UNIVERSIDADE ESTADUAL PAULISTA
"Júlio de Mesquita Filho"

Pós-Graduação em Ciência da Computação

Marcelo Fornazin

Análise de Desempenho do Criptossistema Fuzzy Vault em Aplicações Reais

UNESP

2008

Marcelo Fornazin

Análise de Desempenho do Criptossistema Fuzzy Vault em Aplicações Reais

Orientador: Prof. Dr. Marcos Antônio Cavenaghi

Dissertação de Mestrado elaborada junto ao Programa de Pós-Graduação em Ciência da Computação – Área de Concentração em Sistemas de Computação, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação

UNESP

2008

MARCELO FORNAZIN

Análise de Desempenho do Criptossistema Fuzzy Vault em Aplicações Reais

Dissertação apresentada para obtenção do título de Mestre em Ciência da Computação, área de Sistemas de Computação junto ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista "Júlio de Mesquita Filho", Campus de São José do Rio Preto.

BANCA EXAMINADORA

Prof. Dr. Marcos Antônio Cavenaghi
Professor Assistente Doutor
UNESP – Bauru
Orientador

Prof. Dr. Ivan Rizzo Guilherme
Professor Assistente Doutor
UNESP – Rio Claro

Prof. Dr. Julio César López Hernández
Professor Doutor
UNICAMP

São José do Rio Preto, 04 de julho de 2008

AGRADECIMENTOS

Agradeço ao meu orientador Prof. Marcos Cavenaghi que confiou no meu potencial, me guiou e caminhou comigo no desenvolvimento trabalho.

À minha namorada, companheira e amiga Mariana que sempre esteve comigo, me apoiando nos momentos dificuldade e compartilhando os momentos de alegria.

À FAPESP pelo amparo financeiro ao projeto de pesquisa que resultou neste trabalho.

Aos amigos do Programa de Pós-Graduação em Ciência da Computação, Barbara, César, Fer, Iga e Juan, pelos grupos de estudo e pelas preciosas colaborações prestadas.

Ao Prof. Morgado, Prof Nilceu e Profa. Roberta pelo apoio na realização desse trabalho.

Aos amigos do LTIA e MStech que compartilham comigo o mesmo sonho e vivem na busca pela pesquisa e inovação.

Aos amigos da República Gato Morto que, mesmo sem entender nada, sempre ouviram minhas explicações e discutiram os rumos do projeto.

Ao meu Pai Toninho e minha Mãe Marisa, por terem me transmitido os valores que carrego comigo.

Ao vô Zé Piva, vó Jacinta e vó Angela que mesmo longe estão sempre presentes na minha vida.

E a todos que de alguma maneira colaboraram para o desenvolvimento desse trabalho.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

ACR – American College of Radiology
AES – Advanced Encryption System
DICOM – Digital Imaging and Communications in Medicine
DNA – Deoxyribonucleic Acid
EER – Equal Error Rate
FAR – False Acceptance Rate
FER – Failure to Enroll Rate
FRR – False Rejection Rate
FRVT – Face Recogntion Vendor Test
FTC – Failure to Capture Rate
FVC – Fingerprint Verification Competition
FVS – Fuzzy Vault Scheme
GAR – Genuine Acceptance Rate
GRR – Genuine Rejection Rate
HIPAA – Health Insurance Portability and Accountability Act
ITIRT – Independent Testing of Iris Recognition Technology
NEMA – National Electrical Manufacturers Association
NIST – National Institute of Standards and Technology
PACS – Picture Archiving and Communication System
RSA – Rivest Shamir e Adleman
SHA – Secure Hash Algorithm

LISTA DE FIGURAS

LISTA DE TABELAS

Resumo

Biometria trata do reconhecimento de indivíduos baseado em características fisiológicas ou comportamentais, sendo que umas de suas aplicações é autenticação biométrica. A autenticação biométrica tem vantagens com relação às senhas, no entanto, as informações biométricas também precisam ser protegidas. Ao contrário das senhas, a biometria apresenta variabilidade no sinal, isto é, raramente duas representações de uma mesma característica biometria são idênticas, sendo que, os criptossistemas tradicionais não suportam essa propriedade. Para contornar esse problema, criptossistemas biométricos unem biometria e criptografia para encriptar informações biométricas e protegê-las. *Fuzzy Vault* é uma construção criptográfica que pode ser utilizada na encriptação de características biométricas e atualmente, há propostas de implementação do *Fuzzy Vault* para impressões digitais, íris, face e assinaturas. O presente estudo implementa o *Fuzzy Vault* para impressões digitais e analisa seu desempenho em um ambiente de aplicação real e em um cenário de proteção de imagens médicas usando biometria. A proposta de implementação do *Fuzzy Vault* encripta e desencripta o *Fuzzy Vault* para impressões digitais realizando o alinhamento da impressão digital. A proteção de imagens médicas encripta imagens médicas com um criptossistema tradicional e encripta a chave criptográfica com a implementação do *Fuzzy Vault*. Os experimentos apresentaram entre 92% e 97,96% de GAR e 0% de FAR, esses resultados ocorreram em função dos mecanismos de alinhamento e identificação de pontos candidatos implementados neste estudo. O tempo de encriptação é constante em 0,8 s. Já o tempo de desencriptação apresenta grande variabilidade, e depende da quantidade de pontos candidatos, com mediana variando entre 16 ms e 1 s nas desencriptações com sucesso. No cenário de proteção de imagens médicas, o sistema proposto apresenta uma baixa sobrecarga e desempenho melhor se comparado a um criptossistema assimétrico, com encriptação até 17 vezes mais rápida e desencriptação até 245 vezes mais rápida.

PALAVRAS-CHAVE: Biometria, Impressão Digital, Criptografia, Criptossistemas Biométricos, Fuzzy Vault, Imagens Médicas

Abstract

Biometrics deals with people recognition based physiological or behavioral features where one of its application it biometric authentication. Biometric authentication has some advantages over passwords, but biometric information also need to be protected. Instead of passwords, biometrics has signal variability, i.e., two representation of the same biometric feature rarely are identical, and traditional cryptosystems does not support this feature. To overcome this issue biometric cryptosystems join biometrics and cryptography to encode biometric information and protected them. Fuzzy Vault is a cryptographic construction which can be used to encode biometric features. Today, there are some implementation proposals of Fuzzy Vault for fingerprints, iris, face and handwrite signature. This study implements Fuzzy Vault for fingerprints and analyses its performance in a real application environment and a scenario of medical image protection using biometrics. The proposed Fuzzy Vault implementation encodes and decodes Fuzzy Vault for fingerprints and performs fingerprint alignment. Medical image protecting encodes medical images with a traditional cryptosystems and encode its cryptographic key using the Fuzzy Vault implementation. Performance evaluation achieved between 92% and 97,96% of GAR and 0% of FAR, this results has been achieved with aligning and candidates points identification mechanisms implemented. Encoding time is constant in 0,8 s, but decoding time has a big variance which depends on number of candidates points, median varies between 16 ms and 1 s considering successful decodes. In the medical image protection scenario, the proposed system has a low overhead and better performance compared to a asymmetric cryptosystem, encoding time is 17 times better and decoding time is 245 times better.

KEYWORDS: Biometrics, Fingerprint, Cryptography, Biometric Cryptosystems, Fuzzy Vault, Medical Images

1. INTRODUÇÃO

Diversos sistemas eletrônicos se utilizam de autenticação para evitar que indivíduos não autorizados tenham acesso a serviços ou recursos do sistema. Na autenticação, um indivíduo declara uma identidade e provê uma informação com o objetivo de certificar que ele é a pessoa declarada (NIST, 1995). Atualmente, os sistemas eletrônicos realizam autenticação por meio de senhas. Segundo Uludag e Jain (2004), a utilização senhas é o ponto mais fraco de um sistema de segurança, pois senhas podem ser roubadas, copiadas ou até informadas a pessoas não autorizadas.

Em função de estarem relacionados a senhas ou a um cartão inteligente, os sistemas de autenticação tradicionais são baseados em algo que o indivíduo conhece ou possui (ULUDAG et al., 2004), ou seja, um usuário do sistema é identificado por conhecer uma senha ou possuir um cartão inteligente, mas não por quem ele realmente é. Nesses sistemas, outros indivíduos que saibam a senha ou tenham acesso ao cartão inteligente são capazes de acessar as informações do usuário, mesmo sem sua permissão. Outra maneira de se realizar autenticação em sistemas é a biometria. A biometria trata da verificação de padrões em características fisiológicas ou comportamentais de indivíduos para estabelecimento de métricas (JAIN et al., 2004), sendo que a autenticação biométrica é uma aplicação de biometria. Esta técnica consiste da verificação de uma ou mais características biométricas de um indivíduo para determinar se ele é ou não usuário de um sistema. Um exemplo de autenticação biométrica é a utilização de impressão digital para acessar um computador ou para editar um documento eletrônico.

A autenticação biométrica baseia-se em quem o usuário é, por isso, ela apresenta algumas vantagens com relação às senhas e cartões inteligentes. Ela é mais conveniente do que outros métodos de autenticação, pois não é necessário memorizar uma senha ou carregar um artefato. As características biométricas são mais difíceis de serem fisicamente copiadas ou reproduzidas se comparadas às senhas, no entanto, as características biométricas também precisam ser protegidas. Caso a representação digital de uma característica biométrica seja roubada, ela pode ser utilizada de maneira indevida em um processo de autenticação (ULUDAG et al., 2004).

Pode-se proteger uma característica biométrica utilizando criptografia, sendo esta a principal tecnologia para sistemas de segurança eletrônica. (BUCHMANN, 2002). A criptografia utiliza chaves para encriptar (cifrar) e desencriptar (decifrar) dados, assim, apenas o indivíduo que tem acesso à chave pode desencriptar o dado encriptado. Entretanto, há uma dificuldade em se proteger as chaves criptográficas, visto que, são grandes números aleatórios e isto as torna impossíveis de serem memorizadas, sendo necessário armazená-las em um local seguro (ULUDAG et al., 2005). Caso a característica biométrica seja encriptada com um sistema criptográfico tradicional também é necessário proteger a chave criptográfica.

Uma maneira de se proteger uma característica biométrica é combiná-la com outra informação de modo que ambas fiquem protegidas, esse método une um sistema biométrico a um sistema criptográfico (ULUDAG et al., 2005). Os sistemas com essas propriedades são chamados criptossistemas biométricos, eles encriptam uma característica biométrica e uma informação, protegendo ambas em um conjunto de dados encriptado. A informação é chamada de segredo e pode ser um código de identificação ou uma chave criptográfica tradicional, já a característica biométrica atua como uma chave criptográfica.

Os criptossistemas biométricos apresentam diferenças significativas com relação aos criptossistemas tradicionais, uma delas é a tolerância à variabilidade da representação característica biométrica. As chaves criptográficas tradicionais precisam ser exatas, já as características biométricas sofrem variações em diferentes aquisições, dificultando a aquisição de duas características biométricas exatamente iguais. Por isso, uma característica biométrica não pode ser empregada como chave de criptossistemas tradicionais. Os criptossistemas biométricos devem ser tolerantes à variabilidade existente entre as representações das características biométricas empregadas na encriptação e desencriptação.

Atualmente, há diversas propostas de criptossistemas biométricos, várias delas são baseadas no *Fuzzy Vault Scheme* (JUELS e SUDAN, 2002, 2006). O *Fuzzy Vault Scheme*, também chamado *Fuzzy Vault* ou FVS, é uma construção criptográfica que encripta um segredo e um conjunto de informações protegendo ambos em um cofre. O conjunto de informações pode ser uma característica biométrica, como é o caso do presente estudo. No *Fuzzy Vault* os conjuntos de informações utilizados na encriptação e desencriptação precisam ser semelhantes, mas não idênticos. O *Fuzzy Vault* tem sido objeto de grande investigação científica no campo dos criptossistemas biométricos, com propostas de implementação utilizando as seguintes características biométricas: impressão digital (CLANCY et al., 2002), (ULUDAG et al., 2006), face (WANG e PLATANIOTIS, 2007), íris (REDDY e BABU, 2008) e assinaturas (FREIRE-SANTOS et al., 2006). As implementações propostas na literatura focaram em analisar o desempenho da verificação biométrica na desencriptação do *Fuzzy Vault* com as características biométricas citadas, bem como a segurança que ele proporciona. No entanto, não foi encontrado nenhum estudo que avalie o desempenho de tempo de execução do Fuzzy Vault e a sua aplicabilidade em um ambiente real.

1.1. Objetivos

Implementar o *Fuzzy Vault* para impressões digitais e analisar seu desempenho em um cenário de aplicação real. Este estudo tem por objetivos específicos:

- Desenvolver uma biblioteca que permita a utilização do *Fuzzy Vault* integrado a outros sistemas de segurança;
- Implementar a encriptação e desencriptação do *Fuzzy Vault* para impressões digitais;
- Analisar desempenho do *Fuzzy Vault* para impressões digitais;
- Analisar a viabilidade do *Fuzzy Vault* para impressões digitais em um cenário de aplicação real: proteção de imagens médicas utilizando biometria.

1.2. Justificativas

Atualmente, a biometria vem sendo extensamente investigada como método de reconhecimento em diversas aplicações. As técnicas de reconhecimento biométrico apresentam desempenho aceitáveis para aplicações reais, no entanto, a segurança empregada na proteção das características biométricas é um tema pouco abordado nas pesquisas, sendo necessário desenvolver métodos que realizem verificação biométrica de maneira segura.

Uma maneira de proteger informações biométricas é através de criptossistemas biométricos. As propostas de criptossistemas biométricos atuais focaram em verificar a viabilidade da utilização de características biométricas nos criptossistemas biométricos, por exemplo, impressão digital e íris. A literatura pesquisada não aborda a avaliação de criptossistemas biométricos em aplicações reais como comércio eletrônico, governo eletrônico, gerenciamento de informações pessoais entre outras.

2. REVISÃO BIBLIOGRÁFICA

O presente estudo envolve conceitos de biometria, criptografia e criptossistemas biométricos. Este capítulo apresenta e descreve os conceitos e técnicas abordados neste estudo. As seções 2.1 e 0 descrevem conceitos de biometria e sistemas biométricos, a seção 2.3 descreve medidas de desempenho para avaliação dos sistemas biométricos e a seção 2.4 apresenta conceitos de criptografia. Na seção 2.5 são apresentados e discutidos os métodos de autenticação por senha, *smartcards* e biometria. A seção 2.6 apresenta as vulnerabilidades dos sistemas biométricos, discute as alternativas e descreve os criptossistemas biométricos encontrados na literatura. A seção 2.7 descreve o *Fuzzy Vault* e a seção 2.8 apresenta e discute os trabalhos que abordaram a implementação do *Fuzzy Vault* para impressões digitais.

2.1. Biometria

Os seres humanos sempre se utilizaram de características do corpo para identificação uns aos outros (JAIN et al. 2004) como, por exemplo, voz e face. A biometria utiliza esse mesmo princípio para reconhecimento de pessoas, isto é, trata do reconhecimento automatizado de indivíduos a partir de características biométricas. Característica biométrica é um modelo que representa uma característica fisiológica ou comportamental, com a qual é possível estabelecer uma métrica de similaridade entre duas representações da característica biométrica.

Segundo Jain et al. (2004) uma medida biológica, para ser considerada característica biométrica, deve possuir certas propriedades:

- **Universalidade**: todo indivíduo deve possuir a característica biométrica.
- **Exclusividade**: o sinal da característica deve identificar unicamente a pessoa, ou seja, não devem existir duas pessoas com a mesma representação da característica biométrica.
- **Permanência**: a característica biométrica não deve se alterar ao longo do tempo.
- **Coletabilidade**: a característica pode ser medida quantitativamente.
- **Desempenho**: a característica deve propiciar uma identificação precisa, em tempo aceitável e sem consumir recursos em excesso.
- **Aceitabilidade**: sistemas biométricos que utilizam determinada característica devem ser aceitos facilmente pelos indivíduos.
- **Circunvenção**: a característica biométrica não deve ser facilmente fraudada.

Há vários tipos de características biométricas, tais como os exemplificados na Figura 1. As características podem ser fisiológicas, como face, impressão digital, íris e DNA, ou comportamentais, como assinatura e dinâmica da caminhada.

As características biométricas apresentadas na Figura 1 possuem diferenças nas suas propriedades. A Tabela 1 apresenta uma comparação realizada por Jain et al. (2004) entre características biométricas bastante difundidas. De acordo com a citada tabela, pode-se notar que não há uma característica biométrica ótima para todos os tipos de aplicações. Também nota-se que impressões digitais apresentam um bom desempenho geral, segundo os critérios de comparação. Já íris, retina e DNA são bastante precisos, porém apresentam baixa aceitabilidade.

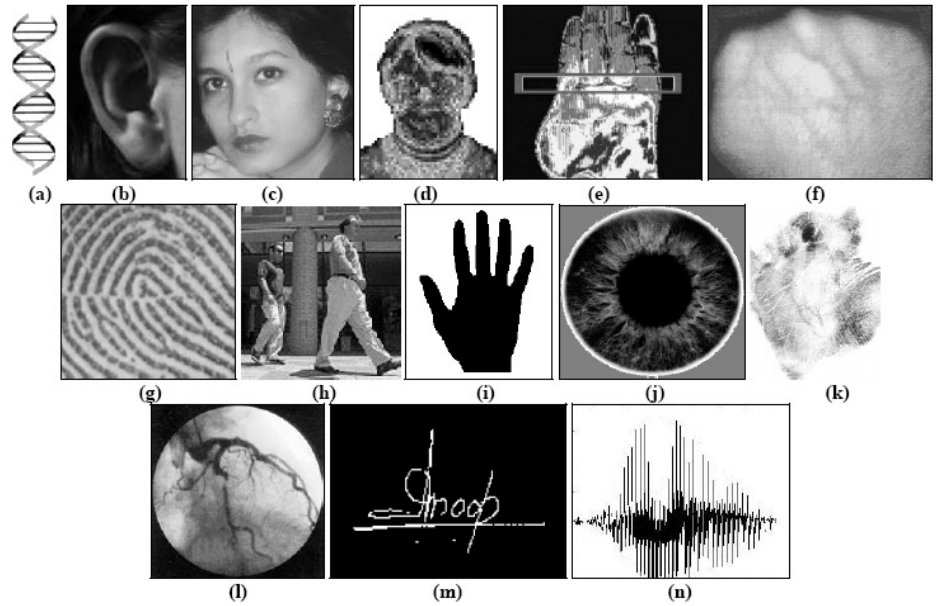

Figura 1 - Exemplos de características biométricas. (a) DNA; (b) orelha; (c) face; (d) termograma facial; (e) termograma da mão; (f) disposição das veias da mão; (g) impressão digital; (h) dinâmica da caminhada; (i) geometria da mão; (j) íris; (k) impressão da palma da mão; (l) retina; (m) assinatura e (n) voz (JAIN et al., 2004).

Tabela 1 - Comparação das propriedades das características biométricas (A = Alta, M = Média e B = Baixa) (JAIN et al., 2004)

Biometria	Universalidade	Exclusividade	Permanência	Coletabilidade	Desempenho	Aceitabilidade	Circunvenção
Face	A	B	M	A	B	A	A
Impressão Digital	M	A	A	M	A	M	M
Geometria das Mãos	M	M	M	A	M	M	M
Íris	A	A	A	M	A	B	B
Retina	A	A	M	B	A	B	B
Assinatura	B	B	B	A	B	A	A
Voz	M	B	B	M	B	A	A
DNA	A	A	A	B	A	B	B

2.2. Sistemas Biométricos

Sistemas biométricos são sistemas de reconhecimento de padrões que comparam representações de uma característica biométrica com o objetivo de estabelecer uma métrica de similaridade entre elas. Os sistemas biométricos podem ser utilizados em vários tipos de aplicações, tais como: autenticação de indivíduos, monitoramento de ambientes, análise forense entre outras (JAIN et al. 2004).

Um sistema biométrico possui duas fases principais: registro (matrícula) e reconhecimento. O registro consiste no armazenamento da representação da característica biométrica de um indivíduo em formato digital, está característica biométrica será utilizada em reconhecimentos posteriores e é denominada característica modelo (*template*). A Figura 2 apresenta os passos da matrícula em um sistema biométrico, a informação biométrica é cadastrada no sistema em conjunto com as informações do usuário.

O reconhecimento consiste em comparar outra representação da característica biométrica com a característica modelo para estabelecer a similaridade entre elas (JAIN et al., 2004). A representação da característica biométrica apresentada para o reconhecimento é chamada característica de entrada (*input*) ou de consulta (*query*). A similaridade entre as características modelo e de consulta identifica se elas pertencem a um mesmo indivíduo ou não.

Sistemas biométricos também implementam algoritmos de tratamento do sinal e extração de informações. O tratamento do sinal tem por objetivo melhorar a qualidade do sinal capturado, enquanto a extração de informações obtém apenas os dados necessários para o reconhecimento, removendo as informações irrelevantes.

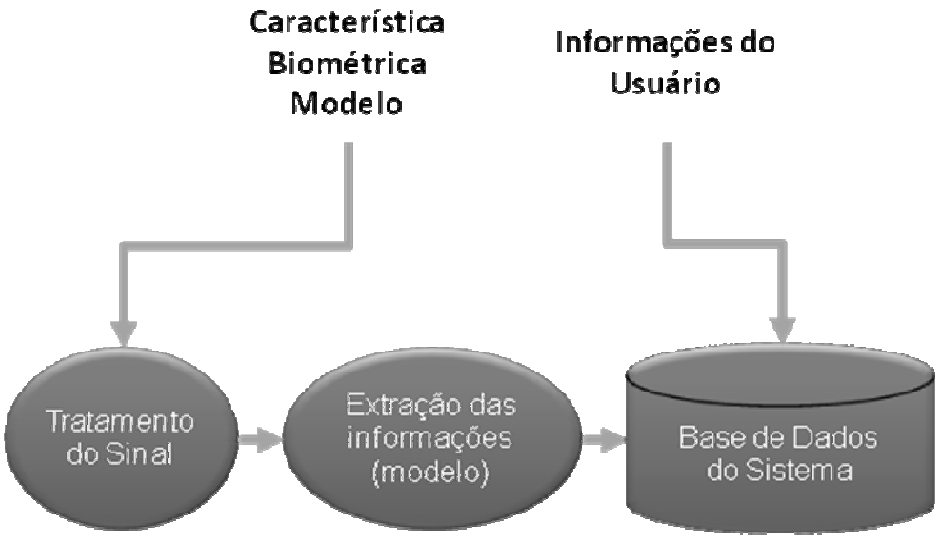

Figura 2 – Matrícula em um sistema biométrico

O reconhecimento pode ser realizado de duas maneiras: verificação (autenticação) ou identificação. Na verificação apresenta-se uma identidade e uma característica biométrica de consulta, o sistema recupera a característica modelo correspondente à identidade apresentada e a compara com a característica de consulta, a comparação verifica se a característica de consulta pertence à identidade apresentada. Ela tem a finalidade confirmar ou negar a identidade declarada. A Figura 3 apresenta o reconhecimento por verificação, nota-se que na verificação há a necessidade de se declarar a identidade.

A identificação compara uma característica biométrica fornecida com as características biométricas armazenadas em uma base de dados, o objetivo da identificação é encontrar na base de dados uma característica biométrica semelhante à fornecida. A Figura 4 apresenta o um sistema biométrico com reconhecimento por identificação. Nota-se que na identificação não é necessário declarar uma identidade, o sistema compara a característica biométrica de consulta com todas as características modelo cadastradas na base de dados e, no caso de identificação positiva, retorna a identidade encontrada.

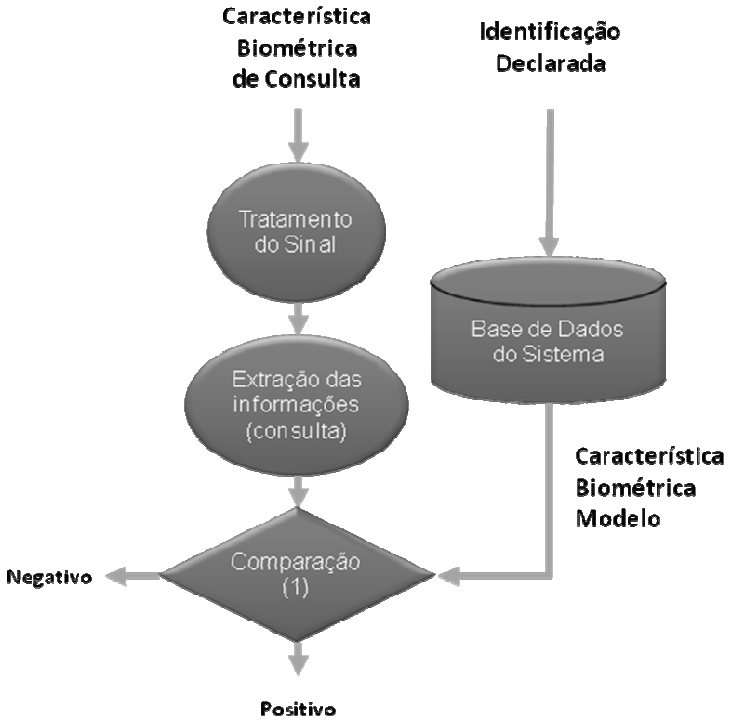

Figura 3 – Reconhecimento por verificação.

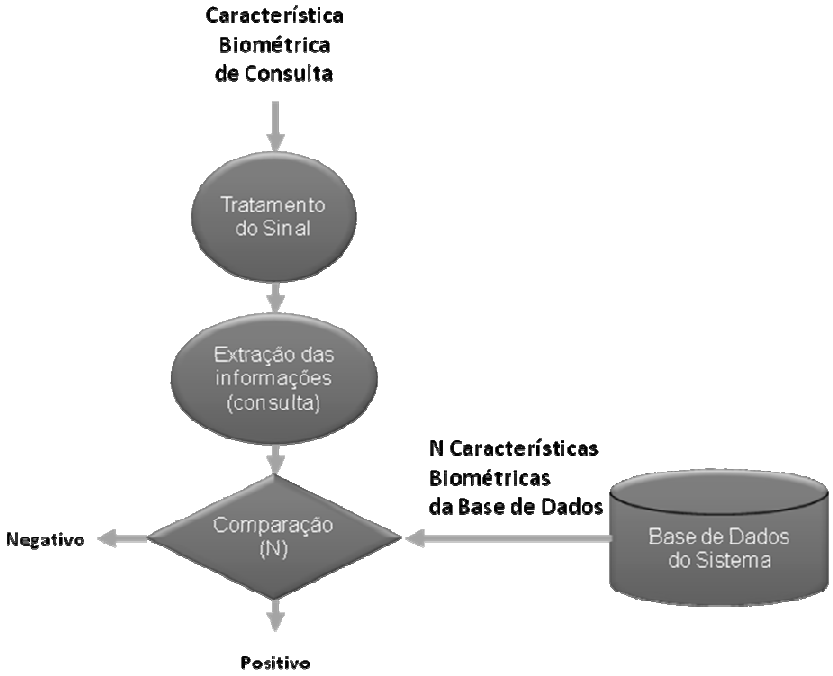

Figura 4 – Reconhecimento por identificação.

Os reconhecimentos por identificação e verificação possuem propriedades diferentes. O reconhecimento por identificação possui a conveniência de não ser necessário declarar uma identificação prévia, o que é necessário na verificação. O reconhecimento por identificação também é importante em aplicações de reconhecimento negativo, por exemplo, sistemas que identificam uma pessoa que ela nega ou ignora ser. Por exemplo, um sistema que identifica a quem pertence uma característica biométrica encontrada na cena de um crime. Já a verificação apresenta desempenho de tempo melhor do que a identificação, pois é necessário realizar apenas uma comparação, enquanto na identificação o número de comparações aumenta de acordo com o número de usuários cadastrados na base de dados (JAIN et al. 2004).

2.3. Desempenho de sistemas biométricos

Um sistema biométrico pode ser avaliado de diferentes maneiras: precisão do reconhecimento, eficiência, escalabilidade na identificação, tamanho do modelo entre outros. No presente estudo, o desempenho dos sistemas propostos foi avaliado em função da precisão do reconhecimento e da eficiência. A precisão do reconhecimento representa a capacidade do sistema biométrico reconhecer corretamente os indivíduos cadastrados e a eficiência representa o tempo consumido no registro da característica biométrica e na verificação.

A precisão de um sistema biométrico pode ser avaliada a partir da medição de erros no reconhecimento. Há dois tipos de erro: 1) erro no qual as características biométricas de duas pessoas diferentes são avaliadas com pertencendo a mesma pessoa (falsa aceitação) e 2) erro no qual duas características biométricas de uma mesma pessoa são avaliadas como pertencendo a pessoas diferentes (falsa rejeição). As medidas que representam esses erros são chamadas de taxa de falsa aceitação (*False Acceptance Rate - FAR*) e taxa de falsa rejeição (*False Rejection Rate - FRR*).

Os erros em um sistema biométrico podem ser formulados através de testes estatísticos de hipótese. A comparação entre duas características biométricas resulta em uma pontuação $S(X_Q, X_I)$ que quantifica a similaridade entre a característica biométrica de consulta (X_Q) e a característica biométrica modelo (X_I). Sendo um modelo de característica biométrica do indivíduo I representada por X_I e a característica biométrica de consulta representada por X_Q, então a hipótese nula e a hipótese alternativa são:

- H_0 - a característica de consulta X_Q não pertence à mesma pessoa da característica modelo X_I (hipótese nula);
- H_1 - a característica de consulta X_Q pertence à mesma pessoa da característica modelo X_I (hipótese alternativa).

As decisões associadas são:

- D_0 - a pessoa não é quem ela declarou ser;
- D_1 - a pessoa é quem ela declarou ser.

A regra de decisão atende ao seguinte requisito: se a pontuação $S(X_Q, X_I)$ for menor que um limiar t, então decide-se por D_0, caso o contrário decide-se por D_1. De acordo com a Tabela 2 existem quatro situações possíveis, que podem resultar no seguintes erros:

- Tipo I: falsa aceitação (decide-se por D_1, quando H_0 é verdade);
- Tipo II: falsa rejeição (decide-se por D_0 quanto H_1 é verdade);

Tabela 2 – Erros existentes em sistemas biométricos

	Decisões	
Hipóteses	**D0**	**D1**
H0	Rejeição Genuína Sem erro	Falsa Aceitação Erro Tipo I
H1	Falsa Rejeição Erro Tipo II	Aceitação Genuína Sem erro

A Taxa de Falsa Aceitação (*FAR*) é a probabilidade do erro Tipo I (nível de significância) no teste de hipótese. A Taxa de Falsa Rejeição (FRR) é a probabilidade do erro Tipo II, como se segue:

- $FAR = P(D_1|H_0)$
- $FRR = P(D_0|H_1)$

Para se avaliar a precisão de um sistema biométrico deve-se calcular a pontuação de múltiplas características de um mesmo indivíduo e a pontuação gerada a partir da comparação entre características de diferentes indivíduos, com isso, pode-se identificar a *FAR* e *FRR*. Caso o limiar *t* utilizado na comparação da pontuação seja ajustável pode-se realizar uma distribuição para os diferentes valores de *t*. A Figura 5 apresenta um gráfico com as distribuições dos erros nos sistemas biométricos em função do limiar *t*. Quanto maior o valor do limiar, menor a *FAR* e maior a *FRR*. Isto significa que o sistema será mais seguro contra impostores, mas menos tolerante com os genuínos. Quanto menor o valor do limiar, o sistema será menos seguro e mais tolerante com os genuínos (JAIN et al., 2004).

É necessário definir um valor de *t* que apresente uma boa relação entre a *FAR* e *FRR* e este valor depende das características da aplicação. A Taxa de Erro Igual (*Equal Error Rate – EER*) representa um valor no qual a *FAR* e a *FRR* são iguais. A *EER* geralmente é utilizada na comparação de dois sistemas biométricos diferentes. Quanto menor a *EER*, maior a precisão de reconhecimento de um sistema biométrico. A Tabela 3 apresenta o desempenho do estado da arte nos sistemas de reconhecimento biométrico para diferentes características. Os resultados foram alcançados em competições internacionais, como a FVC (MAIO et al., 2002) e análises do governo norte-americano. Nota-se que a impressão digital apresenta bons valores para *FAR* e *FRR*, mesmo quando avaliadas em bases de dados com grande número de indivíduos.

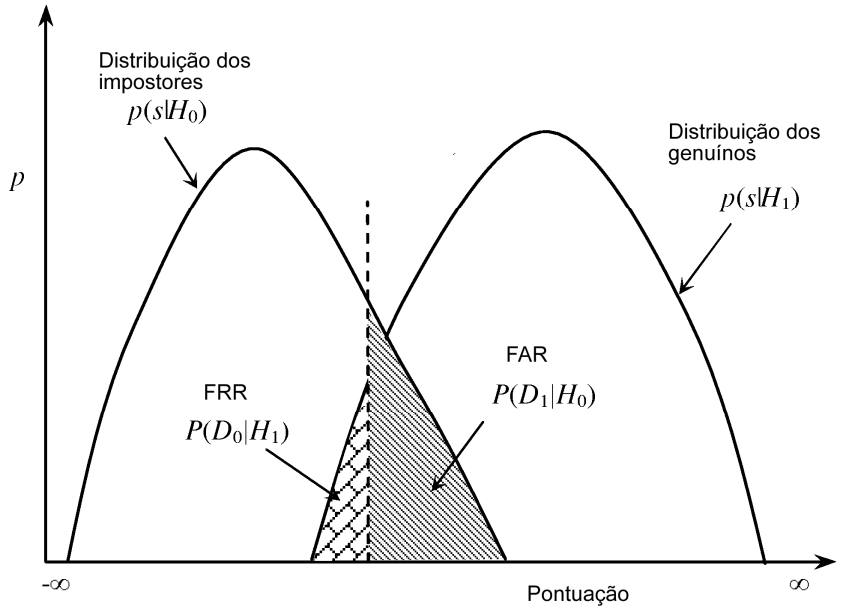

Figura 5 – Distribuição das probabilidades de erros nos sistemas biométricos (JAIN, ROSS e PRABHAKAR, 2001)

Tabela 3 – Desempenho de sistemas biométricos para diferentes características (BIOMETRICS, 2008)

Biometria	EER	FAR	FRR	Indivíduos	Observações
Face	n.d.	1%	10%	37437	Condições variadas de iluminação (FRVT)
Impressão digital	n.d.	1%	0.1%	25000	Dados operacionais do Governo Americano
Impressão digital	2%	2%	2%	100	Rotação e distorção (FVC)
Geometria das mãos	1%	2%	0.1%	129	Utilização de anéis e aquisição inapropriada
Íris	< 1%	0.94%	0.99%	1224	Ambiente fechado (ITIRT)
Íris	0.01%	0.0001%	0.2%	132	Condições ideais (NIST)
Dinâmica de digitação	1.8%	7%	0.1%	15	Durante um perído de 6 meses
Voz	6%	2%	10%	310	Independente de texto, múltiplas línguas (NIST)

Também pode-se medir a precisão de um sistema biométrico pelo sucesso no reconhecimento ou na rejeição. As medidas utilizadas são: Taxa de Aceitação de Genuína (*Genuine Acceptance Rate - GAR*) e a Taxa de Rejeição Genuína (*Genuine Rejection Rate - GRR*). A *GAR* representa a probabilidade do sistema reconhecer corretamente um indivíduo cadastrado no sistema, isto é, duas características de um mesmo indivíduo serem consideradas semelhantes. A *GAR* é representada por *1 – FRR* ou $P(D_1/H_1)$. A *GRR* representa a probabilidade do sistema não reconhecer um indivíduo que não está cadastrado no sistema, isto é, características de indivíduos diferentes não apresentam semelhanças. A *GRR* é representada por *1 – FAR* ou *P(Do/Ho)*. *GAR* e *GRR* representam medidas de acerto do sistema.

Outras medidas devem ser consideradas em sistemas de reconhecimento biométrico automatizado: Taxa de falha de registro (*Failure to Enroll Rate – FER*) e a Taxa de Falha de Captura (*Failure to Capture Rate – FTC*). A *FER* é a porcentagem de características que foram consideradas inválidas durante o registro. Essa falha ocorre quando o sinal biométrico não atende à qualidade mínima necessária para o sistema realizar o registro da característica biométrica. A *FTC* é a probabilidade de o sistema falhar na detecção da característica biométrica quando ela é apresentada corretamente.

2.4. Criptografia

Uma maneira de se manipular informações sigilosas de forma segura é utilizar criptografia. Segundo Buchmann (2002), criptografia é a principal tecnologia dos sistemas de segurança eletrônica e possui duas fases: encriptação e desencriptação.

O processo de encriptação transforma um texto plano (dado que pode ser entendido) em um texto encriptado (dado que não possui significado) o qual só pode se tornar novamente plano após o processo de desencriptação. O texto encriptado não possui nenhum sentido para um indivíduo que não tenha condições de desencriptá-lo.

Ou seja, o texto encriptado não pode ser entendido sem que seja desencriptado. Dessa forma, ele pode ser armazenado ou transmitido de maneira segura, pois somente indivíduos autorizados possuem condições de desencriptá-lo.

Criptossistemas são os mecanismos que realizam encriptação e desencriptação de dados para manter sigilo dos mesmos e também são chamados de esquemas de encriptação criptográfica.

Criptografia utiliza chaves para encriptar e desencriptar dados, dessa forma, apenas o indivíduo que tem acesso à chave pode desencriptar o dado encriptado. Assim, dados encriptados podem ser armazenados e transmitidos com segurança. Se um indivíduo não autorizado acessa o dado encriptado, ele não será capaz de entender a informação protegida.

Um criptossistema (STINSON, 2002) é uma tupla de 5 elementos (P, C, K, E, D), onde as seguintes condições são satisfeitas:

- P é um conjunto finito dos possíveis texto planos;
- C é um conjunto finito dos possíveis textos encriptados;
- K é o espaço das chaves, um conjunto finito das possíveis chaves;
- Para cada $k \in K$, há uma regra de encriptação $e_k \in E$ e uma regra de desencriptação $d_k \in D$. Cada $e_k : P \rightarrow C$ e $d_k : C \rightarrow P$ são funções tais que $d_k(e_k(x)) = x$ para todo texto plano $x \in P$.

Alguns exemplos de criptossistemas são: Advanced Encryption Standard (NIST, 2001), Data Encryption Standard (NIST, 1999) e RSA (RIVEST et al. 1978).

Um criptossistema é considerado seguro quando só é possível se desencriptar um texto encriptado $c \in C$ se a chave $k \in K$ que foi utilizada na encriptação de c for conhecida.

Há diversas propriedades que diferenciam os criptossistemas. Entre elas está a simetria da chave criptográfica. Com relação à simetria da chave criptográfica há dois tipos de criptossistemas: simétricos e assimétricos (STINSON, 2002), (BUCHMANN, 2002).

Criptografia simétrica, também conhecida como criptografia de chave privada, é um modelo criptográfico no qual a mesma chave usada na encriptação é usada na desencriptação, sendo que o dado encriptado com uma determinada chave só pode ser desencriptado com a mesma chave. A chave deve ser mantida em segredo, do contrário qualquer indivíduo que tenha acesso a ela é capaz de desencriptar o dado encriptado. Sendo assim, as chaves de criptossistemas simétricos devem ser armazenadas de maneira segura.

Criptografia assimétrica, também conhecida como criptografia de chave pública é um modelo criptográfico no qual há um par de chaves relacionadas matematicamente: chave pública e chave privada. A chave pública é utilizada na encriptação do texto plano para texto encriptado. Dados encriptados com a chave pública, só podem ser desencriptados com a chave privada correspondente e vice-versa. A chave pública não é capaz de desencriptar um dado encriptado por ela, o mesmo acontece com a chave privada. Um indivíduo que possua par de chave pública e privada pode distribuir a chave pública para outros indivíduos. Os indivíduos que a receberem podem utilizá-la para encriptar dados e enviá-los ao indivíduo que possui a chave privada. Desse modo, é apenas necessário manter em segredo a chave privada e, como ela não precisa ser transmitida a outros indivíduos, é mais fácil protegê-la. Criptografia de chave pública pode ser utilizada para encriptação de dados ou verificação de autenticidade (STINSON, 2001).

A criptografia simétrica possui melhor desempenho de encriptação e desencriptação com relação à criptografia de chave assimétrica. No entanto, ela apresenta a desvantagem da manipulação protegida na chave. A criptografia de chave pública possui a vantagem do par de chaves, onde apenas a chave privada deve ser manipulada de maneira segura e não é necessário transmiti-la ou compartilhá-la. Já na criptografia simétrica, a chave usada na encriptação é a mesma da desencriptação e precisa ser manipulada de maneira segura. Diversas aplicações utilizam criptografia simétrica para proteção de dados e criptografia assimétrica para troca de chaves. Dessa maneira, elas podem proteger dados com bom desempenho e transmitir as chaves com segurança.

As chaves de criptossistemas tradicionais, geralmente, são grandes números aleatórios, como é o caso do AES que possui chaves de comprimento de 128 bits (NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, 2001), isto torna as chaves impossíveis de serem memorizadas. Esta propriedade faz com as chaves precisem ser armazenadas em um local seguro, como, por exemplo, um servidor ou um cartão inteligente (*smartcard*) (ULUDAG et al. 2004). Para que as chaves criptográficas sejam acessadas somente por indivíduos autorizados, é necessário que haja um mecanismo de proteção contra acesso indevido. Uma maneira de se proteger chaves criptográficas é através de autenticação (NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, 1995) (ULUDAG et al. 2004).

2.5. Biometria e outros Métodos de Autenticação

Autenticação é uma medida utilizada para prevenir que pessoas não autorizadas acessem um sistema (NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, 1995), por exemplo, um sistema pode exigir autenticação do usuário para que ele tenha acesso a um recurso ou realize uma operação no sistema.

Para ser autenticado, o indivíduo precisa ser identificado pelo sistema e também prover uma informação que certifique que ele é a pessoa da identificação. Por exemplo, para realizar a operação, o indivíduo precisa informar o seu nome e uma senha. Caso seja a senha correta, o sistema permite que o indivíduo realize as operações disponibilizadas para a identificação apresentada.

A autenticação de um indivíduo em um sistema pode ser realizada de três maneiras, que também podem ser combinadas (BOVELANDER e RENESSE, 1995):

- Algo que o indivíduo saiba, por exemplo, senhas ou códigos de identificação pessoal (PIN);
- Algo que o indivíduo possua, como, cartão de identificação ou *smartcard*;
- Um atributo biométrico, por exemplo, impressão digital, íris entre outros.

Indivíduos podem ser autenticados através de senhas, isto é, um indivíduo é questionado pelo sistema sobre a senha que corresponde à sua identificação e, caso informe a senha correta, acessa o sistema. Segundo Uludag e Jain (2004) a utilização de senhas não é um método seguro para identificação de indivíduos, pois senhas podem ser roubadas, copiadas ou até informadas a pessoas não autorizadas. As senhas também são suscetíveis a ataques por meio de dicionários (KLEIN, 1990), esses ataques se utilizam de uma base de dados de palavras para tentar acessar um sistema que requer autenticação baseada em senhas. Ataques por dicionário possuem relativo sucesso, pois a maioria das senhas são números familiares ao usuário como, por exemplo, o número do telefone e data de nascimento.

A autenticação por meio de um artefato que um indivíduo possui tem crescido nos últimos anos, principalmente pela utilização de *smartcards*. A autenticação por meio de *smartcards* é feita apresentando um *smartcard* ao sistema, em alguns casos além do *smartcard* é necessário informar um número de identificação. Caso o número

de identificação corresponda ao armazenado no *smartcard* o indivíduo acessa o sistema. Uma vantagem dos *smartcards* é a possibilidade de se armazenar informações além das necessárias para autenticação. Outra vantagem é a resistência física a ataques, isto é, não é possível acessar informações protegidas no *smartcard*, como, por exemplo, o número de identificação. Além disso, o *smartcard* não pode ser duplicado como as senhas, dessa forma, dois indivíduos não podem utilizá-lo ao mesmo tempo ou em lugares diferentes. No entanto, apesar de armazenarem informações de maneira segura, os *smartcards* não provêm toda a segurança necessária. O *smartcard* pode ser roubado ou emprestado e o número de identificação apresenta vulnerabilidades parecidas com as senhas. Se um indivíduo obtiver acesso ao *smartcard* e ao número de identificação, ele pode se autenticar em um sistema para o qual ele não possua permissão (CLANCY et al., 2003).

A autenticação biométrica verifica uma característica biométrica de um indivíduo e determina se ele possui permissão para acessar o sistema (JAIN et al., 2004). Caso a característica biométrica de consulta seja suficientemente semelhante à característica biométrica modelo, o usuário acessa o sistema.

Os dois primeiros métodos de autenticação são baseados na posse, ou seja, um indivíduo é identificado por possuir uma senha ou um *smartcard* e não por quem ele realmente é. Já na autenticação biométrica, o indivíduo precisa apresentar a característica biométrica que dificilmente pode ser copiada ou roubada (ULUDAG et al., 2004).

Há vários motivos para se utilizar biometria ao invés de métodos tradicionais de autenticação. Um deles é a conveniência, isto é, não é necessário carregar artefatos ou memorizar senhas para se realizar a autenticação. Outro motivo é a necessidade de métodos de verificação mais confiáveis. A biometria elimina as chances de outro indivíduo roubar um cartão de identificação ou copiar uma senha, pois é necessário que

o indivíduo esteja presente para captura do sinal da característica biométrica. A queda nos custos de equipamentos e a popularização do uso da biometria também favorecem a utilização dela em diversos ambientes (WOODWARD et al., 2003).

Os sistemas de autenticação biométrica requerem que uma característica biométrica modelo esteja armazenada na base de dados do sistema (ULUDAG et al. 2004), sendo que as características biométricas são armazenadas em conjunto com as informações dos usuários. Esses sistemas requerem acesso à característica modelo para comparação e não há nenhum instrumento de proteção a ela, como a característica biométrica é armazenada no sistema, o roubo de informações do sistema dá acesso a esta característica. Apesar dos sistemas biométricos evitarem que usuário gerencie informações complexas, caso alguém obtenha acesso à característica biométrica modelo pode utilizá-la para acessar o sistema, tornando-o vulnerável (ULUDAG et al. 2004).

2.6. Criptossistemas Biométricos

Caso a característica biométrica modelo armazenada no banco de dados de um sistema biométrico seja roubada, ela pode ser utilizada para autenticação no sistema e em outros sistemas que utilizem a mesma característica biométrica. Dessa forma, os sistemas biométricos necessitam de mecanismos que permitam que uma característica biométrica possa ser substituída caso ela seja comprometida. A representação da característica biométrica de um indivíduo não pode ser usada em sistemas diferentes para se evitar que a característica roubada de um sistema seja utilizada em outro (ULUDAG et al. 2004).

Por exemplo, dois sistemas diferentes (A e B) devem armazenar representações distintas de uma mesma impressão digital. Essas representações seriam como duas senhas diferentes, dessa forma, a representação da impressão digital armazenada no sistema A não pode ser usada no sistema B e, caso seja roubada, pode ser revogada e substituída por outra representação da mesma impressão digital.

Uma maneira de se resolver esses problemas é armazenar a característica biométrica modelo de uma maneira modificada. Isto é, em vez de armazenar a representação *x* da característica biométrica, armazena-se uma versão transformada *H(x)* e na autenticação a característica de consulta *x'* é transformada pelo mesmo processo. Diferentes aplicações podem utilizar diferentes transformações, isso impossibilita que uma característica transformada, possa ser utilizada em outra aplicação. Se a transformação *H* for irreversível, ela não revela nenhuma informação do usuário. No entanto, esse processo aumenta em muito a taxa de erro na autenticação. Isto por que a comparação no espaço transformado se torna difícil devido à grande variabilidade que pode existir entre a característica modelo e a característica de consulta. Se *H* for reversível, a comparação se torna mais fácil. No entanto, a segurança do sistema é comprometida (ULUDAG et al. 2004).

Caso uma característica biométrica seja roubada ou copiada há uma limitação física para substituí-la, já que não se pode substituir um atributo físico de um indivíduo como é feito nas senhas que podem ser facilmente substituídas. Uma maneira de evitar esse problema é a biometria "cancelável" (MALTONI et al., 2003), este método associa uma característica biométrica a uma senha que altera as propriedades da característica, dessa forma, cada senha gera uma nova característica que pertence ao mesmo indivíduo. Na autenticação, o indivíduo apresenta a característica biométrica original e a senha e o sistema aplica uma transformação na característica biométrica usando a senha, a versão

transformada é então utilizada na autenticação. Caso a senha seja descoberta, pode-se gerar outra senha e, consequentemente, uma nova representação da característica biométrica. Pode-se utilizar uma senha para cada sistema, evitando que a característica biométrica de um sistema seja utilizada em outro. No entanto, a senha utilizada para gerar a característica biométrica sofre dos mesmos problemas das senhas tradicionais.

Outra maneira de se proteger as características biométricas é alterando a forma como as informações do usuário são armazenadas no banco de dados. Ao invés de se armazenar as informações originais, pode-se escondê-las junto com a característica biométrica. As informações e a característica biométrica podem ser embaralhadas para armazenamento, desse modo, um atacante não consegue entender as informações e a característica biométrica, a não ser que seja utilizada uma característica biométrica semelhante. Esse modelo é conhecido como geração de chave biométrica ou ligação (ULUDAG et al. 2004). As informações armazenadas no sistema podem ser chaves criptográficas ou um número de identificação pessoal. Assim, só consegue acesso às chaves criptográficas ou ao número de identificação, o indivíduo que se autenticar apresentando a característica biométrica correta.

Esquemas de combinação tradicionais baseados em espalhamentos não se aplicam às características biométricas. Criptossistemas tradicionais foram desenvolvidos para aceitarem apenas chaves criptográficas idênticas, eles não suportam chaves criptográficas geradas a partir de características biométricas. A utilização de características biométricas possui uma diferença com relação à utilização de senhas. Senhas são geralmente armazenas como uma combinação de P na forma de um espalhamento $H(P)$, esta técnica é conhecida como *hashing*, com ela, pode-se identificar se o usuário apresentou a senha correta comparando o *hash* armazenado com o *hash* da senha digitada pelo usuário. No entanto, isto não é válido para biometria, pois as

características biométricas possuem grande variabilidade em diferentes aquisições e ao longo do tempo. Combinações de características biométricas de um mesmo indivíduo raramente serão idênticas (JUELS e WATTENBERG, 1999). A variação nas diferentes aquisições da característica pode gerar espalhamentos muito distintos.

Os sistemas biométricos também não são perfeitos, o que pode gerar erros no processo de encriptação e desencriptação. Eles também não podem ser executados no domínio criptográfico por diversos fatores, e um deles é a dificuldade em se estabelecer uma métrica de similaridade na representação encriptada. Os comparadores biométricos necessitam fazer alinhamento das representações das características biométricas e este processo é muito difícil de ser feito no espaço criptográfico. Identificadores biométricos possuem representação variável e fora de ordem, é difícil ordenar a característica encriptada para que haja uma correspondência entre as características nas duas representações (ULUDAG et al. 2004). Isto faz com que os mecanismos de geração de chaves biométricas tenham o desafio de realizar a comparação biométrica no domínio desencriptador sem revelar informações significativas ao atacante. Também é desejável que não haja uma correlação sistemática entre a identidade e a característica biométrica, de modo que ela possa ser explorada por um atacante (ULUDAG et al. 2004). Atualmente, há diversas pesquisas focadas em encontrar criptossistemas que solucionem os problemas envolvidos na utilização de criptografia e biometria, os resultados dessas pesquisas são os chamados criptossistemas biométricos.

Os criptossistemas biométricos têm por objetivo proteger uma característica biométrica em conjunto com outra informação (segredo). Dessa forma, a característica biométrica funciona como uma chave criptográfica que é capaz de encriptar ou desencriptar um dado.

A Figura 6 apresenta o processo de encriptação em um criptossistema biométrico. Um segredo e uma característica biométrica modelo passam por um processo de encriptação que gera um conjunto de dados que não revela nenhuma informação sobre a característica biométrica ou sobre o segredo.

A Figura 7 apresenta o processo de desencriptação nos criptossistemas biométricos, os dados protegidos passam pela desencriptação em conjunto com uma característica biométrica de consulta. Caso a característica de consulta seja suficientemente semelhante à característica modelo, o segredo é desencriptado e liberado para utilização.

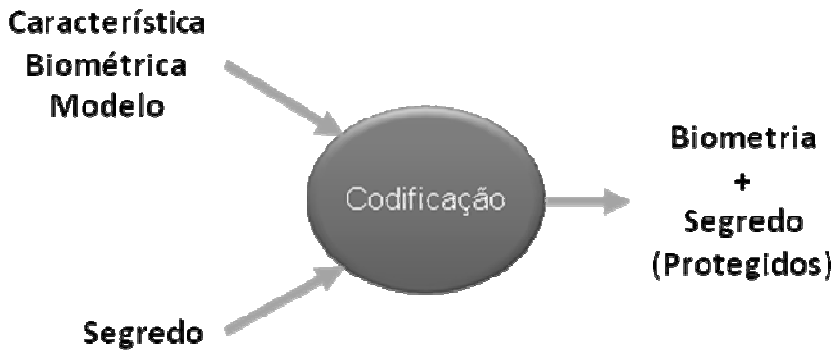

Figura 6 – Encriptação em um criptossistema biométrico

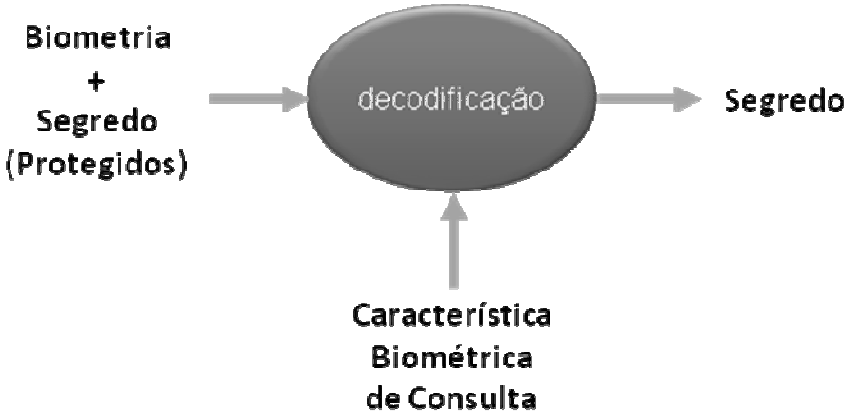

Figura 7 – Desencriptação em um criptossistema biométrico

Há diversas propostas de criptossistemas biométricos sendo que elas têm por objetivo principal lidar com os problemas relacionados à variabilidade do sinal das características biométricas mantendo níveis de segurança aceitáveis. Isto é, permitir que as chaves biométricas empregadas na encriptação e desencriptação sejam semelhantes, mas não idênticas, sem apresentar falhas de segurança que permitam que as informações protegidas sejam acessadas por indivíduos não autorizados.

Alguns criptossistemas biométricos foram propostos, no entanto, o modelo que tem sido mais investigado pela comunidade científica é o *Fuzzy Vault* (JUELS e SUDAN, 2002, 2006). O *Fuzzy Vault* é uma construção criptográfica e pode utilizar características biométricas como chave criptográficas, ou seja, ele pode ser implementado para diversos tipos de características biométricas (JUELS e SUDAN, 2006). O *Fuzzy Vault* foi implementado para impressões digitais (CLANCY et al., 2003), (ULUDAG e JAIN, 2004, 2006), (YANG e VERBAUWHEDE, 2004), face (WANG e PLATANIOTIS, 2007), íris (REDDY e BABU, 2008) e assinaturas (FREIRE-SANTOS et al. 2006). Este modelo de criptossistema biométrico e suas implementações serão apresentados em detalhes nas seções 2.7 e 2.8.

Antes do *Fuzzy Vault*, alguns modelos de criptossistemas biométricos foram propostos. No entanto, eles não foram abordados em trabalhos posteriores. Segue uma breve descrição desses criptossistemas.

Davida et al. (1998, 1999), propuseram um criptossistema biométrico que utiliza códigos correção de erros para tratar o problema da variabilidade do sinal biométrico na desencriptação. O método proposto faz uso de várias imagens de uma característica biométrica de um indivíduo para gerar um sinal modelo e então gera códigos que contém redundância, o que auxilia na desencriptação correta a partir de imagens que sejam semelhantes às imagens que formaram o modelo. Esta técnica é capaz de

armazenar um sinal biométrico de maneira segura. Os testes realizados com íris através do *IrisCode* (DAUGHMAN, 1993) assumiram que há diferenças de 10% dos valores entre os *IrisCodes* de um mesmo indivíduo. No entanto, em aplicações reais, essas diferenças podem chegar a 30%, tornando o sistema inviável.

Soutar et al. (1998, 1999) propuseram um sistema parecido com o de Davida et al. (1999) com diferença de aplicar a correção de erros através de transformadas de Fourier. Este modelo é composto por um algoritmo de nome Biometric Encryption e o bloco de dados protegido possui nome comercial de Bioscrypt^{TM}. O trabalho de Soutar et al. (1999) utiliza impressões digitais como característica biométrica. Este sistema não possui as mesmas deficiências que o de Davida et al. (1999). Apesar de ser mais eficiente esta técnica é considerada pouco segura (ULUDAG et al., 2004).

Monrose et al. (1999) propuseram um mecanismo que avalia a dinâmica de digitação em conjunto com a verificação de senhas. Segundo Uludag et al. (2004) a deficiência desse trabalho é que ele adiciona no máximo 15 bits de entropia em uma senha tornando-as apenas um pouco mais seguras. Monrose et al. (2001) também aplicou esse mecanismo com algumas modificações em características biométricas baseadas em voz.

A Tabela 4 apresenta uma comparação entre as propostas de criptossistemas biométricos realizada por Uludag et al. (2004), a comparação apresenta características de segurança, praticidade e aplicabilidade em situações reais (sensibilidade a variações) dos criptossistemas biométricos.

Tabela 4 – Características das propostas de criptossistemas biométricos (ULUDAG et al., 2004).

Algoritmo	Característica Biométrica	Proteção e Privacidade	Praticidade	Sensibilidade a Variações	Segurança
Soutar et al. (1999)	Impressão Digital (imagem)	Alta	Média	Alta	Indef.
Davida et al. (1999)	Íris (Iris Code)	Alta	Alta	Baixa	Indef.
Monrose et al. (2001)	Dinâmica de Digitação e Voz	Alta	Alta	Alta	Média
Linnartz e Tuyls (2003)	Sem avaliação	Alta	Baixa	Baixa	Alta
Juels e Sudan (2002)	Sem avaliação	Alta	Alta	Baixa	Alta
Clancy et al. (2003)	Impressão Digital (Minúcias)	Alta	Alta	Média	Alta

2.7. Fuzzy Vault

Juels e Sudan (2002, 2006) propuseram uma construção criptográfica chamada *Fuzzy Vault Scheme*, ou simplesmente *Fuzzy Vault* ou FVS. O *Fuzzy Vault* pode ser apresentado da seguinte maneira: Alice deseja conhecer pessoas na Internet que possuam o mesmo gosto musical que ela. Por isso, ela armazena o número de seu telefone encriptado em um conjunto de grupos musicais A de seus estilos favoritos. Bob pode ter acesso ao telefone encriptado com o conjunto A. No entanto, ele só é capaz de saber o número do telefone de Alice se utilizar um conjunto de grupos musicais B para desencriptar o número de telefone encriptado com o conjunto A, o conjunto B deve ser parecido com o A, mas não necessariamente idêntico.

O Fuzzy Vault é baseado no *Fuzzy Commitment Scheme* de Juels e Watemberg (1999), por isso, antes de se descrever o *Fuzzy Vault*, o *Fuzzy Commitment Scheme* será apresentado. O *Fuzzy Commitment Scheme* apresenta uma generalização e significativa melhora no trabalho de Davida et al. (1999). O trabalho de Davida et al. (1999) utiliza técnicas de correção de erros para desencriptar o sinal de impressões digitais armazenadas de maneira segura a partir de um sinal extraído, este sinal pode conter deformações que o fazem ser ligeiramente diferente do sinal armazenado. O *Fuzzy Commitment Scheme* propõe a utilização de correção de erros para reconstrução do sinal modelo da característica biométrica e o armazenamento do espalhamento de uma palavra para verificação da validade da reconstrução. Diferentemente do trabalho de Davida et al (1999)., o *Fuzzy Commitment Scheme* é genérico, ou seja, poder ser aplicado em diversos tipos de características biométricas.

Um esquema de combinação de bits convencional é um esquema no qual um indivíduo chamado remetente deseja enviar um bit b seguro para um segundo indivíduo conhecido como destinatário, o remetente então envia uma encriptação y de b para o destinatário. Um esquema de combinação de bits deve tornar impossível a um indivíduo diferente do remetente descobrir b a partir de y. O remetente também deve ser capaz de desencriptar a combinação y. Isto é, provar para o destinatário que y é a combinação de b. Intuitivamente, isto quer dizer que o remetente coloca b em um cofre e o entrega ao destinatário. Apenas o remetente pode abrir o cofre, pois somente ele conhece a combinação. Ele também não é capaz de alterar o valor contido no cofre y enquanto o mesmo está em poder do destinatário (DAVIDA et al., 1999).

Formalmente um esquema de combinação de bits consiste de uma função $F:\{0, 1\} \times X \rightarrow Y$. Para combinar um bit b, o remetente escolhe uma testemunha $x \in X$ geralmente uniformemente aleatória. O remetente então computa $y = F(b, x)$, onde y é

conhecido como um *blob* que representa o bit *b* protegido em um cofre. Para "abrir" ou "descombinar" o cofre *y*, o remetente então produz o bit *b* e a testemunha *x*. O *blob* é aberto se o destinatário se convence de que *y* representa uma encriptação de *b*. Um esquema de combinação de bits é tido como oculto se é impraticável para o destinatário descobrir b com probabilidade maior de ½. E ligado se é impraticável para o remetente "descombinar" o *blob y* com o bit incorreto, isto é *1 – b*.

Nota-se que é possível aplicar um esquema de combinação de bits a um esquema de combinação em uma cadeia de bits combinando cada bit independentemente. O termo esquema de combinação refere-se ao esquema de envolve a combinação de uma cadeia de bits *c* em um *blob* no qual é possível extrair *c* dado uma testemunha para o *blob* (JUELS e WATTENBERG 1999).

Assim, assume-se que $f: C \times X \rightarrow Y$, onde *C* é um espaço não binário. Além disso, este esquema permite a produção de uma testemunha *x* válida que habilita o valor *c* combinando ser eficientemente determinado de uma combinação *F(c, x)*. Este não é o caso dos esquemas de combinação convencionais, nos quais ambos *c* e a testemunha *x* são requeridos (JUELS e WATTENBERG 1999), isto é, no *Fuzzy Commitment Scheme*, para se desencriptar a combinação *F(c, x)* é necessário apenas a testemunha *x*, ao invés de *c* e *x* necessárias nos esquemas de combinação convencionais.

O interesse em se desenvolver um esquema de combinação *fuzzy F* é encontrar uma propriedade chamada incerteza (*fuzziness*). Assim, o esquema de combinação é tolerante a pequenas distorções no valor da testemunha. Com isso, o foco é poder "descombinar" o *blob y = F(c, x)* utilizando uma testemunha *x'* próxima a *x*, mas não necessariamente idêntica. Esta proximidade é determinada por alguma métrica, como, por exemplo, a distância de *Hamming* (JUELS e WATTENBERG 1999).

Em alguns casos, o processo de encriptação não pode ser utilizado, pois a mensagem não pode ser modificada, este é o caso da biometria. Na biometria a mensagem "suja" é a própria característica biométrica (JUELS e WATTENBERG 1999). Ou seja, não há sentido em mapeá-la para as palavras códigos, pois o sinal adquirido já representa a mensagem enviada pelo canal "sujo". Em função disso, o *Fuzzy Commitment Scheme* apenas utiliza a etapa de desencriptação de um esquema de combinação, isto é, ele não mapeia o sinal das características biométricas para a palavra código. Ele apenas utiliza os algoritmos de correção de erros para identificar as diferenças entre os sinais que serão comparados.

Observa-se que uma testemunha x com n bits pode ser expressa em função de uma palavra código combinada c com um *offset* $\delta \in \{0, 1\}^n$, tal que $x = c + \delta$ ou $\delta = x - c$, assim, δ provê informações parciais sobre x. A palavra c também precisa ser mantida em segredo, por isso, armazena-se δ e $h(c)$. Dessa forma, $y = (h(c), \delta)$. Não é possível identificar x caso n seja grande o suficiente (JUELS e WATTENBERG, 1999), caso n seja pequeno, um atacante pode descobrir valor de x pela força bruta realizando a operação $c' = x - \delta$ para todos os valores de x e compara-se $h(c)$ com $h(c')$ para se identificar qual é o x' gerado que gerou o c' correto.

Sendo $h : \{0,1\}^n \rightarrow \{0, 1\}^l$ uma função de espalhamento como por exemplo SHA-1 (NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, 1995) e $f:(\{0,1\}^n,\{0,1\}n) \rightarrow (\{0,1\}^l,\{0,1\}^n)$, assim $f(c, x) = (h(c), (x-c))$. Para desencriptar $f(c, x) = (\alpha,\delta)$ utilizando uma testemunha x' o destinatário computa $c' = f(x' - \delta) - f(c+(x-x'))$. Se $\alpha = h(c')$ então a desencriptação teve sucesso (JUELS e WATTENBERG 1999).

O *Fuzzy Commitment Scheme* introduz um conceito muito importante que é a tolerância a diferenças entre as testemunhas x e x'. Isto é essencial para criptossistemas biométricos, já que dificilmente consegue-se extrair dois sinais biométricos x e x'

exatamente iguais para um mesmo indivíduo. No entanto, esta técnica possui uma deficiência que é a necessidade se ter uma ordem invariável nos elementos da testemunha *x*, isto inviabiliza a utilização do *Fuzzy Commitment Scheme* com características biométricas. Por exemplo, para a utilização do *Fuzzy Commitment Scheme* com impressões digitais seria necessário que as minúcias da impressão digital modelo estivessem na mesma ordem das minúcias da impressão digital de consulta, na prática, isto torna o sistema inviável.

O *Fuzzy Commitment Scheme* possui deficiências, mas apresenta conceitos imporante e que foram empregados no *Fuzzy Vault* que é construído da seguinte maneira. Suponha-se que Alice deseja guardar um segredo *k* em um conjunto *A*. Ela seleciona um polinômio *p* de uma variável *x*, de modo que *k* seja representado por *p* (JUELS e SUDAN, 2006). Por exemplo, *k* pode ser representado através dos *n+1* coeficientes de um polinômio *p* de ordem *n*, onde cada coeficiente de *p* representa uma parte de *k*, a Figura 8 representa um possível polinômio *p*. Tomando os elementos do conjunto *A* como variáveis *x* distintas, Alice calcula as avaliações de *p* nos elementos de *A*, isto é, ela projeta os pontos *p(x)* obtendo coordenadas cartesianas *(x, p(x))*. Estes pontos são chamados verdadeiros. Assim, têm-se o conjunto *G* de pontos verdadeiros. Alice, então, gera um conjunto de pontos *(x, y)* aleatórios, os quais não se sobrepõem às projeções de *p(x)*. Estes pontos são chamados impostores e formam o conjunto *C*. A união dos conjuntos verdadeiros (*G*) e impostores (*C*) gera um terceiro conjunto *R* que representa o cofre no qual estão protegidos o segredo *k* e o conjunto *A*. A Figura 9 representa os pontos *G* e *C* no plano cartesiano, *C* é representado pelos quadrados vazio e *G* é representado pelos quadrados pintados. A Figura 10 representa o cofre *R* no plano cartesiano sem distinção entre os pontos *G* e *C*.

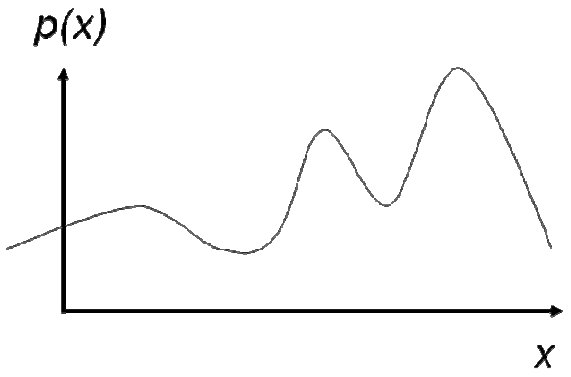

Figura 8 - Polinômio *p* que representa o segredo *k*.

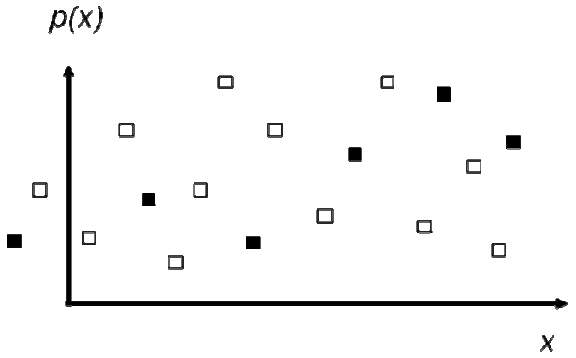

Figura 9 - Conjunto *G* de pontos verdadeiros e conjunto *C* de pontos impostores. *C* são os quadrados vazios e *G* são os quadrados pretos.

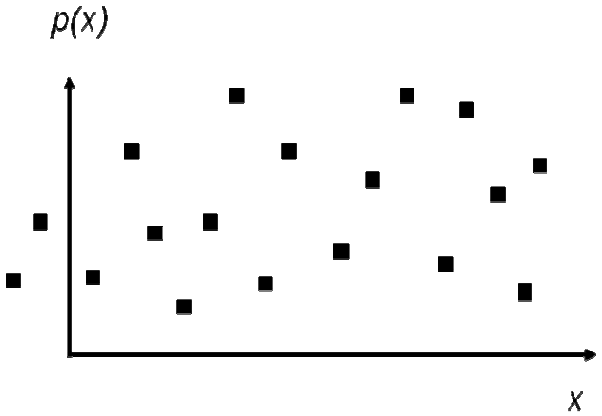

Figura 10 - Conjunto *R* com os pontos verdadeiros e impostores. *R* é o cofre que esconde *G* e *C*.

Para reconstruir *k*, deve-se detectar os valores *x* que representam o conjunto *A* ou boa parte dele, para então reconstruir o polinômio *p* e conseqüentemente *k*. O conjunto *R* representa a combinação de *k* e *A*. Um indivíduo que não possui conhecimento de *k* ou *A*, não consegue extraí-los do cofre, uma vez que é impraticável reconstruir o polinômio pela força bruta a partir de todos os valores *x* presentes em *R*. Desta forma, os pontos impostores *C* ocultam *G* no cofre *R*, provendo a segurança do sistema.

Supondo que Bob deseje abrir o cofre *R* utilizando um conjunto *B* de valores *x*. Se *B* sobrepõe substancialmente *A*, ou seja, vários pontos de *B* sejam próximos de *A*, Bob é capaz de reconstruir *p* e *k*. Caso haja pequenas diferenças entre *B* e *A*, elas podem ser corrigidas através do algoritmo de correção de erros empregado na desencriptação do cofre.

O algoritmo de correção de erros utilizado por Juels e Sudan (2002) é da classe dos *Reed-Solomon* (BERLEKAMP, 1968). Os algoritmos *Reed-Solomon* utilizam palavras código para encriptar e desencriptar uma mensagem. Essas palavras código são elementos de um campo finito de tamanho *t*. Isto é, um campo finito de tamanho *t* possui *t* palavras código. As palavras código do *Fuzzy Vault* são as avaliações de *p(x)*.

Na desencriptação, Juels e Sudan utilizam algoritmos *Reed-Solomon* lineares, pois eles representam uma desencriptação mais confiável para possíveis erros. O algoritmo utilizado por Juels e Sudan (2006) é o Peterson-Berlekamp-Massey (BERLEKAMP, 1968).

Juels e Sudan (2006) propuseram definições teóricas para o *Fuzzy Vault*, bem como uma demonstração teórica da segurança desta técnica. O trabalho não define um grau para o polinômio que será utilizado na construção do cofre e endereça apenas os problemas de segurança relacionados a possíveis implementações do *Fuzzy Vault*. De acordo com Juels e Sudan, a segurança do *Fuzzy Vault* está ligada a dois fatores: o

número de pontos impostores adicionados ao cofre e o número de palavras código presentes no campo utilizado, quanto maiores esses fatores, maior é a dificuldade de se abrir o cofre por força bruta.

Juels e Wattenberg (1999) propuseram uma técnica que apresenta uma generalização e significativa melhora no trabalho de Davida *et al.* (1999). Apesar de genérica, ou seja, poder ser aplicada em diversos tipos de características biométricas, esta técnica possui uma deficiência relacionada à invariabilidade na ordem dos valores de uma característica biométrica. Por não permitir variabilidade nos elementos do sinal de uma característica biométrica, o *Fuzzy Commitment Scheme* é muito difícil de ser aplicado em situações reais. O *Fuzzy Vault* soluciona o problema da não variação na ordem dos elementos do sinal das características biométricas, esta técnica permite que os conjuntos de dados utilizados para encriptação e desencriptação possuam diferenças entre eles, aspecto comum em características biométricas, sendo que o nível de aceitação dessas diferenças pode ser ajustado. O *Fuzzy Vault* também permite que os valores dos conjuntos utilizados para encriptação e desencriptação dos dados não estejam na mesma ordem. Ou seja, pode haver variação na ordem em que os elementos de um sinal são apresentados.

2.8. Fuzzy Vault para Impressões Digitais

Atualmente há algumas implementações do *Fuzzy Vault*, sendo que a maioria delas foi feita utilizando impressões digitais como característica biométrica e algumas para face, íris e assinatura escrita. Embora existam outras características biométricas, elas ainda não foram investigadas como alternativas de utilização no *Fuzzy Vault*.

Impressões digitais são as características biométricas mais antigas e mais utilizadas. As impressões digitais são universais, únicas para cada indivíduo e de fácil coletabilidade e possuem pouca variabilidade ao longo do tempo, razão do uso delas ser

amplamente difundido (WOODWARD et al. 2003). Impressão digital é a imagem formada pelas micro-elevações presentes nas polpas dos dedos das mãos. Essas elevações são formadas pelas papilas dos dedos. Quando os relevos têm uma elevação são chamados de cristas papilares e quando se assemelham a um vale são chamados de sulcos interpapilares. A Figura 11 apresenta duas imagens de impressão digital. As linhas escuras representam as cristas, enquanto linhas de cor clara formadas pelos os espaços entre as linhas escuras representam os sulcos.

Outras características presentes na impressão digital são as minúcias. As minúcias representam os inícios e terminações das cristas de uma impressão digital. As minúcias são representadas por um trio *(x, y, θ)* que correspondem à posição horizontal, a posição vertical e o ângulo da crista nas coordenadas da minúcia. Na Figura 11, em (b) a impressão digital apresenta as minúcias identificadas em vermelho.

As abordagens para reconhecimento de impressões digitais podem ser divididas em três famílias (MALTONI *et al.*, 2003):

- C**orrelação**: O reconhecimento baseado em correlação sobrepõe duas imagens de impressões digitais e calcula a correlação dos pixels para diferentes alinhamentos.
- M**inúcias**: O reconhecimento baseado em minúcias é o método mais conhecido e mais utilizado atualmente. As minúcias são extraídas de duas impressões digitais e os conjuntos de pontos são comparados com diferentes alinhamentos a fim de se encontrar o maior número de pares.
- C**ristas**: O reconhecimento baseado em cristas compara características extraídas dos padrões das cristas. As principais características das cristas são: tamanho, número, tipo e posição.

(a) **(b)**

Figura 11 - Imagens de Impressão Digital. (a) cristas e sulcos (b) Minúcias identificadas.

A primeira implementação do *Fuzzy Vault* para impressões digitais foi feita por Clancy et al. (2003), este trabalho implementou o *Fuzzy Vault* para impressões digitais utilizando valores das minúcias como as variáveis *x* para o polinômio *p*. O trabalho definiu uma ordem fixa para o polinômio *p* de ordem *k* no qual o segredo será escondido e utilizou coeficientes e variáveis de tamanho de 16 bits, dessa forma, o segredo a ser protegido deve possuir *16*(k +1)* bits de tamanho. Já as componentes das minúcias devem possuir 8 bits de tamanho, pois cada variável do polinômio é representada pela concatenação e das componentes *x* e *y* da minúcias, totalizando 16 bits. Ao invés de utilizar códigos de correção de erros na desencriptação do *Fuzzy Vault*, a implementação de Clancy *et al.* (2003) utiliza interpolação polinomial, a interpolação polinomial é feita com um subconjunto pontos candidatos de tamanho *k+1* por força bruta a partir do primeiro subconjunto até o subconjunto que desencripte com sucesso o *Fuzzy Vault*. Caso sejam desencriptados todos os subconjuntos sem sucesso, pontos

candidatos utilizados na desencriptação não foram capazes de desencriptar o *Fuzzy Vault*. Clancy et al. (2003) justificam que a utilização de interpolação polinomial com subconjuntos dos pontos candidatos é uma simplificação dos algoritmos de correção de erros e possui complexidade computacional menor. Esta implementação também foi importante, pois identificou restrições impostas pela implementação prática do *Fuzzy Vault*, como o limite de 69 bits de segurança para um polinômio de grau 14 com 38 pontos verdadeiros e 275 pontos impostores. No entanto, a implementação de Clancy *et al*.(2003) possui uma taxa de erro em torno de 20% a 30%, muito alta para aplicações reais e assume que os sinais extraídos das impressões digitais estejam pré-alinhados, o que não acontece em ambientes de aplicações reais.

Uludag e Jain (2004), propusram uma implementação do *Fuzzy Vault* para impressões digitais baseada na implementação de Clancy et al. (2003). A diferença entre as duas implementações é a representação das minúcias, a implementação de Uludag e Jain (2004) propõe outro mecanismo de representação de minúcias que evita o problema de pré-alinhamento da implementação de Clancy et al. (2003). A representação proposta por Uludag e Jain (2004) é uma variação da representação de minúcias baseadas em linhas (*line-based*) proposto por Malickas e Vitkus (1999).

Uludag et al. (2005) propuseram uma nova versão do *Fuzzy Vault* para impressões digitais, esta proposta considera a concatenação de código de verificação redundante (CRC) ao segredo como mecanismo para identificação de sucesso na desencriptação (ULUDAG et al., 2005). O código CRC é necessário pois a interpolação polinomial não possui um mecanismo de identificação de erros. Uludag et al. (2005) utilizaram o algoritmo de extração de características proposto por Jain et al. (1997) nesta implementação que considera um polinômio de grau 8, o qual compreende o segredo e o código CRC, totalizando 144 bits. Uludag et al. (2005) avaliaram o

desempenho de desencriptação do *Fuzzy Vault* em um banco com imagens de impressões digitais pré-alinhadas, isto é, as imagens do banco possuíam os parâmetros de rotação e translação já definidos. A implementação deste trabalho apresentou desempenho de precisão em torno de 80% de aceitação genuína (GAR) e 0% de falsa aceitação (FAR).

Uludag et al. (2005) utilizaram um mecanismo de mosaico para representação das minúcias, no mosaico os pontos genuínos são alinhados aos pontos médios de blocos de tamanho fixo, por exemplo, caso sejam usados blocos de tamanho 7, um ponto que conter coordenadas *(6, 13)*, seria enquadrado para *(4, 12)*. Esta abordagem tem por objetivo aumentar a tolerância do sistema à variabilidade do sinal biométrico, no entanto, ela pode ser considerada uma falha de segurança, uma vez que diminui em 64 vezes as possibilidades de coordenadas dos pontos genuínos. Caso sejam utilizadas imagens com tamanho 256x256 pixels a quantidade de coordenadas possíveis diminui de 65536 para 64.

Yang e Verbauwhede (2004), (2005) também propuseram uma implementação do *Fuzzy Vault* para impressões digitais. A proposta de Yang e Verbauwhede (2004) utilizou algoritmos de correção de erros na desencriptação do *Fuzzy Vault*, de acordo com o modelo proposto pode Juels e Sudan (2006). As minúcias da impressão digital, ao invés de serem representadas como coordenadas cartesianas são representadas por coordenadas polares a partir de um ponto de referência. No entanto, de acordo com Uludag e Jain (2006), o ponto de referência precisa ser sempre o mesmo, o que não acontece na prática e dificulta a utilização com sucesso desse modelo.

Uludag e Jain (2006) propuseram uma melhoria no *Fuzzy Vault* para impressões digital de Uludag et al. (2005) adicionando informações de alinhamento das impressões digitais. Estas informações são chamadas *Orientation Field Flow Curves (OFFC)*

(DASS e JAIN, 2004), elas são coeficientes que representam a orientação das linhas das impressões digitas e auxiliam no processo de alinhamento das impressões digitais de consulta durante a desencriptação. Uludag e Jain (2006) realizaram os experimentos utilizando a base de dados DB 2 FVC 2002 (MALTONI et al., 2003), esta base de dados é pública e representa diversas situações em que as impressões digitais podem ser adquiridas. Com o alinhamento a partir das informações dos coeficientes, Uludag e Jain (2006) conseguiram 72.6% de sucesso na desencriptação do *Fuzzy Vault* utilizando uma imagem na desencriptação e 84,5% de sucesso utilizando duas imagens na desencriptação.

O trabalho de Uludag e Jain (2006) foi o primeiro que apresentou resultados em uma base de dados pública com impressões digitais adquiridas em diversas situações e sem alinhamento prévio. As OFFC empregadas por Uludag e Jain (2006) não revelam informações sobre os valores encriptados no cofre mantendo os níveis de segurança da informação protegida.

Nagar e Chaudhury (2006) apresentaram uma proposta de modificação no *Fuzzy Vault*. Esta implementação não utiliza projeção polinomial, ela encripta um segredo e uma característica biométrica de tamanho *n* em duas matrizes de tamanho 3 x *n* (*GridC* e *GridB*). Cada elemento do segredo é disposto em uma linha da matriz *GridC* em uma coluna aleatória e cada elemento da característica biométrica é disposto nas linha da matriz *GridB* nas colunas respectivas de *GridC*, sendo as outras posições da matriz preenchida com números aleatórios (impostores). A Figura 12 apresenta o *Fuzzy Vault* proposta por Nagar e Chaudhury (2006). Define-se um limiar de tolerância para identificação dos elementos na desencriptação e o valor desse limiar define a flexibilidade e segurança da implementação, quanto maior o limiar o sistema, este se torna mais flexível e menos seguro (maior GAR e FAR). Nagar e Chaudhury (2006)

propuseram um sistema que encripta um documento com uma chave RSA e encripta essa chave usando a implementação do *Fuzzy Vault* proposta. Nagar e Chaudhury (2006) analisaram a implementação com 29 impressões digitais de 9 indivíduos e conseguiram GAR 0% e FAR entre 0% e 19,4% dependendo do valor de limiar. Esses resultados foram obtidos em uma base de dados com poucas imagens.

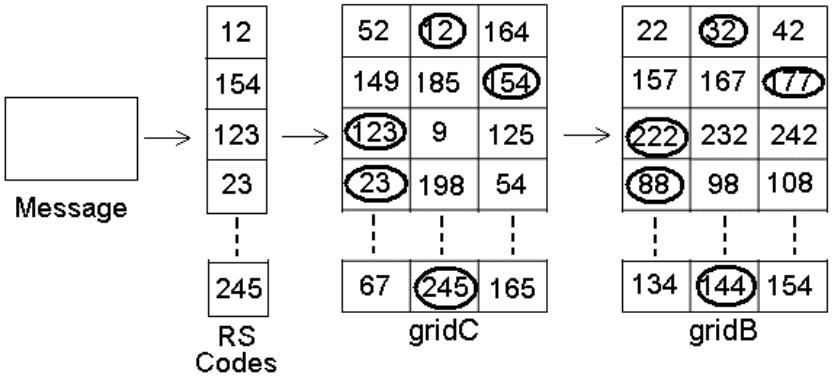

Figura 12 – *Fuzzy Vault* proposto por Nagar e Chaudhury (2006).

Outra proposta de implementação do *Fuzzy Vault* para impressões digitais (NANDAKUMAR, NAGAR e JAIN, 2007) utilizou senhas para aumentar a segurança dos dados protegidos. Ela se baseou no trabalho de Uludag e Jain (2006) e criou duas camadas de proteção: criação de uma impressão digital cancelável a partir de uma transformação usando a senha e encriptação criptográfica do cofre com uma chave gerada a partir da senha. Para desencriptar o cofre é necessário possuir a impressão digital e a senha, caso alguém tenha acesso somente à senha poderá acessar o cofre, mas não é capaz de desencriptá-lo e caso alguém tenha somente acesso à impressão digital, é necessário conhecer a senha para acessar o cofre e aplicar a transformação da impressão digital de consulta para se gerar uma impressão digital semelhante à transformada na encriptação. O trabalho de Nandakumar, Nagar e Jain (2007) analisou o desempenho de implementação proposta com a base de dados FVC2002-DB2 (MAIO, 2002) e os resultados ficaram em torno de 5% piores, se comparados à implementação sem senha, que obteve em torno de 90% de taxa de aceitação de genuínos (GAR).

A maioria dos trabalhos encontrados na literatura avaliaram as implementações utilizando bases de dados proprietárias, estas bases não estão disponíveis para outros pesquisadores e dificultam a reprodução dos experimentos. Dessa forma, não é possível estabelecer comparações diretas entre os resultados do presente estudo e os relatados pela literatura.

Nota-se que é necessário implementar métodos de alinhamento das impressões digitais, e não se pode assumir que as impressões digitais estarão pré-alinhadas. Os trabalhos que implementaram alinhamento conseguiram resultados entre 85% e 90% de sucesso quando avaliados em bases de dados de tamanho médio (ULUDAG e JAIN, 2006), (NANDAKUMAR, NAGAR e JAIN, 2007). O único trabalho encontrado que propôs a integração do *Fuzzy Vault* com outros sistemas criptográficos foi o de Nagar e Chaudhury (2006), isto evidencia que há poucas pesquisas focadas em analisar o *Fuzzy Vault* integrado a outras aplicações.

3. IMPLEMENTAÇÃO DO FUZZY VAULT PARA IMPRESSÕES DIGITAIS

Este capítulo descreve uma implementação do *Fuzzy Vault* para impressões digitais e apresenta a avaliação de desempenho da implementação proposta. Esta implementação tem por objetivo realizar a desencriptação do *Fuzzy Vault* com a precisão e eficiência necessárias para uma aplicação real, além de apresentar segurança contra ataques aos dados protegidos. A encriptação encripta um segredo em conjunto com as minúcias da impressão digital modelo e extrai informações de alinhamento da impressão digital modelo. A desencriptação alinha a impressão digital de consulta e tenta desencriptar o segredo encriptado utilizando as minúcias da impressão digital de consulta.

3.1. Proposta de implementação do Fuzzy Vault para impressões digitais

A implementação proposta realiza a encriptação e desencriptação do Fuzzy Vault para impressões digitais. O algoritmo de encriptação recebe como parâmetros de entrada um segredo e a impressão digital modelo e encripta o segredo e as minúcias da impressão digital, retornando um cofre com os dados encriptados. Além disso, o algoritmo extrai informações de alinhamento da impressão digital modelo, estas informações são utilizadas na desencriptação para alinhamento da impressão digital de consulta. A Figura 13 apresenta um diagrama do algoritmo de encriptação.

A encriptação contém os seguintes passos:

1. **Encriptação CRC**: Gera-se um código de checagem de redundância cíclica (CRC) a partir do segredo. O código CRC é anexado ao segredo gerando a cadeia que será encriptada, esta cadeia possui nome *k*. O código CRC é necessário no processo de desencriptação, a partir dele será possível verificar se o segredo *k* foi desencriptado corretamente. No presente estudo, o segredo possui tamanho 128 bits e o código CRC possui tamanho 16 bits, dessa forma, a cadeia *k* possui 144 bits.
2. **Construção do polinômio**: Constrói-se um polinômio *p* a partir da cadeia *k* gerada na etapa 1. Divide-se a cadeia *p* em *N* elementos, assim, o polinômio *p* possui ordem $O=N-1$. Cada coeficiente do polinômio corresponde a um elemento da cadeia *k*, ou seja, $p(u) = k_{N-1}u^{N-1} + k_{N-2}u^{N-2} + k_{N-3}u^{N-3} + ... + k_1u + k_0$. No presente estudo o polinômio *p* possui ordem $O=8$, isto é, *p* possui $N=9$ coeficientes, sendo que, cada coeficiente de *p* representa 16 bits da cadeia *k*.
3. **Extração das minúcias da impressão digital**: As minúcias da impressão digital modelo são extraídas e geram o conjunto *A* que será utilizado na encriptação do *Fuzzy Vault*. Cada minúcia é composta por três valores *x'*, *y'* e *Θ'* que representam a posição vertical, horizontal e o ângulo da minúcia. Na encriptação do *Fuzzy Vault* são utilizadas apenas as componentes *x'* e *y'*, descartando-se a componente *Θ'*. As componentes *x'*e *y'* são concatenadas na forma *x'*|*y'*. Considerando que sejam extraídas *L* minúcias $A=\{(x'_0|y'_0), (x'_1|y'_1), (x'_2|y'_2), ..., (x'_{L-1}|y'_{L-1})\} = \{u_0, u_1, u_2, ..., u_{L-1}\}$.

4. **Projeção polinomial**: Nesta etapa são gerados os pontos genuínos do cofre, o conjunto dos pontos genuínos recebe nome *G*. Os pontos genuínos são compostos pela projeção dos elementos de *A* em *p*. Dessa forma, $G = \{(u_0, p(u_0)), (u_1, p(u_1)), (u_2, p(u_2)), ..., (u_{L-1}, p(u_{L-1}))\}$, ou seja, o elemento *i* do conjunto *G* corresponde a $(u_i, p(u_i)) = (x'_i|y'_i, p(x'_i|y'_i))$. O conjunto *G* contém a encriptação das minúcias da impressão digital modelo e do segredo.
5. **Geração dos pontos impostores**: Nesta etapa são gerados pontos aleatórios chamados pontos impostores. Os pontos impostores representam pontos de minúcias falsos que serão unidos com o conjunto *G* para que não seja possível identificar os elementos de *G* sem conhecer que foram encriptadas. O conjunto de pontos impostores recebe nome *C* e possui tamanho *Q*, dessa forma, O conjunto é $C = \{(c_0, d_0), (c_1, d_1), (c_2, d_2), ..., (c_{Q-1}, d_{Q-1})\}$. Os elementos de C não devem coincidir com os elementos de G e nem coincidir com projeções do polinômio *p*. Isto é, $c_j \neq u_i$, $j=1,2,...Q$, $u_i, i=1,2,..,L$ e $d_j \neq p(c_j)$, $j=1,2,...,Q$.
6. **União dos conjuntos**: Os pontos dos conjuntos *G* e *C* são unidos em um único conjunto *V*, este conjunto representa o cofre que protege o segredo e a impressão digital. O conjunto *V* possui tamanho $VS = L+Q$. Isto é, $V = \{(u_0, p(u_0)), (u_1, p(u_1)), (u_2, p(u_2)), ..., (u_{L-1}, p(u_{L-1})),(c_0, d_0), (c_1, d_1), (c_2, d_2), ..., (c_{Q-1}, d_{Q-1})\}$
7. **Mistura dos pontos**: Nesta etapa os elementos do conjunto *V* trocam de posição aleatoriamente para se evitar que os elementos do conjunto *G* fiquem concentrados no início do conjunto. Assim, $V = \{(v_0, w_0), (v_1, w_1), (v_2, w_2), ..., (v_{VS-1}, w_{VS-1})\}$
8. **Extração das informações de alinhamento**: São extraídas as informações de alinhamento da impressão digital modelo, estas informações serão anexadas ao cofre e serão utilizadas no alinhamento da impressão digital de consulta. Esta etapa será descrita na seção 3.2.

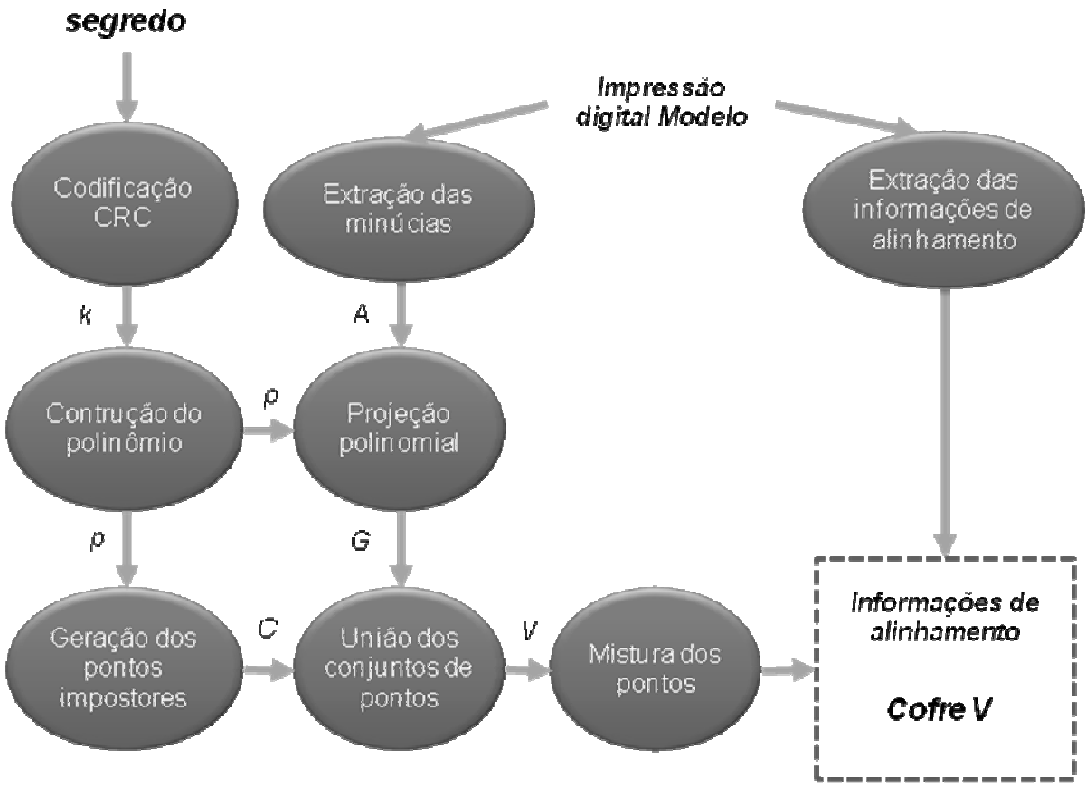

Figura 13 - Diagrama do processo de encriptação da implementação proposta.

A desencriptação tenta recuperar o segredo e a informação biométrica protegidos no cofre *V* utilizando uma impressão digital de consulta. Se a impressão de consulta for suficientemente similar a impressão modelo, o algoritmo de desencriptação desencripta o segredo. Para isso, o algoritmo de desencriptação recebe como parâmetros de entrada uma impressão digital de consulta, o cofre *V* e as informações de alinhamento. O algoritmo retorna o segredo desencriptado ou um código de erro na desencriptação.Primeiro, a desencriptação alinha a impressão digital de consulta e identifica os pontos do cofre *V* que estão próximos às minúcias de da impressão digital de consulta. A partir desses pontos são realizadas sucessivas interpolações de Lagrange que geram polinômios *k'* de tamanho *N*, para cada polinômio é feita a verificação CRC para se identificar o sucesso da desencriptação. A Figura 14 apresenta um diagrama do processo de desencriptação da implementação proposta.

Os passos do algoritmo de desencriptação são os seguintes:

1. **Extração das minúcias de consulta**: As minúcias são extraídas da impressão digital de consulta. As minúcias extraídas geram o conjunto *B* com tamanho *R*. Isto é, $Q = \{(x''_0, y''_0), (x''_1, y''_1), (x''_2, y''_2), ..., (x''_{R-1}, y''_{R-1})\}$.
2. **Extração das informações de alinhamento**: Extrai-se as informações de alinhamento da impressão digital de consulta. Esta etapa é descrita em detalhes na seção 3.2.
3. **Alinhamento da impressão digital**: Alinha-se a impressão digital de consulta de acordo com o método proposto na seção 3.2
4. **Identificação dos pontos candidatos:** Nesta etapa são identificados os pontos candidatos para a desencriptação. Considerando *V* o cofre que contém o segredo e a impressão digital encriptados e que o segredo encriptado em *V* possui tamanho *VS*. Primeiramente é gerado um conjunto *VQ* com as a componentes *v* dos elementos do cofre *V*. Os elementos *v* de *VQ* são divididos em duas componentes (*x'*, *y'*), sendo que *x'* é a metade superior de *v* e *y'* é a metade inferior de *v*. Isto é, $v_i = (x'_i, y'_i)$, $i=1,2,..,VS$. Sendo assim $VQ = \{(x'_0, y'_0), (x'_1, y'_1), (x'_2, y'_2), ..., (x'_{VS-1}, y'_{VS-1})\}$ As componentes *x'* e *y'* representam as coordenadas das minúcias encriptadas, tanto as genuínas quanto as impostoras. Para cada elemento de *Q*, calcula-se a distância euclidiana entre ele e os elementos de *VQ* e identifica-se elemento com menor distância. Caso essa distância seja menor do que a distância limite *t* o elemento de *V* correspondente ao elemento de *VQ* é marcado como candidato, caso o contrário, a minúcia é descartada. Dessa forma, é gerado um conjunto de *VC* com os *S* pontos candidatos identificados, sendo $VC = \{(u'_0, v'_0), (u'_1, v'_1), (u'_2, v'_2), ..., (u'_{S-1}, v'_{S-1})\}$. O conjunto *VC* representa os elementos de *V* que correspondem aos elementos de *Q*, ou seja, os pontos candidatos.

5. **Combinação dos pontos Candidatos**: A partir do conjunto *VC* gera-se uma combinação *c* com *N* pontos, sendo que *N* é o tamanho da cadeia *k* que gerou o polinômio *p* de grau *O=N-1*. Essa combinação será utilizada para reconstrução do polinômio através da interpolação polinomial de Lagrange. Caso não existam mais combinações possíveis, o algoritmo retorna um código de erro na desencriptação.
6. **Interpolação de Lagrange**: Nesta etapa é realizada a interpolação de Lagrange na combinação *c* para se reconstruir um polinômio *p'* e, conseqüentemente, a cadeia *k'* que pode representar o segredo.
7. **Desencriptação CRC**: Nesta etapa é realizada a desencriptação CRC, a desencriptação CRC é feita aplicando o CRC nos *N-1* elementos da cadeia *k'*. O resultado do CRC é comparado com o elemento *N* da cadeia *k'*, caso os valores sejam idênticos, $p = p'$ e $k = k'$, ou seja, o polinômio reconstruído é igual ao polinômio usado na encriptação e o segredo é desencriptado. Caso os valores sejam diferentes, volta-se para a etapa 5 a fim de se encontrar outra cadeia que possa desencriptar o segredo corretamente.

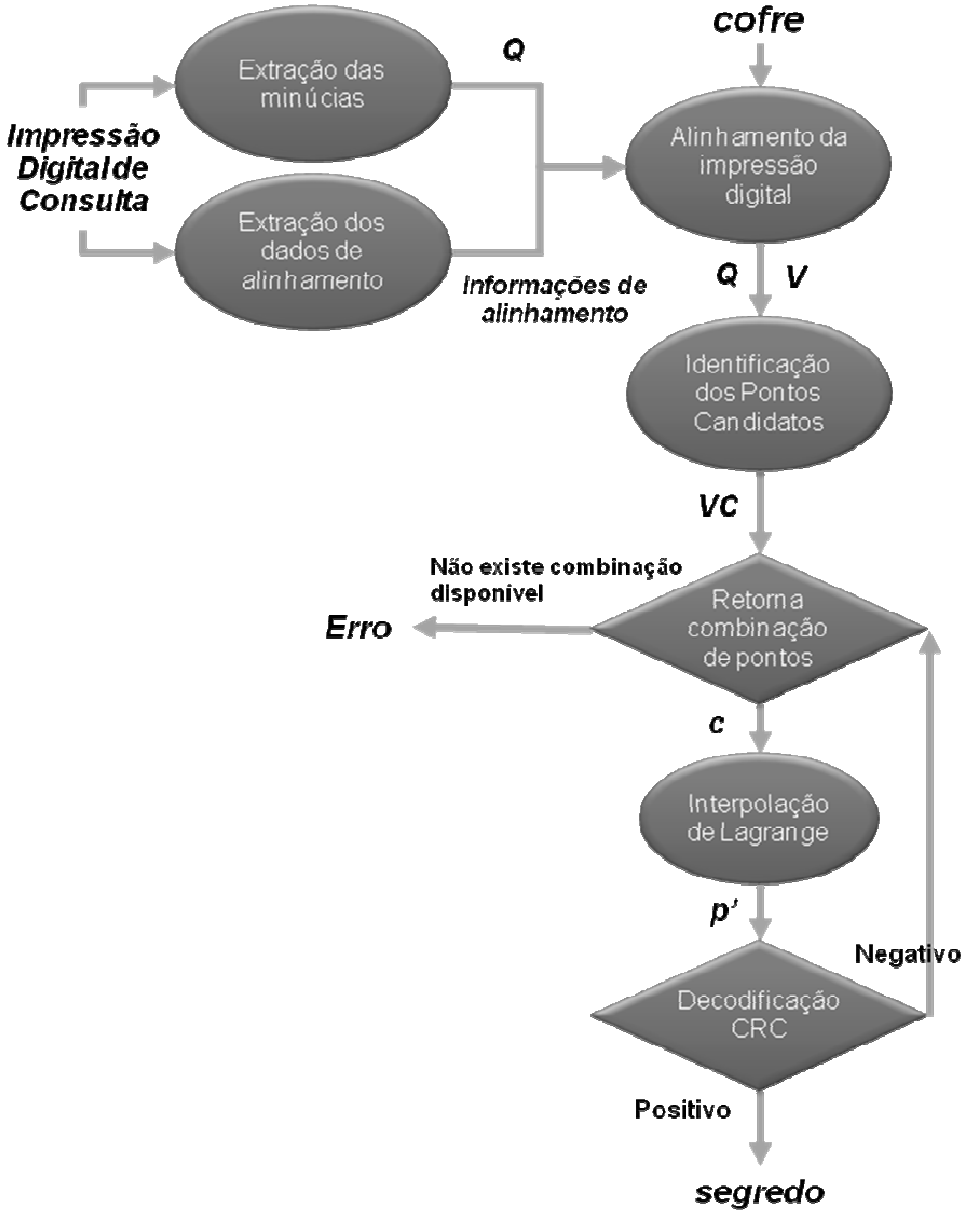

Figura 14 - Diagrama do processo de desencriptação da implementação proposta

3.2. Alinhamento da Imagem de Consulta

Os principais problemas apresentados na encriptação de impressões digitais são relacionados ao alinhamento das imagens modelo e de consulta. As aquisições podem apresentar rotações e translações diferentes e é necessário estabelecer medidas de comparação para que a impressão digital de consulta apresente a mesma rotação e translação da impressão digital modelo. No entanto, essas medidas de alinhamento não devem revelar nenhuma informação sobre as minúcias, caso o contrário, elas representariam uma falha de segurança no algoritmo.

Uludag e Jain (2006) propuseram uma melhoria no *Fuzzy Vault* para impressões adicionando informações de alinhamento das minúcias. Estas informações são chamadas *Orientation Field Flow Curves (OFFC)* (DASS e JAIN 2004), elas são coeficientes que representam a orientação das linhas das impressões digitas e auxiliam no processo de alinhamento das minúcias durante a desencriptação. As OFFC empregadas por Uludag e Jain (2006) não revelam informações sobre os valores encriptados no cofre mantendo os níveis de segurança da informação protegida. No entanto, há outras maneiras de alinhar as impressões digitais.

3.2.1. Reconhecimento de Impressões Digitais Baseado em Cristas

O presente estudo utilizou o algoritmo de reconhecimento de cristas proposto por Marana e Jain (2005) para realizar o alinhamento das impressões digitais. Marana e Jain (2005) apresentaram um método para verificação de similaridade entre impressões digitais baseado em cristas, este método aplica a transformada de Hough para extrair linhas retas das cristas e alinhar duas impressões digitais. As retas são extraídas da impressão digital modelo e da impressão digital consulta. Este método três etapas principais: extração das informações, alinhamento e casamento das impressões digitais

3.2.1.1. Extração das Informações da Impressão Digital

A extração das informações da impressão digital detecta as cristas da impressão digital, aplica a transformada de Hough nas cristas para gerar linhas retas que serão utilizadas no alinhamento e classifica as cristas de acordo com sua curvatura. Esta etapa consiste dos passos s seguir:

1. **Extração das cristas:** utiliza-se o algoritmo de detecção de cristas do método proposto por Jain et al. (1997) para retornar uma lista contendo as cristas da impressão digital afinadas.

2. **Extração das Retas que Passam pelas Cristas:** este passo detecta as retas que passam pelos pixels das cristas. Na Figura 15 é possível observar uma crista previamente afinada com as suas retas detectadas.

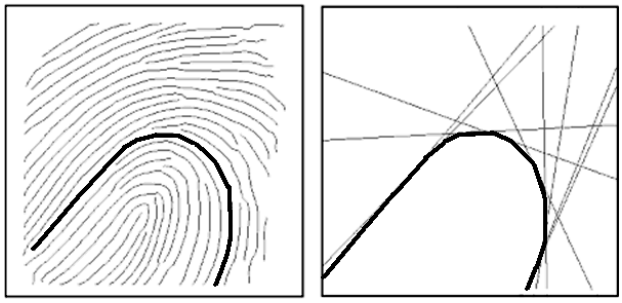

Figura 15: Extração das cristas e das linhas. À esquerda, as cristas detectadas; À direita, as retas que passam pela crista em destaque (MARANA e JAIN, 2005).

A transformada de Hough (*TH*) é aplicada em cada crista separadamente para detectar as retas. A *TH* permite a detecção de curvas que são facilmente parametrizadas, por exemplo, retas, círculos e elipses. No caso de retas, a equação $\rho = x\cos\theta + y\,sen\theta$ é geralmente utilizada, onde ρ é a distância da reta à origem e θ a orientação do vetor normal à reta. A Figura 16 apresenta os parâmetros da equação da reta e o mapeamento de dois pontos pertencentes à reta do domínio espacial para o domínio de Hough. Utilizando uma matriz acumuladora *HS*, o procedimento de Hough examina cada pixel

de uma dada crista e incrementa os elementos *HS (r, t)* que correspondem a todas as retas que passam por este pixel, onde *r* e *t* são os valores quantizados de ρ e θ, respectivamente.

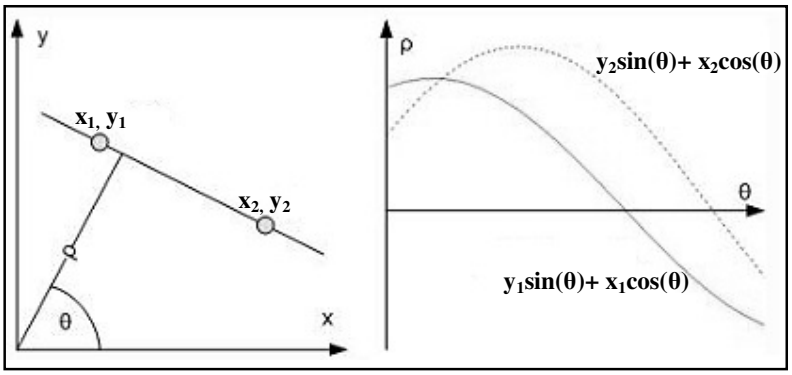

Figura 16: Mapeamento de pontos de uma reta do domínio espacial para o domínio da transformada de Hough (MARANA e JAIN, 2005).

3. **Detecção de picos do Espaço de Hough:** depois que todos os pixels de uma determinada crista são processados, é realizada uma busca no acumulador e um limiar é utilizado para indicar os picos (valores máximos no acumulador). Os picos do Espaço de Hough da impressão digital, caracterizados pelo trio (θi, ρi, *vi*), onde *vi* (valor do pico) é o número de pontos colineares da crista que se encontram sobre a reta definida por θi e ρi, são armazenados e utilizados para identificar as retas que passam por aquela crista.

4. **Classificação das Cristas:** as retas (picos do Espaço de Hough) detectadas são utilizadas para classificar as cristas em categorias de curvaturas, sendo que na categoria 1 a crista é praticamente reta e na categoria 5 a crista é quase circular.

3.2.1.2. Alinhamento

Nesta etapa os parâmetros de alinhamento (transformações) são estimados e a impressão digital de entrada é alinhada com a impressão digital modelo de acordo com essas transformações. Para estimar esses parâmetros são utilizados os picos do Espaço

de Hough da impressão modelo (*St*) e de consulta (*Sq*) caracterizadas pelo trio (θi, ρi, *vi*), calculados na etapa de extração de características.

A rotação pode ser facilmente estimada utilizando um espaço de parâmetros unidimensional, onde, para cada par de picos (*qi*, *ti*), com $qi \in Sq$ e $ti \in St$, a célula do acumulador de rotação $R(\theta ti\text{-}\theta qi)$ é incrementada. Na Figura 17 pode-se observar como as retas que passam pelas cristas são utilizadas no cálculo da rotação.

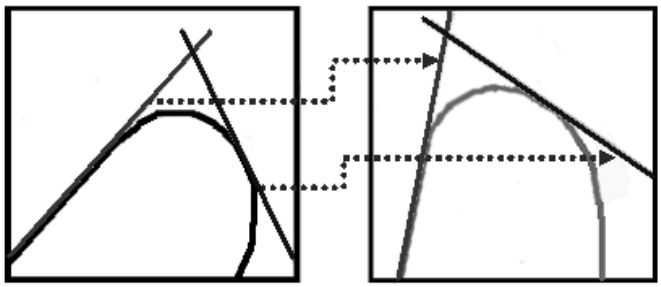

Figura 17: Cálculo da rotação baseado nos picos do espaço de Hough (MARANA e JAIN, 2005).

A translação, por sua vez, requer um espaço de parâmetros bidimensional, *TR*, onde cada par de picos de *Sq* da mesma crista (par de retas que passam pela crista) é rotacionado de acordo com o ângulo de rotação calculado no passo anterior. Então, o ponto de intersecção, *pq*, das duas retas correspondentes a esse pico é computado. O mesmo procedimento é realizado para cada par de picos de *St*, onde *pt* é o ponto de intersecção das duas retas que correspondem a esses picos. A célula do acumulador de translação *TR (pty-pqy, ptx-pqx)* é, então, incrementada por um peso baseado no tamanho máximo das retas utilizadas para encontrar os pontos de intersecção *pq* e *pt*. A Figura 18 apresenta um exemplo de ponto de interseção de duas retas que passam pela mesma crista.

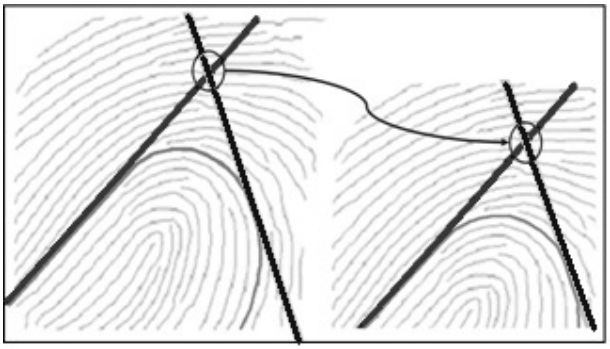

Figura 18: Pontos de intersecção entre duas retas que passam pela mesma crista utilizados para cálculo da translação (MARANA e JAIN, 2005).

Os espaços dos parâmetros *R* e *TR* acumulam evidências sobre os parâmetros mais prováveis de rotação e translação, respectivamente. Um limiar é, então, utilizado e apenas os parâmetros maiores que esse limiar são considerados como possíveis parâmetros de alinhamento. A Figura 19 apresenta um exemplo de alinhamento obtido para duas impressões digitais de um determinado indivíduo.

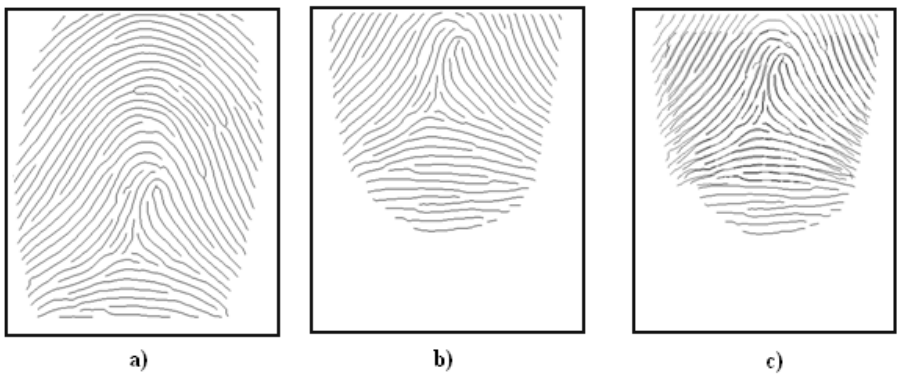

Figura 19: a) Impressão digital de consulta; b) Impressão digital modelo e c) Alinhamento da impressão digital de consulta (MARANA e JAIN, 2005).

3.2.1.3. Casamento

Para cada trio ($\Delta\theta$, Δx, Δy) calculado na etapa de alinhamento, alinha-se a impressão digital de consulta à impressão digital modelo e a pontuação de casamento é calculada tendo como base o número de cristas que casaram entre essas duas imagens. Para isso, uma matriz de alinhamento de cristas, $C_{m,n}$, é computada, onde *m* e *n* são os

números de cristas detectadas na imagem de consulta e na imagem modelo, respectivamente. Sendo assim, o elemento *(i,j)* da matriz *C* indica quantos pixels da crista *i* da impressão digital de consulta coincidem com os pixels da crista *j* da impressão digital modelo (MARANA e JAIN, 2005).

Uma característica importante dessa matriz é que se a imagem de consulta e modelo forem a mesma, *C* é uma matriz diagonal, onde o elemento *k* da diagonal é exatamente o número de pixels da crista *k*. Por isso, para um alinhamento genuíno, é esperado que *C* tenha os valores mais altos na diagonal principal, enquanto que em um alinhamento impostor, é esperado que *C* tenha valores baixos e espalhados por toda matriz (MARANA e JAIN, 2005). A pontuação de casamento é, então, calculada a partir da matriz *C*. A Figura 20 apresenta duas matrizes de alinhamentos de cristas, para casamento genuíno e impostor.

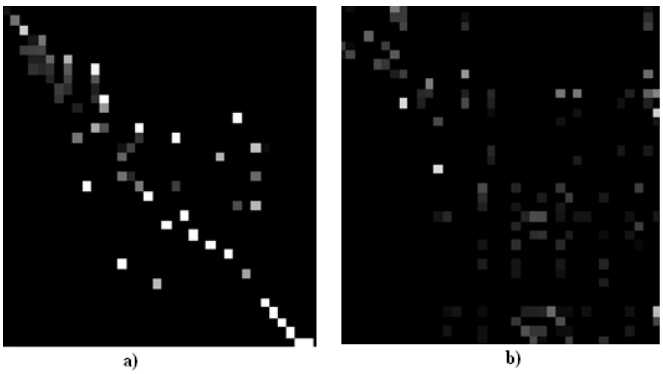

Figura 20: Matrizes de alinhamento de cristas. a) Casamento genuíno; b) Casamento impostor (MARANA e JAIN, 2005).

3.2.2. Proposta de Alinhamento da imagem de consulta no Fuzzy Vault para impressões digitais

A implementação proposta no presente estudo utiliza as etapas de extração de características e alinhamento da técnica de Marana e Jain (2005) para estimar os possíveis trios ($\Delta\theta$, Δx, Δy) de alinhamento da impressão digital de consulta. Após estimar os trios de alinhamento, identifica-se o melhor trio e alinha-se a impressão digital de consulta utilizando suas minúcias e os pontos de cofre que protege a impressão digital modelo.

O alinhamento proposto possui uma etapa executada na encriptação e outra etapa executada na desencriptação. Na encriptação, as retas da impressão digital modelo são extraídas para serem utilizadas no alinhamento da desencriptação. Na desencriptação extraem-se as retas da impressão digital de consulta e estimam-se os parâmetros de alinhamento a partir das retas das impressões digitais modelo e de consulta. Alinham-se as minúcias de consulta com os diferentes parâmetros de transformação (trios) e faz-se um casamento entre as minúcias de consulta e os pontos do cofre para se identificar o melhor trio de parâmetros de alinhamento. A Figura 21 apresenta os passos do algoritmo de alinhamento da impressão digital de consulta. O alinhamento possui os seguintes passos:

1. **Extração das linhas retas da impressão digital de consulta**: Extraem-se as linhas retas que serão utilizadas na estimativa dos trios de parâmetros de alinhamento.
2. **Estimativa dos parâmetros alinhamento**: São estimados os trios ($\Delta\theta$, Δx, Δy) de alinhamento de acordo com a técnica proposta por Marana e Jain (2005).

3. **Cálculo da pontuação para os trios de alinhamento estimados**: Para cada trio de alinhamento estimado em 2 aplica-se a transformação nas minúcias da impressão digital e calcula-se uma pontuação alinhamento do trio. A transformação corresponde à rotação da minúcia no ângulo $\Delta\theta$ e à translação em Δx e Δy. Calcula-se a pontuação de alinhamento de um trio somando as pontuações das minúcias de consultas alinhadas com esse trio. Cada minúcia alinhada recebe uma pontuação que corresponde a um coeficiente *t* dividido pela distância euclidiana entre a minúcia e o ponto do cofre com menor distância da minúcia (*t/(dist + 1)).* A pontuação total do trio de alinhamento é a soma das pontuações de todas as minúcias alinhadas usando os parâmetros do trio. O trio de alinhamento com maior pontuação é o trio que melhor alinha a imagem de consulta.
4. **Alinhamento da imagem de consulta**: Nesta etapa a imagem de consulta é alinhada utilizando o melhor trio de alinhamento ($\Delta\theta$, Δx, Δy) identificado na etapa 3.

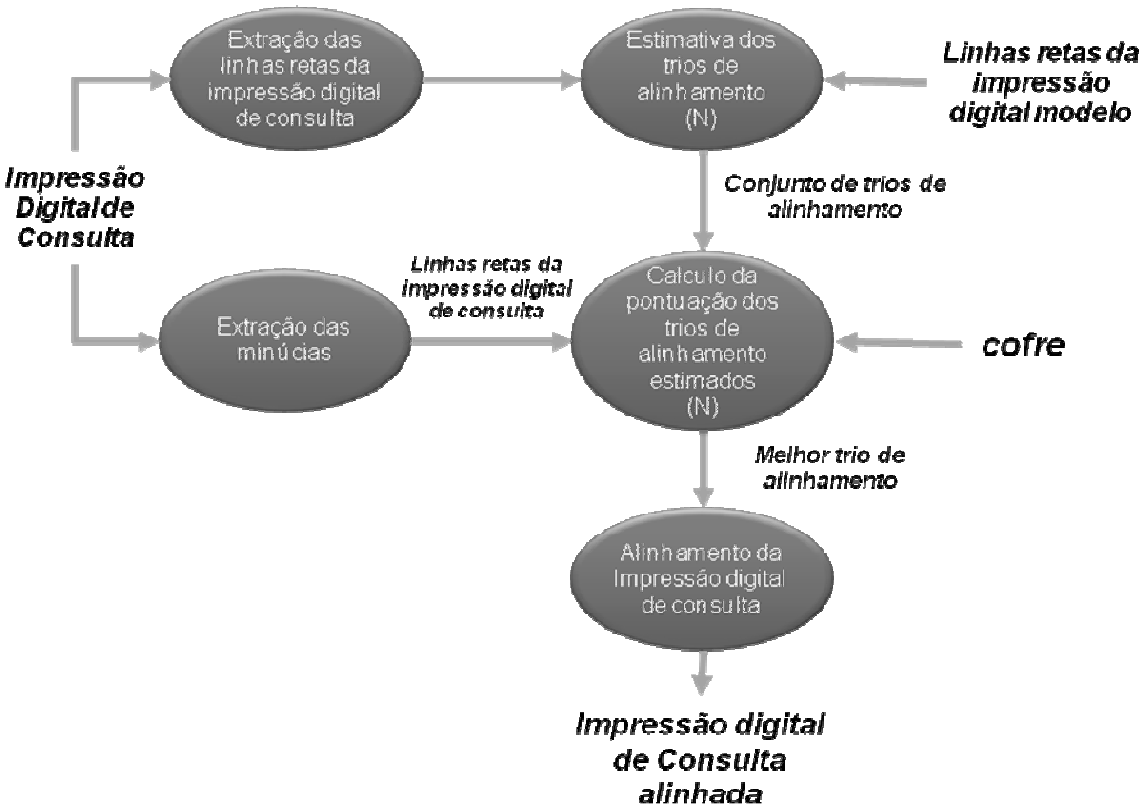

Figura 21 – Passos do algoritmo de alinhamento da impressão digital de consulta.

As cristas das impressões digitais não são usadas diretamente, são utilizadas apenas suas linhas retas. A utilização direta das cristas representaria uma falha de segurança do sistema, pois a impressão digital pode ser reconstruída a partir das cristas. Já as linhas retas não revelam qualquer informação sobre o início e fim das cristas os quais representam as minúcias. Dessa forma, as linhas retas anexadas ao cofre não representam uma falha de segurança no sistema proposto. O presente estudo utiliza as minúcias da impressão digital de consulta e os pontos do cofre encriptado no cálculo da pontuação de similaridade em vez das cristas que foram utilizadas por Marana e Jain (2005).

3.3. Material

A implementação do *Fuzzy Vault* para impressões digitais utilizou os seguintes parâmetros, o cofre foi encriptado com 200 pontos impostores e a quantidade de pontos genuínos utilizada correspondeu à quantidade de minúcias encontradas na impressão digital modelo. Na desencriptação a quantidade de minúcias utilizadas correspondeu à quantidade de minúcias da impressão digital de consulta. Impressões digitais com menos de 9 minúcias foram descartadas. O valor do coeficiente *t* utilizado no alinhamento foi 12 (coeficiente divido pela distância euclidiana entre a minúcia e o ponto mais próximo no cofre). O limiar para identificação de pontos candidatos estabelecido foi 6 (distância euclidiana entre a minúcia e o ponto mais próximo no cofre).

A avaliação utilizou imagens da base de dados FVC 2002 DB1 (MAIO, 2002). Esse banco de dados possui oito imagens de impressões digitais para cada um dos 100 indivíduos, totalizando 800 imagens, a Figura 22 apresenta algumas imagens da base de dados. O banco de dados foi capturado utilizando-se um coletor ótico modelo Biometrika FX2000, sendo que a coleta foi feita em três seções diferentes, com intervalos de duas semanas entre cada seção. Nenhum esforço para controlar a qualidade das imagens foi realizado. As imagens do banco possuem tamanho original de 560x296 pixels e foram redimensionadas para 256x256 pixels para que as coordenadas da imagem fossem representadas por oito bits cada.

A aplicação foi desenvolvida na linguagem de programação C++ no ambiente de desenvolvimento *Microsoft Visual Studio .NET 2005*. As rotinas de extração das minúcias das impressões digitais estão implementadas na biblioteca *Griaule GRFinger 4.2*. As rotinas que implementam o algoritmo AES e o algoritmo RSA são as rotinas presentes no *Microsoft .NET Framework 2.0*.

O desenvolvimento e a avaliação foram realizados utilizando dois computadores equipados com processador Intel Core 2 Duo T5250, memória RAM de 1 GB e disco rígido SATA de 300 GB.

Figura 22 – Impressões digitais da base de dados DB2 FVC 2002 (MAIO et al., 2002), cada linha apresenta as impressões digitais de um mesmo indivíduo.

3.4. Método

Foram realizados experimentos para avaliar o desempenho da implementação proposta no capítulo 3. Avaliou-se a precisão da desencriptação e o tempo de execução da implementação proposta.

A o desempenho do *Fuzzy Vault* foi avaliado executando-se a encriptação e desencriptação nos seguintes experimentos:

- **Experimentos para detecção da taxa de aceitação genuína**: executou-se a encriptação para cada impressão digital presente no banco e executou-se a desencriptação com as impressões restantes do mesmo indivíduo totalizando 5600 tentativas.
- **Experimentos para detecção da taxa de falsa aceitação**: executou-se a encriptação com uma impressão digital de cada individuo e a desencriptação com 100 impressões digitais dos indivíduos restantes totalizando 10000 tentativas.

Em todas as execuções as seguintes variáveis foram capturadas: quantidade de minúcias extraídas na encriptação, quantidade de minúcias extraídas na desencriptação, quantidade de pontos candidatos identificados, número de combinações realizadas na interpolação de Lagrange, tempo de encriptação, tempo de alinhamento, tempo de desencriptação e sucesso na desencriptação.

A precisão da desencriptação foi avaliada pela *GAR* e *FAR*. O valor de *GAR* foi obtido nos experimentos para detecção da taxa de aceitação genuína, dividiu-se o a quantidade de reconhecimentos genuínos pelo total de tentativas válidas, dessa forma, obteve-se a proporção de reconhecimentos genuínos em função do total de tentativas. A FAR foi obtida nos experimentos para detecção da taxa de falsa aceitação, dividiu-se a quantidade de falsos reconhecimentos em função do total de tentativas válidas, assim, obteve-se a proporção de reconhecimentos falsos em função do total de tentativas.

Analisou-se a distribuição da proporção de sucessos em função da quantidade de minúcias genuínas extraídas na encriptação, quantidade de minúcias extraídas na desencriptação e quantidade de pontos candidatos identificados.

O desempenho de tempo foi analisado em função do tempo de encriptação, de alinhamento e de desencriptação. Na desencriptação analisou-se o tempo em função da quantidade de pontos candidatos identificados para as desencriptações com sucesso e fracasso.

3.5. Resultados

A análise considerou 5534 tentativas de desencriptação, em 66 tentativas (1,18%), as impressões digitais de consulta apresentaram menos de nove minúcias e foram descartadas por não possuírem os requisitos mínimos para a desencriptação.

Os resultados dos experimentos para avaliação da precisão de desencriptação são apresentados na Tabela 5 e na

Tabela 6. Em todos os casos as Taxa de Falsa Aceitação (*FAR*) foi de 0%. A Taxa de Aceitação Genuína (*GAR*) considerando-se todas as tentativas foi de 73,31%.

Tabela 5 – Freqüência e porcentagem de desencriptações positivas e negativas.

Resultado da Desencriptação	Freqüência	Porcentagem
Positivo	4057	73,31%
Negativo	1477	26,69%
Total	5534	100 %

Tabela 6 – Porcentagens de desencriptações positivas para cada imagem e função da imagem de encriptação e desencriptação.

		Imagem de Desencriptação								
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Todas**
Imagem de Encriptação	**1**		93,00%	88,00%	60,20%	67,68%	77,00%	77,78%	78,00%	77,44%
	2	97,96%		89,00%	67,35%	69,00%	75,76%	83,00%	89,80%	81,67%
	3	88,78%	88,00%		64,95%	59,79%	68,00%	75,76%	86,87%	76,09%
	4	61,22%	67,01%	60,20%		48,42%	52,04%	56,00%	60,82%	57,98%
	5	67,35%	66,00%	62,63%	47,47%		62,24%	70,10%	68,00%	63,39%
	6	76,77%	81,82%	75,76%	53,61%	61,00%		78,00%	79,00%	72,33%
	7	78,57%	82,00%	74,75%	58,16%	68,04%	83,00%		92,00%	76,73%
	8	78,79%	91,00%	86,00%	62,24%	72,45%	83,00%	90,00%		80,58%

Considerando-se apenas uma impressão digital de cada indivíduo para encriptação e desencriptação, a Taxa de aceitação Genuína (*GAR*) para se seguintes casos foi:

- encriptação com Imagem 1 e desencriptação com imagem 2: 93,00% de GAR;
- encriptação com Imagem 2 e desencriptação com imagem 1: 97,96% de GAR;
- encriptação com Imagem 7 e desencriptação com imagem 8: 92% de GAR;

Em 1477 tentativas (26,69%) houve erro na tentativa de desencriptação. Entre essas tentativas, em 320 não foi possível identificar ao menos 9 pontos candidatos, em 1073 tentativas foram identificados entre 9 e 16 pontos candidatos e nas 84 tentativas restantes foram identificados mais de 16 pontos candidatos.

A Figura 23 apresenta a porcentagem de acerto nas desencriptações em função da quantidade de minúcias da impressão digital de consulta. Nota-se que a porcentagem de acerto aumenta de acordo com o número de minúcias. Com 30 minúcias, as desencriptações com sucesso são em torno de 79,73% de um total de 1401 tentativas. A porcentagem de acertos aumenta para entre 85% e 95% quando a quantidade de minúcias de consulta está entre 31 e 42 minúcias em um total de 1389 tentativas. Essa taxa se mantém no mesmo patamar quando as minúcias de consulta passam de 42, no entanto, a quantidade de desencriptações nessas condições diminui consideravelmente (235).

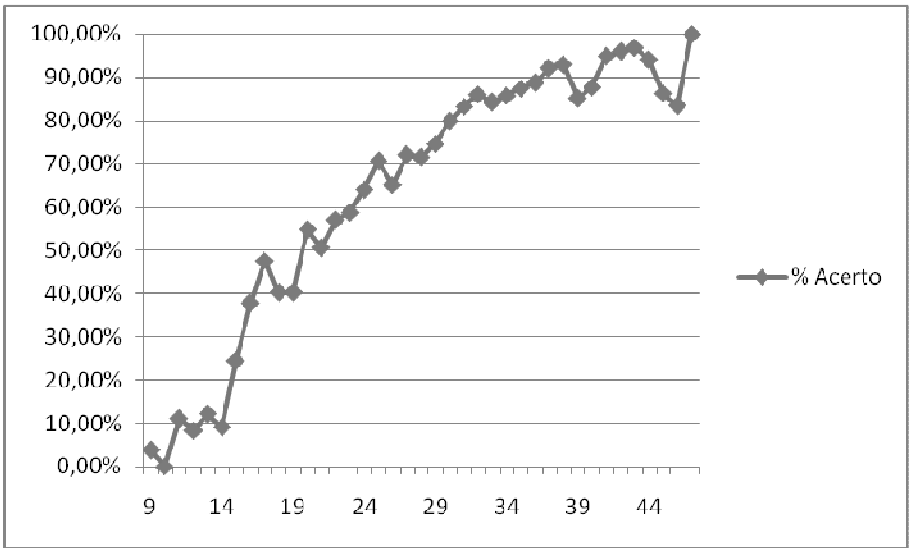

Figura 23 – Distribuição dos acertos nas desencriptações em função da quantidade de minúcias da impressão digital de consulta.

Analisando-se a distribuição das encriptações e desencriptações em função da quantidade de minúcias e pontos candidatos (Figura 24, Figura 25 e Figura 26), pode-se notar um comportamento parecido. Há uma grande quantidade de impressões digitais com em torno de 30 minúcias e menos impressões digitais com menos de 20 ou mais de 40 minúcias.

A Figura 24 apresenta a porcentagem de encriptações em função do número de minúcias da impressão digital modelo. Nota-se um pico (1177 encriptações, ou 21,27%) nas encriptações que utilizaram impressões digitais com 30 minúcias. Já nas extremidades, existiram poucas encriptações, isto é, em poucas situações houve encriptações com impressões digitais com menos de 20 ou com mais de 40 minúcias.

Nota-se o mesmo comportamento nas desencriptações em função das minúcias da impressão digital de consulta (Figura 26), com pico em 30 minúcias (1401, ou 25,32%) e nas desencriptações em função da quantidade pontos candidatos identificados (Figura 25), com pico em 20 pontos (414 encriptações, ou 7,48%). Enquanto as desencriptações em função dos pontos candidatos apresentam uma distribuição mais

dispersa, as desencriptações em função das minúcias da impressão digital modelo e de consulta apresentam picos nas impressões digitais com em torno de 30 minúcias.

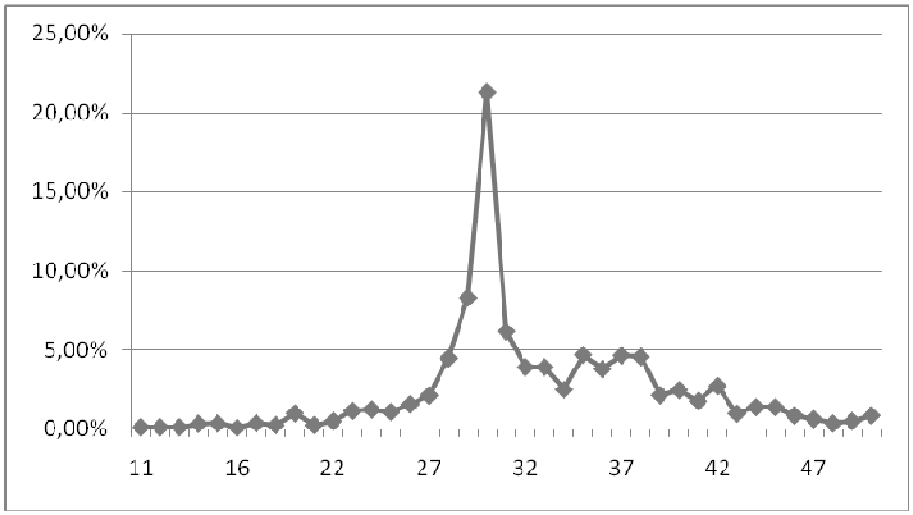

Figura 24 – Distribuição das encriptações em função da quantidade de minúcias da impressão digital modelo.

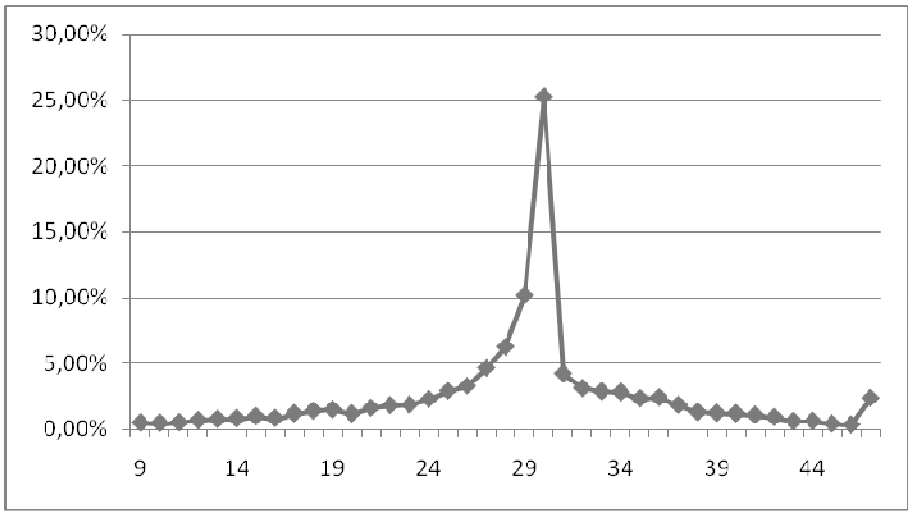

Figura 25 – Distribuição das desencriptações em função da quantidade de minúcias da impressão digital de consulta.

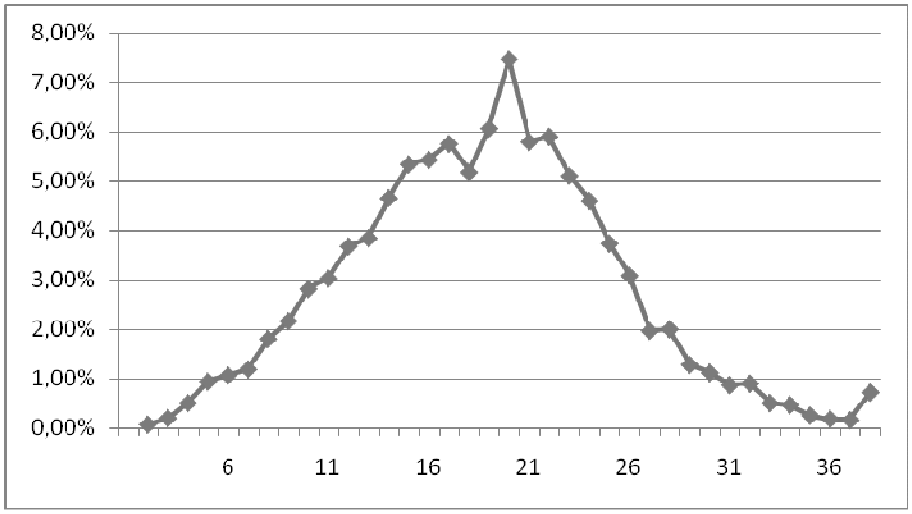

Figura 26 – Distribuição das desencriptações em função da quantidade de pontos candidatos.

A avaliação de eficiência da implementação proposta neste estudo analisou o tempo das etapas de encriptação e alinhamento pelo tempo médio das execuções. O tempo de desencriptação foi analisado em função da mediana, pois foi encontrada uma grande variabilidade no tempo aferido nas execuções.

O experimento de avaliação de desempenho de eficiência da encriptação apresentou tempo de 0,8 segundos, já o tempo do alinhamento das minúcias foi de 3,35 segundos. O tempo médio de desencriptação foi 14,49 segundos com desvio padrão de 44,48 segundos. A Tabela 7 apresenta a média, desvio padrão, mínimo, máximo dos tempos capturados nas etapas de encriptação, alinhamento e desencriptação.

A Figura 27 apresenta um gráfico com o tempo de desencriptação em função da quantidade de pontos candidatos nas desencriptações corretas. Nota-se picos nas desencriptações que utilizam entre 14 e 20 pontos candidatos, acima de 20 pontos candidatos com tempo entre 16 e 30 milissegundos.

A Figura 28 apresenta um gráfico com o tempo de desencriptação em função da quantidade de pontos candidatos. Há um aumento exponencial do tempo de desencriptação em função da quantidade de pontos.

Tabela 7 – Tempo médio das etapas de encriptação, desencriptação e alinhamento.

	Tempo (s)			
	Média	**Desvio Padrão**	**Mínimo**	**Máximo**
Encriptação	0,80	0,13	0,39	1,39
Alinhamento	3,35	0,91	0,45	6,06
Desencriptação	14,49	44,88	0,00	773,97

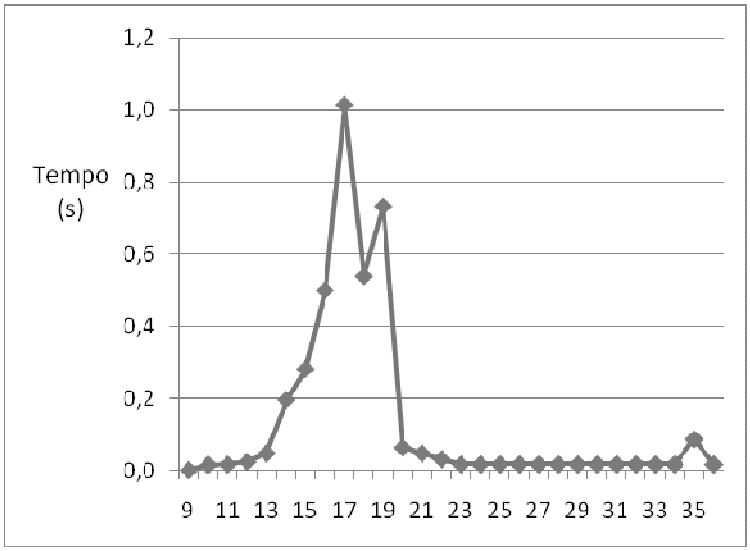

Figura 27 – Tempo desencriptação em função da quantidade pontos candidatos (desencriptações corretas).

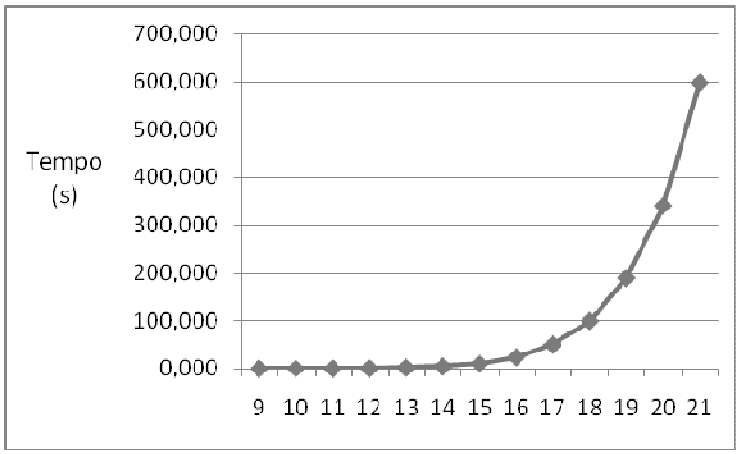

Figura 28 – Tempo de desencriptação em função da quantidade de pontos candidatos (desencriptações erradas).

3.6. Discussões e Conclusões

Com base nos resultados obtidos pode-se concluir que a implementação proposta apresenta resultados parecidos com as implementações propostas por Clancy et al. (2003) e abaixo da implementação proposta por Uludag e Jain (2006) e por Nandakumar, Nagar e Jain (2007). No entanto, as implementações de Uludag e Jain (2006) e de Nandakumar, Nagar e Jain (2007) consideraram apenas uma imagem por indivíduo para encriptação e desencriptação. Usando essa mesma metodologia a implementação do presente estudo apresenta GAR entre 92% e 97,96% contra 72,6% e 84,5% do trabalho de Uludag e Jain (2006) e 90% do trabalho de Nandakumar, Nagar e Jain (2007). Nota-se que as implementações da literatura apresentam resultados em situações específicas das bases de dados.

O bom desempenho de desencriptação da implementação proposta neste estudo pode ser atribuído ao método de alinhamento empregado e ao método de identificação de pontos candidatos que permitiram identificar em torno de 20 pontos candidatos nas desencriptações. As linhas retas utilizadas no alinhamento das impressões digitais, não revelam nenhuma informação relativa às minúcias, dessa forma, elas não representam uma falha de segurança no sistema

Nota-se que a maioria das tentativas com sucesso utilizou-se impressões digitais modelo e de consulta com entre 30 e 40 minúcias, no entanto, na maioria das tentativas identificou-se em torno de 20 pontos candidatos. São indícios de que uma parte das minúcias da impressão digital modelo não é encontrada na desencriptação.

Também nota-se que houve poucas tentativas com sucesso que utilizaram impressões digitais modelo e de consulta com menos de 20 ou mais de 40 minúcias, já a distribuição dos pontos candidatos foi mais dispersa. Isto mostra que em uma parte das desencriptações foram identificados menos pontos candidatos do que quantidade de

minúcias da impressão digital modelo, e em outra parte pontos impostores foram identificados como sendo pontos candidatos. Também conclui-se que se forem utilizadas impressões digitais com em torno de 30 minúcias, as chances da desencriptação ocorrer com sucesso aumentam consideravelmente.

Os experimentos de avaliação de eficiência revelam que a encriptação apresentou tempo baixo, em torno de 0,80 segundos com baixo valor de desvio padrão. Isso demonstra que há pouca variabilidade no processo de encriptação. O tempo do alinhamento das minúcias foi de 3,35 segundos, esse tempo é resultado das sucessivas rotações e translações realizadas no conjunto de minúcias de consulta para identificação dos melhores parâmetros de alinhamento e, apesar de constante, representa um gargalo de desempenho na implementação.

Analisou-se a eficiência da desencriptação em função da quantidade de pontos candidatos identificados. Nas desencriptações corretas, o tempo se apresentou com valor entre 16 e 33 milissegundos quando foram utilizados entre 31 e 40 pontos candidatos e picos de entre 0,3 e 1 segundo nas desencriptações com entre 16 e 20 pontos candidatos. Nas desencriptações erradas, o tempo de desencriptação aumenta exponencialmente na medida em que se aumenta o número de minúcias candidatas encontradas.

Este comportamento se deve à quantidade de interpolações de Lagrange necessárias para se encontrar o segredo. Nas desencriptações corretas o algoritmo pode interpolar o polinômio correto logo nas primeiras combinações de pontos, enquanto nas desencriptações erradas, é necessário se interpolar todas as combinações possíveis. As desencriptações corretas se beneficiam de uma característica do algoritmo que inicia as iterações das interpolações com os pontos mais próximos das minúcias consulta.

4. PROTEÇÃO DE IMAGENS MÉDICAS USANDO BIOMETRIA

Nenhuma das propostas da literatura procurou avaliar o *Fuzzy Vault* em aplicações reais, isto é, verificar as sua viabilidade quando integrado a outras aplicações. Este capítulo apresenta um estudo de caso para a aplicação do *Fuzzy Vault* na proteção de imagens médicas. Neste estudo, uma imagem médica é encriptada com um criptossistema tradicional e a chave do criptossistema é protegida com o *Fuzzy Vault*.

4.1. Proteção de Imagens Médicas

Imagens médicas podem ser armazenadas e manipuladas em formato digital através do Sistema PACS (*Picture Archiving and Communication System*) (HUANG, 1999). PACS é um sistema integrado de gerenciamento para arquivamento e distribuição de imagens médicas (HUANG, 1999). Imagens armazenadas no PACS devem ser privativas e disponibilizadas apenas ao médico e ao paciente. Em função disso, essas imagens devem ser manipulas de maneira segura. Isto é, elas devem ser armazenadas e transmitidas com segurança para protegê-las de acessos não autorizados.

Atualmente, existem padrões que definem a manipulação segura de imagens médicas. Um padrão é o DICOM (*Digital Image and Communication in Medicine*), outro é o HIPAA (*Health Insurance Portability and Accountability Act*), não foi encontrado na literatura um padrão brasileiro para segurança de informações médicas.

O padrão DICOM é mantido pelo Conselho Americano de Radiologia (ACR) e pela Associação Nacional de Produtores e Elétricos (NEMA) e tem por objetivo assegurar a interoperabilidade de sistemas usados para produzir, armazenar, apresentar,

processar, enviar, recuperar e imprimir imagens médicas (NATIONAL, 2008). A parte 15 do Padrão DICOM (PS 3.15-2007) apresenta um modelo para comunicação segura e assinatura digital de imagens médicas (NATIONAL, 2008), são definidos mecanismos que podem ser utilizados na implementação de políticas de segurança relacionadas ao intercâmbio de objetos DICOM entre entidades de aplicações. O padrão DICOM define quatro perfis de segurança: perfil de uso seguro, perfil de conexão de transporte seguro, perfil de assinatura digital e perfil de armazenamento seguro de mídia.

O padrão HIPAA (US DEPARTMENT OF HEALTH AND HUMAN SERVICES, 2003) é uma lei federal dos Estados Unidos que apresenta um *framework* conceitual para segurança e integridade de dados na saúde e define penas federais significantes para aqueles que não atendam ao padrão. No entanto, os guias não determinam soluções específicas, ao contrário, eles enfatizam a necessidade de soluções apropriadas para a variedade de cenários clínicos cobertos pela linguagem HIPAA. Atualmente o HIPAA endereça quarto áreas chave: transações eletrônicas, privacidade, identificadores únicos e segurança.

Uma maneira de se proteger imagens médicas é armazenar e transmita-las criptografadas. Imagens médicas podem ser encriptadas para armazenamento e transmissão e desencriptadas posteriormente pelo médico ou paciente. Além disso, é possível verificar a autenticidade de uma imagem médica enviada através de uma rede pública utilizando esquemas de assinatura digital.

Trabalhos recentes aplicaram criptografia para proteção de imagens médicas. Cao et al. (2003) utilizaram criptografia de chave pública para proteger informações médicas e verificar sua autenticidade. A autenticidade é verificada através de assinatura digital onde o remetente da imagem gera uma assinatura da imagem médica com a sua chave privada e pode-se verificar a autenticidade da imagem médica com a chave

pública do remetente. A segurança da imagem é provida pela encriptação e assinatura da imagem médica e assinatura com a chave pública do destinatário. Assim, a imagem só pode ser desencriptada pelo destinatário, o dono da chave privada. O trabalho de Cao et al. (2003) apresenta um método para segurança de imagens médicas, mas o método proposto possui alguns problemas de desempenho e não aborda a proteção da chave privada do destinatário.

Outros trabalhos abordam problemas de desempenho em imagens médicas (NORCEN, et al., 2003) e verificação de autenticidade de imagens médicas (SMITH, 1995).

4.2. Proteção de Imagens Médicas Usando Biometria

O método proposto para o cenário de proteção de imagens médicas utiliza criptografia de chave privada para encriptar uma imagem médica e protege a chave privada com informações biométricas, evitando acesso não autorizado à chave. Este método encripta uma imagem médica com um criptossistema tradicional (AES) e usa a implementação do *Fuzzy Vault* para impressões digitais proposta no capítulo 3 para encriptar a chave criptográfica usada na encriptação AES. Assim, é possível transmitir e armazenar com segurança tanto a imagem médica quanto a chave AES. Neste método, apenas a pessoa que possui a mesma impressão digital usada na encriptação pode desencriptar a imagem.

O cenário de proteção de imagens médicas usando biometria é composto por duas fases: encriptação e desencriptação. A fase de encriptação encripta uma imagem médica com o criptossistema AES, após a encriptação da imagem a utiliza-se o *Fuzzy Vault* para proteger a chave criptográfica e uma impressão digital modelo. Dessa forma, tanto a imagem médica quanto a chave criptográfica usada na encriptação só podem ser acessadas pelo indivíduo ao qual pertence a impressão digital modelo. A fase de

desencriptação recupera a imagem encriptada após desencriptar o *Fuzzy Vault* com a impressão digital de consulta e extrair a chave criptográfica usada na encriptação.

A fase de encriptação é composta pelos seguintes passos (a **Figura 29** apresenta um diagrama com os passos do processo de encriptação):

1. **Geração da chave AES**: Uma chave AES com tamanho 128 bits é gerada para encriptar a imagem médica;
2. **Encriptação da Imagem médica**: A imagem médica é encriptada pelo algoritmo AES;
3. **Encriptação *Fuzzy Vault***: A chave AES é protegida pelo *Fuzzy Vault* usando uma impressão digital modelo. Esta impressão digital deve pertence ao indivíduo que encriptou a imagem, por exemplo, um médico ou o paciente.

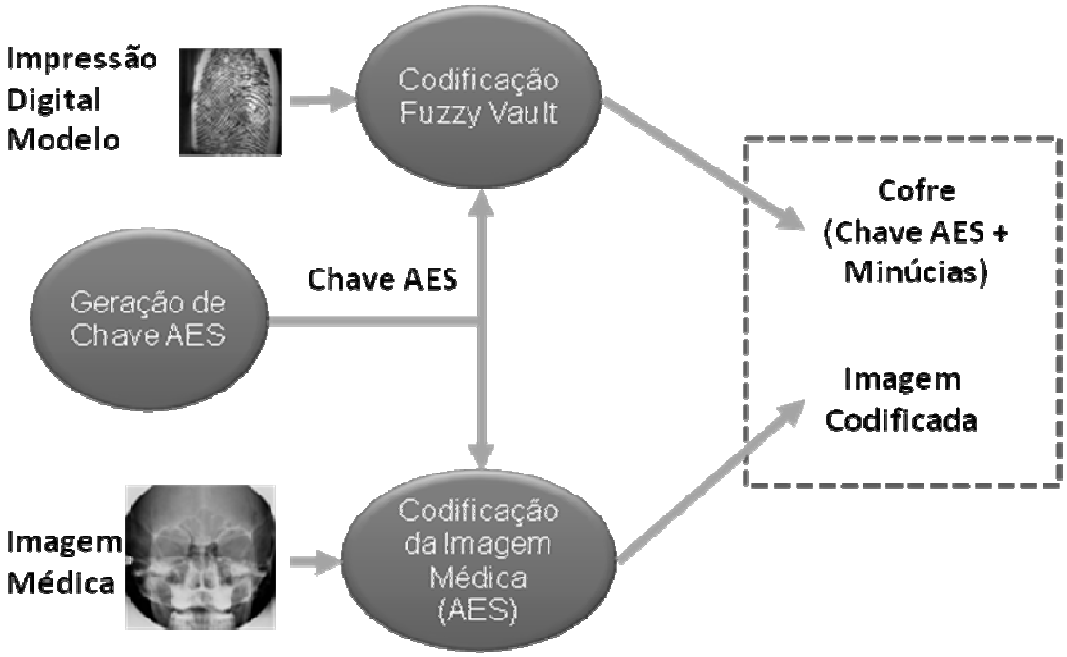

Figura 29 - Algoritmo de encriptação do cenário de proteção de imagens médicas usando biometria.

A fase de desencriptação possui os seguintes passos (a **Figura 30** apresenta um diagrama com os passos do processo de desencriptação):

1. **Desencriptação *Fuzzy Vault***: A chave AES e a informação biométrica são extraídas do *Fuzzy Vault* a partir da impressão digital de consulta. Se a impressão digital de consulta for suficientemente igual à impressão digital modelo, o *Fuzzy Vault* libera a chave AES. Caso o contrário, ele retorna um erro;
2. **Desencriptação da imagem médica**: Caso a chave AES seja liberada, a imagem encriptada é desencriptada e o indivíduo consegue visualizá-la.

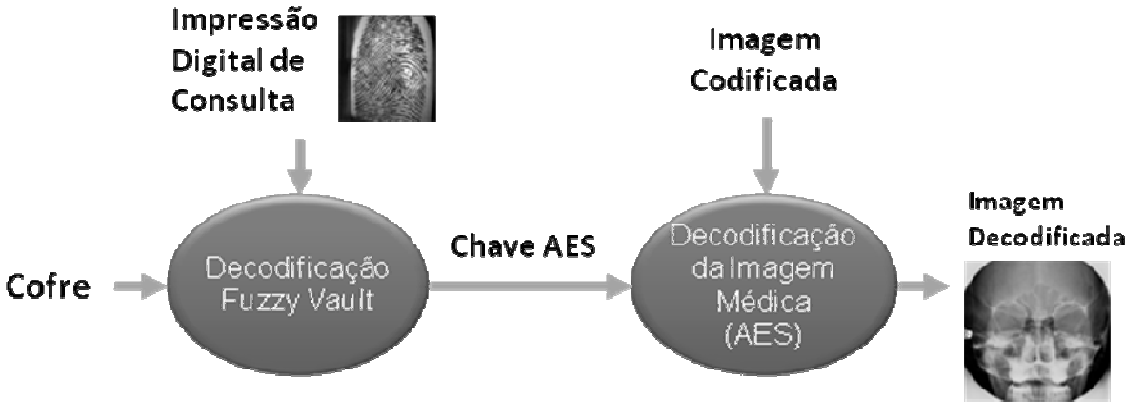

Figura 30 – Algoritmo de desencriptação do cenário de proteção de imagens médicas usando biometria.

4.3. Material

A avaliação utilizou imagens da base de dados FVC 2002 DB1 (MAIO, 2002). Esse banco de dados possui oito imagens de impressões digitais para cada um dos 100 indivíduos, totalizando 800 imagens, a Figura 22 apresenta algumas imagens da base de dados. O banco de dados foi capturado utilizando-se um coletor ótico modelo Biometrika FX2000, sendo que a coleta foi feita em três seções diferentes, com intervalos de duas semanas entre cada seção. Nenhum esforço para controlar a

qualidade das imagens foi realizado. As imagens do banco possuem tamanho original de 560x296 pixels e foram redimensionadas para 256x256 pixels para que as coordenadas da imagem fossem representadas por oito bits cada.

A implementação do *Fuzzy Vault* para impressões digitais utilizou os seguintes parâmetros, o cofre foi encriptado com 200 pontos impostores e a quantidade de pontos genuínos utilizada correspondeu à quantidade de minúcias encontradas na impressão digital modelo. Na desencriptação a quantidade de minúcias utilizadas correspondeu à quantidade de minúcias da impressão digital de consulta. Impressões digitais com menos de 9 minúcias foram descartadas. O valor do coeficiente *t* utilizado no alinhamento foi 12 (coeficiente divido pela distância euclidiana entre a minúcia e o ponto mais próximo no cofre). O limiar para identificação de pontos candidatos estabelecido foi 6 (distância euclidiana entre a minúcia e o ponto mais próximo no cofre).

A aplicação foi desenvolvida na linguagem de programação C++ no ambiente de desenvolvimento *Microsoft Visual Studio .NET 2005*. As rotinas de extração das minúcias das impressões digitais estão presentes na biblioteca *Griaule GRFinger 4.2*. As rotinas que implementam o algoritmo AES e o algoritmo RSA são as rotinas presentes no *Microsoft.NET Framework 2.0*. O desenvolvimento e a avaliação foram realizados utilizando 2 computadores equipados com processador Intel Core 2 Duo T5250, memória RAM de 1GB e disco rígido SATA de 300 GB.

Na etapa de encriptação de imagens médicas foram utilizadas três imagens com tamanho de 5 MB, 27 MB e 108 MB. Essas imagens representam imagens médicas de vários tamanhos. Para encriptação e desencriptação de imagens médicas foram utilizados dois algoritmos: o AES 128 bits e o RSA 512 bits, simétrico e assimétrico respectivamente.

4.4. Método

Foram realizados experimentos para avaliar a viabilidade do *Fuzzy Vault* no cenário de proteção de imagens médicas usando biometria. Os experimentos aferiram o a eficiência (tempo de execução) das etapas de encriptação e desencriptação. O tempo foi aferido nas seguintes etapas, encriptação da imagem médica, encriptação do *Fuzzy Vault*, alinhamento da imagem de consulta, desencriptação do *Fuzzy Vault* e desencriptação da imagem médica. Calculou-se a mediana do tempo das etapas a partir dos tempos aferidos.

Foram realizadas encriptações e desencriptações para os 100 indivíduos da base de dados com as imagens selecionadas aleatoriamente. As desencriptações do Fuzzy Vault que resultaram em erro foram descartas, pois não seria possível desencriptar a imagem médica nessas situações, para cada iteração que resultou em erro, uma nova iteração foi executada para substituí-la.

Analisou-se o tempo de encriptação de desencriptação de uma imagem médica por um criptossistema assimétrico sendo que o criptossistema utilizado foi o RSA. As etapas de encriptação e desencriptação desse algoritmo foram executadas 10 vezes para imagem médica e se calculou o tempo médio de encriptação e desencriptação do algoritmo assimétrico.

Também analisou-se o tempo de encriptação de desencriptação de uma imagem médica pela combinação de criptossistema simétrico e um criptossistema assimétrico, o criptossistema simétrico utilizado foi o AES 128 bits enquanto o criptossistema assimétrico foi o RSA 512. O AES encripta a imagem médica e o RSA encripta a chave AES. As etapas de encriptação e desencriptação desse algoritmo foram executadas 10 vezes para imagem médica e se calculou o tempo médio de encriptação e desencriptação da combinação dos algoritmos.

Após todas as desencriptações as imagens médicas desencriptadas eram comparadas às imagens médicas originais para se verificar se houve alguma alteração nas suas propriedades.

4.5. Resultados

A Tabela 8 apresenta a mediana do tempo encriptação da implementação proposta. Nas três situações a encriptação do *Fuzzy Vault* apresenta valores parecidos. O tempo de encriptação do AES aumenta em função do tamanho da imagem encriptada. O tempo de encriptação da imagem de tamanho menor foi de 1,5 segundos, nessa situação o sistema foi capaz de encriptar 3,26 MB/s. Já o tempo de encriptação da imagem de 108 MB foi de 11,01 segundos, nessa situação o sistema foi capaz de encriptar 9,92 MB/s.

Tabela 8 - Tempo de encriptação utilizando biometria.

		Tamanho da Imagem (MB)		
		5	**27**	**108**
Tempo de Encriptação Mediana (s)	**AES**	0,66	2,56	10,10
	Fuzzy Vault	0,87	0,84	0,78
	Total	1,53	3,40	10,88

A Tabela 9 apresenta a mediana da desencriptação da implementação proposta. A desencriptação proposta apresenta valores parecidos para o alinhamento das imagens e para a desencriptação do *Fuzzy Vault* nas três situações. O tempo da desencriptação AES aumenta em função do tamanho da imagem desencriptada. O tempo de desencriptação da imagem de tamanho menor foi de 4,22 segundos, nessa situação o sistema foi capaz de encriptar 1,18 MB/s. Já o tempo de encriptação da imagem de 108 MB foi de 13,75 segundos, nessa situação o sistema foi capaz de encriptar 7,85 MB/s.

Tabela 9 - Tempo de desencriptação utilizando biometria.

		Tamanho da Imagem (MB)		
		5	**27**	**108**
Tempo de Desencriptação Mediana (s)	**Alinhamento**	3,53	3,54	3,66
	Fuzzy Vault	0,03	0,03	0,03
	AES	0,66	2,57	10,06
	Total	4,22	6,14	13,75

A Tabela 10 apresenta o tempo médio de encriptação e desencriptação das imagens médicas utilizando algoritmo RSA. Tanto na encriptação quanto na desencriptação o tempo aumenta em função do tamanho da imagem encriptada, sendo que o algoritmo RSA é capaz de encriptar 561 KB/s e desencriptar 32 KB/s.

A Tabela 11 apresenta o tempo médio de encriptação e desencriptação das imagens médicas utilizando AES e RSA. Tanto na encriptação quanto na desencriptação o tempo aumenta em função do tamanho da imagem encriptada, sendo que a combinação dos algoritmos é capaz de encriptar entre 6,74 MB/s e 11,14 MB/s e desencriptar 7,18 MB/s e 10,58 MB/s.

Tabela 10 - Tempo de encriptação e desencriptação utilizando o algoritmo RSA.

		Tamanho da Imagem (MB)		
		5	**27**	**108**
Tempo Médio(s)	**Encriptação**	10,78	44,33	169,48
	Desencriptação	228,91	904,36	3.475,84

Tabela 11 - Tempo de encriptação e desencriptação utilizando AES e RSA.

		Tamanho da Imagem (MB)		
		5	**27**	**108**
Tempo Médio (s)	**Encriptação**	0,74	3,54	9,70
	Desencriptação	0,70	3,16	10,20

A Tabela 12 apresenta um comparação entre a implementação proposta neste trabalho, a combinação de AES e RSA, a implementação da literatura e o algoritmo RSA. Enquanto a implementação proposta e a combinação de AES e RSA encriptam e desencriptam imagens na ordem de MB/s a implementação da literatura e o RSA encriptam e desencriptam imagens na ordem de KB/s.

Tabela 12 - Comparação entre a implementação proposta e as técnicas da literatura.

	Sistema Proposto (MB/s)	**AES e RSA (M|B/s)**	**Cao et al. (KB/s)**	**RSA (KB/s)**
Encriptação	3,26 - 9,92	7,84 - 10,83	175	651
Desencriptação	1,18 - 7,85	10,42	46	32

4.6. Discussões e Conclusões

A implementação proposta no cenário de proteção de imagens médicas apresenta uma maneira de se proteger imagens médicas através criptossistemas biométricos. O método pode ser usado para oferecer segurança no armazenamento e comunicação de imagens médicas. Apenas o indivíduo que possui a informação biométrica pode desencriptar a imagem encriptada, isto evita que impostores acessem as chaves criptográficas e desencriptem a imagem. No entanto, caso uma mesma impressão digital seja utilizada duas vezes, pode-se comparar os pontos do cofre e identificar os pontos genuínos. Uma maneira de evitar essa vulnerabilidade é utilizar biometrias canceláveis e para cada imagem utilizar uma senha diferente.

Após analisar os tempos de execução coletados nos experimentos, nota-se que o sistema proposto neste estudo apresenta desempenho de execução melhor do que os sistemas propostos na literatura. O sistema proposto é capaz de encriptar uma imagem médica com tamanho 27 MB em 3,40 segundos e desencriptá-la em 6,14 segundos. Uma imagem de tamanho 108 MB pode ser encriptada em 10,88 segundos e desencriptada em 13,75 segundos. O sistema proposto por Cao et al. (2003) encripta ou desencripta uma imagem médica de 7 MB em 40 segundos e toma 2 a 3 minutos para encriptar ou desencriptar uma imagem de 36 MB em um computador multi-processado Sun Sparc 690MP.

Se comparado a combinação do AES com o RSA, o método proposto apresenta tempos médios de encriptação e desencriptação próximos, mas piores do que os da combinação. Enquanto o método proposto encripta imagens entre 3,26 MB/s e 9,92 MB/s e desencripta imagens entre 1,18 MB/s e 7,85 MB/s a combinação do AES e RSA encripta entre 6,74 MB/s e 11,14 MB/s e desencripta 7,18 MB/s e 10,58 MB/s. Esta diferença se dá no tempo de encriptação e desencriptação do *Fuzzy Vault* e do RSA, pois o AES possui o mesmo desempenho nos dois métodos. Nagar e Chaudhury (2006) propuseram uma técnica que utilizada o *Fuzzy Vault* e o criptossistema RSA para encriptação de documentos. O método proposto no presente estudo apresentou desempenho melhor ao ser comparado com o RSA em um ambiente computacional idêntico. Enquanto o algoritmo RSA é capaz de encriptar 651 KB/s e desencriptar 32 KB/s, o sistema proposto consegue desencriptar entre 3,26 MB/s e 9,92 MB/S e desencriptar entre 1,18 MB/s e 7,85 MB/s dependendo do tamanho da imagem médica. No entanto, como há uma grande variabilidade no tempo de desencriptação do *Fuzzy Vault* para impressões digitais, em algumas situações o desempenho da desencriptação pode variar.

As implementações propostas na literatura (Cao et. al, 2003 e Nagar e Chaudhury, 2006) utilizam algoritmo assimétrico para encriptar imagens médicas, por isso apresentam desempenho pior do que o método proposto (AES e Fuzzy Vault) e da combinação de AES e RSA. A combinação de um algoritmo simétrico com um método de proteção da chave criptográfica apresenta desempenho melhor do que a apenas utilização de um algoritmo assimétrico, pois o algoritmo simétrico encripta a imagem e apenas a chave simétrica é encriptada com um método mais lento como, por exemplo, um algoritmo assimétrico ou um criptossistema biométrico. O algoritmo assimétrico possui a vantagem de ter um tempo de execução mais regular do que o método proposto no presente estudo, já o criptossistema biométrico proposto no presente estudo apresenta a vantagem de não ser necessário manipular chaves criptográficas.

REFERÊNCIAS BIBLIOGRÁFICAS

BIOMETRICS - Wikipedia, the free encyclopedia, **Biometrics - Wikipedia, the free encyclopedia**, disponível em http://en.wikipedia.org/wiki/Biometrics, último acesso em 13 de março de 2008.

BOVELANDER, E.; RENESSE, R. L. Smartcards and biometrics: an overview. **Computer Fraud & Security Bulletin**, London, v. 1995, n. 12, p 8-12, 1995.

BUCHMANN, J. **Introdução à Criptografia**. São Paulo: Berkeley, 2002.

CAO, F.; HUANG, H. K.; ZHOU, X. Q. Medical image security in a HIPAA mandated PACS environment, **Computerized Medical Imaging and Graphics**, Elmsford, v. 27, n. 2, p. 185-196, 2003.

CLANCY, T. C.; KIYAVASH, N.; LIN, D. J. Secure smartcard-based fingerprint authentication. **Proceedings of the 2003 ACM SIGMM Workshop on Biometrics Methods and Applications**, New York, p. 45-52, 2003.

DASS, S. C.; JAIN, A. K. Fingerprint classification using orientation field flow curves. **In Proc. Indian Conference on Computer Vision, Graphics and Image Processing**, India, p. 650-655, 2004.

DAUGHMAN, J. G. High confidence visual recognition of persons by a test of statistical independence. **IEEE Trans. Pattern Anal. Machine Intelligence**, [S.I.], v. 15, n. 11, p 1148–1161, 1993.

DAVIDA, G. I. et al. On the relation of error correction and cryptography to an off-ine biometric based identification scheme. **Proceedings of WCC99, Workshop on Coding and Cryptography**, France, p. 129-138, 1999.

DAVIDA, G. I.; FRANKEL, Y.; MATT, B. J. On enabling secure applications through off-line biometric identification. **IEEE Symposium on Privacy and Security**, Oakland, p. 3-6, 1998.

ELLISON, C. et al. Protecting secret keys with personal entropy. **Future Generation Computer Systems**, Amsterdam, v. 16, n. 4, p. 311-318, 2000.

FREIRE-SANTOS, M.; FIERREZ-AGUILAR, J.; ORTEGA-GARCIA, J. Cryptographic key generation using handwritten signature. **Proceedings of SPIE, the International Society for Optical Engineering**, Bellingham, p. 255-231, 2006.

HUANG, H. K. **Picture archiving and communication systems:** principles and applications**,** New York: Wiley, 1999.

JAIN, A. K.; ROSS, A.; PRABHAKAR, S. An introduction to biometric recognition. **IEEE Transactions on Circuits and Systems for Video Technology**, [S.I.] , v. 14, n. 1, pp. 4-20, 2004.

JAIN, A. K.; ROSS, A.; PRABHAKAR, S. Fingerprint matching using minutiae and texture features, **Proc. Int. Conference on Image Processing (ICIP)**, Thessaloniki, Greece, v. 3, p. 282-285, 2001.

JAIN, A. K.; HONG, L.; PANKANTI, S.; BOLLE, R. An identity authentication system using fingerprints. **Proceedings of the IEEE**, [S.I.], v. 85, n. 9, p. 1365-1388, 1997.

JUELS, A.; SUDAN, M. A fuzzy vault scheme. **IEEE International Symposium on Information Theory**, [S.I.], p. 408, 2002.

JUELS, A.; SUDAN, M. A fuzzy vault scheme. **Designs, Codes and Cryptography**, Norwell, v. 38, n. 2, p. 237-257, 2006.

JUELS, A.; WATTENBERG, M. A fuzzy commitment scheme. **Proceedings of the 6th ACM conference on Computer and communications security**, New York, p. 28-36, 1999.

KLEIN, D. V. Foiling the cracker: a survey of, and improvements to, password security. **Proceedings of the 2nd USENIX Unix Security Workshop**, London, p. 5-14, 1990.

MALICKAS, A.; VITKUS, R. Fingerprint registration using composite features consensus. **Informatica, Institute of Mahematics and Informatics (Vilnius)**, [S.I.] v. 10, n. 4, p. 389-402, 1999.

MAIO, D. et al. FVC2002: second fingerprint verification competition. **Proceedings of the 16th International Conference on Pattern Recognition**, [S.I.], v. 3, pp. 811-814, 2002.

MALTONI, D. et al. **Handbook of Fingerprint Recognition**. New York: Springer, 2003.

MARANA, A. N.; JAIN, A. K. Ridge-based fingerprint matching using hough transform. **Proceedings of the XVIII Brazilian Symposium on Computer Graphics and Image Processing (SBIGRAPI)**, Washington, p. 112, 2005;

MONROSE, F.; REITER, M. K.; WETZEL, S. Password hardening based on keystroke dynamics. **Proceedings of the 6th ACM conference on Computer and communications security**, New York, p. 73-82, 1999.

MONROSE, F.; REITER, M. K.; LI, Q.; WETZEL, S. Cryptographic key generation from voice. **Proc. 2001 IEEE Symp. Security and Privacy**, Oakland, p. 202-213, 2001.

MONROSE, F.; REITER, M. K.; LI, Q.; WETZEL, S. Using voice to generate cryptographic keys. **Proc. 2001: A Speaker Odyssey, Speaker Recognition Workshop**, Crete, Greece, p. 237-242, 2001.

NAGAR, A. e CHAUDHURY, S. Biometrics based Asymmetric Cryptosystem Design Using Modified Fuzzy Vault Scheme, **Proceedings of the 18th International Conference on Pattern Recognition,** Washington, v. 4, p. 537-540, 2006.

NANDAKUMAR, K.; NAGAR, A. e JAIN, A. K. Hardening Fingerprint Fuzzy Vault Using Password. **Advances in Biometrics**. Berlin: Springer, 2007. p. 927-937.

NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION. **Digital Imaging and Communications in Medicine (DICOM), Part 15:** Security and System Management Profiles**,** PS 3.15-2008. Rosslyn, VA, 2007.

NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION. **Digital Imaging and Communications in Medicine (DICOM), Part 1:** Introduction and Overview**, PS 3.1-2008**, Rosslyn, VA, 2008.

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. **Advanced Encryption Standard (AES)**, 2001, Disponível em: <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>. Acesso em 19 de maio de 2007.

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. An Introduction to Computer Security: The NIST Handbook. **NIST Computer Security Special Publications**, 1995. Disponível em: <http://csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf>. Acesso em 19 de maio de 2007.

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. **Data Encryption Standard (DES)**, 1999, Disponível em: < http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf>. Acesso em 05 de maio de 2007.

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. **SECURE HASH STANDARD (SHA)**, 1995, Disponível em: <http://www.itl.nist.gov/fipspubs/fip180-1.htm>. Acesso em 05 de maio de 2007.

NORCEN R. et al. Confidential storage and transmission of medical image data. **Computers in Biology and Medicine**, [S.I.], v. 33, n. 3, p. 277-292, 2003.

REDDY, E. S.; BABU I. R. Performance of iris based hard fuzzy vault. **International Journal of Computer Science and Network Security**, [S.I.], v. 8, n. 1, p. 297-304, 2008.

RIVEST, R.; SHAMIR, A.; ADLEMAN, L. A method for obtaining digital signatures and public-key cryptosystems. **Communications of the ACM**. New York, v. 21, n. 2, p. 120-126, 1978.

SMITH, J.P. Authentication of digital medical images with digital signature technology. **Radiology**, [S.I.], v. 194, p. 771-774, 1995.

SOUTAR, C., et al. Biometric encryption, **ICSA Guide to Cryptography**, New York: McGraw-Hill, 1999.

SOUTAR, C., et al. Biometric encryption using image processing., **Proc. Spie, Optical Security and Counterfeit Deterrence Techniques II**, [S.I.], v. 3314, p. 178-188, 1999.

STINSON, D. R. **Cryptography:** theory and pratice. 2. ed. Ontario, Canada : Chapman & Hall/CRC, 2002. 339 p. 1-58488-206-9.

ULUDAG, U.; PANKANTTI, S.; JAIN, A. K. Fuzzy vault for fingerprints. **AVBPA, Lecture Notes in Computer Science**. [S.I.] v. 3546, p. 310-319, 2005.

ULUDAG, U. et al. Biometric cryptosystems: issues and challenges. **Proceedings of the IEEE, Special Issue on Enabling Security Technologies for Digital Rights Management**, [S.I.], v. 92, n. 6, p. 948- 960, 2004.

ULUDAG, U; JAIN, A. K. Fuzzy fingerprint vault. **Proc. Workshop: Biometrics: Challenges Arising from Theory to Practice**, [S.I.], p. 13-16, 2004.

ULUDAG, U.; JAIN, A. K; Securing fingerprint template: fuzzy vault with helper data. **Computer Vision and Pattern Recognition Workshop**. Washington, p. 163, 2006.

US DEPARTMENT OF HEALTH AND HUMAN SERVICES. **HIPAA**, 2003. Disponível em: <http://aspe.os.dhhs.gov/admnsimp>. Acesso de 17 de outubro de 2007.

WANG, Y.; PLATANIOTIS, K.N. Fuzzy vault for face based cryptographic key generation. **Biometrics Symposium**, [S.I.], pp. 1-6, 2007.

WOODWARD, J. D.; ORLANS, N. M.; HIGGINS P. T. **Biometrics**. Berkeley: McGraw-Hill/Osborne, 2003.

YANG, S.; VERBAUWHEDE, I. Automatic secure fingerprint verification system based on fuzzy vault scheme. **IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2005)**. [S.I.], v.5 , p. 609-612, 2005.

YANG, S.; VERBAUWHEDE, I. Secure fuzzy vault based fingerprint verification system. **Proc. 38th Asilomar Conference on Signals, Systems, and Computers**, Piscataway, p. 577-581, 2004.

Autorizo a reprodução xerográfica para fins de pesquisa.

São José do Rio Preto, 31/07/2008

_____

Assinatura