュール１０３は、ユーザＩＤ及びカウンタなど、ユーザ毎又は登録用の生体情報毎に異なる値を用いてソルトを生成してもよい。ソルトの長さは、総当たり攻撃が困難な大きさ、例えば、１２８以上であればよい。
【０１２１】
次に、テンプレート生成モジュール１０３は、丸め小数ベクトルＸＤｒ及びソルトに基づいてテンプレートＴを生成する（ステップＳ５０５）。具体的には、テンプレート生成モジュール１０３は、式（２８）に示すような丸め小数ベクトルＸＤｒ及びソルトの組を、テンプレートとして生成する。
【０１２２】

【数２８】

$$T = (XDr, salt) \quad \ldots (28)$$

【０１２３】

なお、必ずしも小数ベクトルＸＤに対して丸め処理を実行しなくてもよい。また、必ずしもソルトを用いてテンプレートＴを生成しなくてもよい。すなわち、小数ベクトルＸＤ又は丸め小数ベクトルＸＤｒのみからテンプレートＴを生成してもよい。

【０１２４】

以上がテンプレートの生成処理の説明である。なお、テンプレート生成モジュール１０３は、テンプレートＴをテンプレートＤＢ１２２に登録するとともに、整数ベクトルＸＩ及びソルトを秘密情報生成モジュール１０４に出力する。

【０１２５】

秘密情報生成モジュール１０４は、整数ベクトルＸＩ及びソルトに基づいて秘密情報ｓｋを生成する（ステップＳ５１１）。

【０１２６】

例えば、秘密情報検証モジュール１１４は、式（２９）に示すように、ＳＨＡ２５６又はＳＨＡ３などの一方向性関数Ｈａｓｈ（）に整数ベクトルＸＩ及びソルトを連結したデータを入力することによって、秘密情報ｓｋを生成する。式（２９）の記号「｜｜」はデータの連結を表す。

【０１２７】

【数２９】

$$sk = Hash(XI \;||\; salt) \quad \ldots (29)$$

【０１２８】

また、秘密情報検証モジュール１１４は、式（３０）に示すように、ＡＥＳなどの暗号化関数又は鍵付きハッシュ関数などの関数Ｅｎｃ（）に、整数ベクトルＸＩ及びソルトを連結したデータ、並びに、任意のパラメータｐ’を入力することによって、秘密情報ｓｋを生成する。

【０１２９】

【数３０】

$$sk = Enc(XI \;||\; salt, p') \quad \ldots (30)$$

【０１３０】

以上が秘密情報の生成処理の説明である。

【０１３１】

ここで、本実施例のテンプレートＴの特徴について説明する。

【０１３２】

前述したように丸め小数ベクトルＸＤｒのデータサイズはｎｋビットである。ソルトの長さをｓビットとした場合、テンプレートＴのサイズは、式（２８）から（ｎｋ＋ｓ）ビットとなる。

【０１３３】

ここで、ｎ＝１０００、ｋ＝８、ｓ＝１２８とした場合、テンプレートＴのサイズは８１２８ビット（１０１６バイト）であり、約１ｋＢである。非特許文献１に記載の技術に基づいて生成されたテンプレートのサイズは３８．５ｋＢであることから、本実施例のテンプレートは、従来のテンプレートのサイズを約４０分の１に圧縮できる。したがって、従来技術と比較して、テンプレートのサイズを大幅に小さくできるため、生体認証システムの効率性を高めることができる。

【０１３４】

なお、小数ベクトルＸＤを用いてテンプレートＴを生成する場合であっても、特徴ベクトルＸの整数部分が削除されていることから、従来のテンプレートのサイズより十分小さ

い。

【０１３５】

また、テンプレートＴに含まれるソルトは生体情報に依存しない値である。そのため、攻撃者がテンプレートＴから生体情報を推定するためには、テンプレートＴに含まれる丸め小数ベクトルＸＤｒから登録用の正規化特徴ベクトルＸを推定する以外に方法がない。

【０１３６】

丸め小数ベクトルＸＤｒは、登録用の正規化特徴ベクトルＸの小数部分を丸めることによって生成されたデータである。より具体的には、丸め小数ベクトルＸＤｒは、特徴ベクトルＸの整数部分及び小数点第ｋ＋１位以下が削除されたデータである。生体情報は、同じユーザから取得した場合であっても誤差が生ずることから、正規化特徴ベクトルＸについても各要素について±０．５程度の範囲内で確率的な誤差が生じる。したがって、小数部分にはユーザを識別する情報がほとんど含まれていないことから、丸め小数ベクトルＸＤｒから正規化特徴ベクトルＸを推定又は復元するのは十分に困難である。

【０１３７】

以上のことから、本実施例のテンプレートの生成方法を適応することによって、テンプレートのサイズを削減し、かつ、十分な安全性を確保できる。

【０１３８】

なお、本実施例では特徴ベクトルを用いてテンプレートＴを生成しているが、特徴ベクトル以外のデータを用いても同様の特徴を有するテンプレートＴを生成できる。すなわち、登録端末１００は、特徴データ又は特徴データを用いて生成されたデータに基づいて、生体情報に発生する誤差部分のデータである誤差データ及び誤差データ以外のデータである定常データを生成する。登録端末１００は、誤差データを用いてテンプレートＴを生成し、また、定常データを用いて秘密情報ｓｋを生成する。

【０１３９】

次に、秘密情報ｓｋ’の復元処理の詳細を図６を用いて説明する。図６は、実施例１の秘密情報ｓｋ’の復元処理を説明するフローチャートである。

【０１４０】

秘密情報復元モジュール１１３は、検証用の特徴データに対して変換処理及び正規化処理の少なくともいずれかを実行することによって、式（２１）に示す検証用の正規化特徴ベクトルＸ’を生成する（ステップＳ６０１）。なお、特徴データが既に正規化されている場合には、正規化処理は実行されなくてもよい。

【０１４１】

次に、秘密情報復元モジュール１１３は、取得したテンプレートＴに含まれる丸め小数ベクトルＸＤｒ及び検証用の正規化特徴ベクトルＸ’に基づいて、式（３１）に示す差分特徴ベクトルＸＣを算出する（ステップＳ６０２）。

【０１４２】

【数３１】

$$XC = (XC\_1, \ldots, XC\_n) \quad \ldots(31)$$

【０１４３】

具体的には、秘密情報復元モジュール１１３は、式（３２）に示す演算を実行する。

【０１４４】

【数３２】

$$XC = X' - XDr \ (XC\_i = X'\_i - XDr\_i) \ldots(32)$$

【０１４５】

次に、秘密情報復元モジュール１１３は、差分特徴ベクトルＸＣに基づいて、式（３３）に示す復元整数ベクトルＸＩ’を生成する（ステップＳ６０３）。

【０１４６】

【数３３】

$$XI' = (XI'\_1, \ldots, XI'\_n) \quad \ldots (33)$$

【０１４７】

具体的には、秘密情報復元モジュール１１３は、式（３４）に示す演算を実行する。記号「［　］」は、括弧内の値の小数部分を切り捨てて、整数部分を取り出す演算を表す。

【０１４８】

【数３４】

$$XI'\_i = [XC\_i + 0.5] \quad \ldots (34)$$

【０１４９】

次に、秘密情報復元モジュール１１３は、復元整数ベクトルＸＩ’及び取得したテンプレートＴに含まれるソルトに基づいて復元秘密情報ｓｋ’を生成する（ステップＳ６０４）。秘密情報の生成方法は、ステップＳ５１１と同様の処理である。

【０１５０】

ここで、秘密情報ｓｋ及び復元秘密情報ｓｋ’の検証について説明する。

【０１５１】

ε＿ｉを式（３５）で定義する。ここで、ε＿ｉは、丸め小数ベクトルＸＤｒの丸め誤差を表す値であり、小数点第ｋ＋１位以下の値である。したがって、ε＿ｉの絶対値は2^{-k}未満となる。

【０１５２】

【数３５】

$$\varepsilon\_i = XD\_i - XDr\_i \quad \ldots (35)$$

【０１５３】

式（２６）、式（３２）、及び式（３５）より、式（３４）は式（３６）のように変形できる。

【０１５４】

【数３６】

$$\begin{aligned} XI'\_i &= [XC\_i + 0.5] \\ &= \left[X'\_i - XDr\_i + 0.5\right] \\ &= \left[X'\_i - (XD\_i - \varepsilon\_i) + 0.5\right] \\ &= \left[X'\_i - (X\_i - XI\_i - \varepsilon\_i) + 0.5\right] \\ &= \left[XI\_i - \left(X\_i - X'\_i + \varepsilon\_i\right) + 0.5\right] \quad \ldots (36) \end{aligned}$$

【０１５５】

式（２２）で定義した距離が式（３７）を満たし、かつ、ε＿ｉ＝０である場合、式（３８）が必ず成立する。また、式（３８）が成立する場合、秘密情報ｓｋ、ｓｋ’は、整数ベクトルＸＩ、ＸＩ’及びソルトから生成され、かつ、同じ生成方法であることから、式（３９）が成立する。また、式（３９）が成り立つ場合、検証情報ｖｋ、ｖｋ’は一致する。

【０１５６】

【数３７】

$$d(X, X') < 0.5 \Leftrightarrow |X\_i - X'\_i| < 0.5 \quad \ldots (37)$$

【０１５７】
【数３８】
$$XI'\_i = XI\_i \ldots (38)$$

【０１５８】
【数３９】
$$sk = sk' \ldots (39)$$

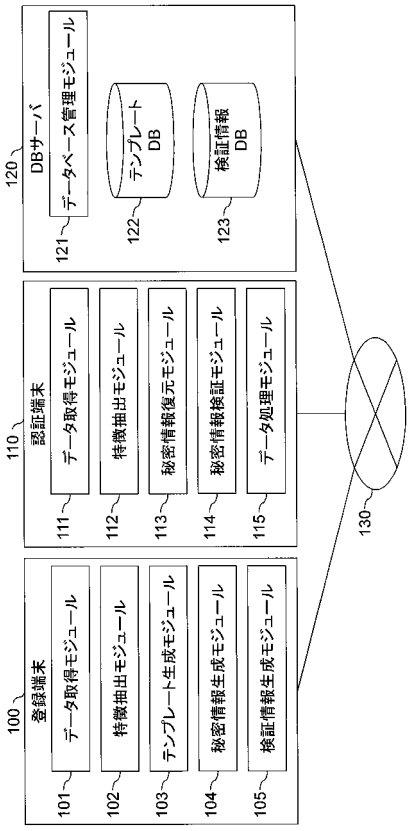
【０１５９】
式（３７）が成立する場合、０ではないε＿ｉが十分小さいと高い確率で式（３８）が成立する。逆に、式（３８）が成立しないケースは式（４０）を満たす場合に限られる。したがって、エラー確率は、おおよそ2^{-k}に比例するため、丸め処理におけるパラメータｋの増加に伴って指数関数的に減少する。したがって、ｋが８から１６の間の値の場合、検証処理の精度は十分高い。
【数４０】
$$0.5 - 2^{-k} < |X\_i - X'\_i| < 0.5 \ldots (40)$$

【０１６０】
ただし、テンプレートＴのサイズは、パラメータｋの大きさに比例するため、ｋの値は検証処理の精度及び効率性のバランスを考慮して決定する必要がある。
【０１６１】
以上で説明したように、本実施例によれば、テンプレートＴのサイズを大幅に削減でき、また、テンプレートＴから元の特徴データＸを推定するための攻撃に対する耐性を高めることができる。したがって、高い効率性及び安全性が保証されたバイオメトリック暗号技術を実現できる。
【０１６２】
特許請求の範囲に記載した以外の発明の観点の代表的なものとして、次のものがあげられる。
（１）データの秘匿に用いる秘密情報を用いた暗号学的処理を実行する計算機であって、
前記計算機は、
演算装置、前記演算装置に接続される記憶装置、及び前記演算装置に接続されるインタフェースを有し、
前記インタフェースを介して、入力された第１生体情報から生成された第１特徴データに発生する誤差を示す誤差特徴データに基づいて生成されたテンプレート、及び前記特徴データの誤差特徴データ以外の部分を示す定常特徴データに基づいて生成された第１検証情報を管理するデータベースと接続し、
前記演算装置は、
前記暗号学的処理の実行要求を受け付けた場合、ユーザから取得された第２生体情報に基づいて第２特徴データを生成し、
前記テンプレート及び前記第２特徴データに基づいて、第２秘密情報を生成し、
前記第２秘密情報に基づいて、第２検証情報を生成し、
前記第１検証情報及び前記第２検証情報を比較することによって前記第２秘密情報を検証し、
前記第２秘密情報の検証の結果に基づいて、前記暗号学的処理を実行することを特徴とする計算機。
（２）（１）に記載の計算機であって、
前記第１特徴データ及び前記第２特徴データは、実数を要素とする特徴ベクトルであり、
前記誤差特徴データは、前記特徴ベクトルの各要素の小数部分を要素とするベクトルであり、

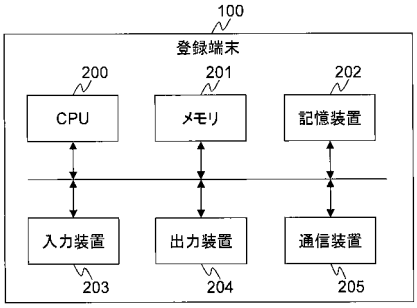
前記定常特徴データは、前記特徴ベクトルの各要素の整数部分を要素とするベクトルであることを特徴とする計算機。
（３）（２）に記載の計算機であって、
前記演算装置は、
前記第２特徴データ及び前記テンプレートに含まれる前記第１生体情報の誤差特徴データに基づいて、前記第２生体情報の定常特徴データを生成し、
ハッシュ関数、鍵付きハッシュ関数、及び暗号化関数の少なくともいずれかに、前記第２生体情報の定常特徴データを入力することによって前記第２秘密情報を生成することを特徴とする計算機。
【０１６３】
なお、本発明は上記した実施例に限定されるものではなく、様々な変形例が含まれる。また、例えば、上記した実施例は本発明を分かりやすく説明するために構成を詳細に説明したものであり、必ずしも説明した全ての構成を備えるものに限定されるものではない。また、各実施例の構成の一部について、他の構成に追加、削除、置換することが可能である。
【０１６４】
また、上記の各構成、機能、処理部、処理手段等は、それらの一部又は全部を、例えば集積回路で設計する等によりハードウェアで実現してもよい。また、本発明は、実施例の機能を実現するソフトウェアのプログラムコードによっても実現できる。この場合、プログラムコードを記録した記憶媒体をコンピュータに提供し、そのコンピュータが備えるプロセッサが記憶媒体に格納されたプログラムコードを読み出す。この場合、記憶媒体から読み出されたプログラムコード自体が前述した実施例の機能を実現することになり、そのプログラムコード自体、及びそれを記憶した記憶媒体は本発明を構成することになる。このようなプログラムコードを供給するための記憶媒体としては、例えば、フレキシブルディスク、ＣＤ－ＲＯＭ、ＤＶＤ－ＲＯＭ、ハードディスク、ＳＳＤ（Ｓｏｌｉｄ Ｓｔａｔｅ Ｄｒｉｖｅ）、光ディスク、光磁気ディスク、ＣＤ－Ｒ、磁気テープ、不揮発性のメモリカード、ＲＯＭなどが用いられる。
【０１６５】
また、本実施例に記載の機能を実現するプログラムコードは、例えば、アセンブラ、Ｃ／Ｃ＋＋、ｐｅｒｌ、Ｓｈｅｌｌ、ＰＨＰ、Ｊａｖａ（登録商標）等の広範囲のプログラム又はスクリプト言語で実装できる。
【０１６６】
さらに、実施例の機能を実現するソフトウェアのプログラムコードを、ネットワークを介して配信することによって、それをコンピュータのハードディスクやメモリ等の記憶手段又はＣＤ－ＲＷ、ＣＤ－Ｒ等の記憶媒体に格納し、コンピュータが備えるプロセッサが当該記憶手段や当該記憶媒体に格納されたプログラムコードを読み出して実行するようにしてもよい。
【０１６７】
上述の実施例において、制御線や情報線は、説明上必要と考えられるものを示しており、製品上必ずしも全ての制御線や情報線を示しているとは限らない。全ての構成が相互に接続されていてもよい。
【符号の説明】
【０１６８】
１００　登録端末
１０１、１１１　データ取得モジュール
１０２、１１２　特徴データ抽出モジュール
１０３　テンプレート生成モジュール
１０４　秘密情報生成モジュール
１０５　検証情報生成モジュール
１１０　認証端末

１１３　秘密情報復元モジュール
１１４　秘密情報検証モジュール
１１５　データ処理モジュール
１２０　ＤＢサーバ
１２１　データベース管理モジュール
１２２　テンプレートＤＢ
１２３　検証情報ＤＢ
１３０　ネットワーク
２００　ＣＰＵ
２０１　メモリ
２０２　記憶装置
２０３　入力装置
２０４　出力装置
２０５　通信装置

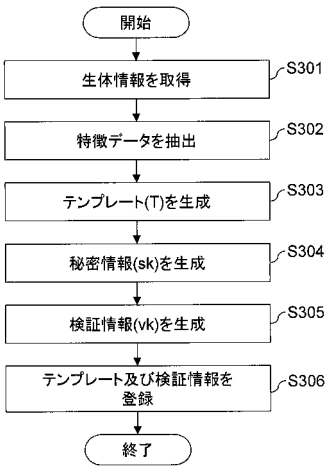
【図１】



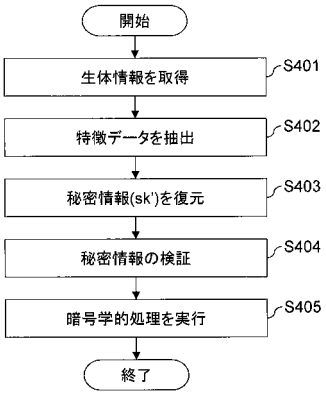
【図２】



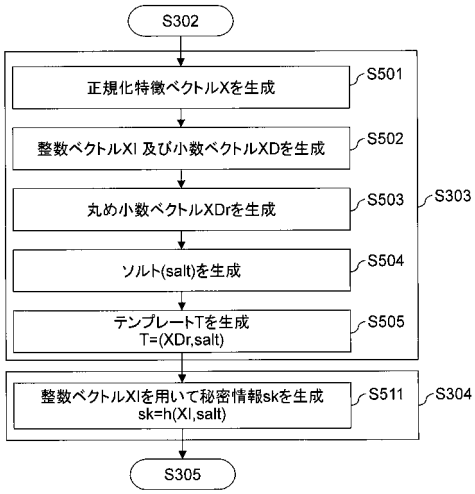
【図３】



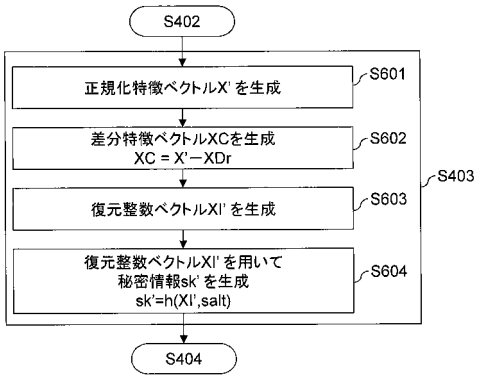
【図４】


開始
生体情報を取得
S401
特徴データを抽出
S402
秘密情報(sk')を復元
S403
秘密情報の検証
S404
暗号学的処理を実行
S405
終了

【図５】


S302
正規化特徴ベクトルXを生成
S501
整数ベクトルXI 及び小数ベクトルXDを生成
S502
丸め小数ベクトルXDrを生成
S503
S303
ソルト(salt)を生成
S504
テンプレートTを生成
T=(XDr,salt)
S505
整数ベクトルXIを用いて秘密情報skを生成
sk=h(XI,salt)
S511
S304
S305

【図６】


S402
正規化特徴ベクトルX' を生成
S601
差分特徴ベクトルXCを生成
XC = X'－XDr
S602
復元整数ベクトルXI' を生成
S603
S403
復元整数ベクトルXI' を用いて
秘密情報sk' を生成
sk'=h(XI',salt)
S604
S404