一种改进的指纹 Fuzzy Vault 加密方案*

谭台哲，章红燕
（广东工业大学 计算机学院，广州 510006）

摘　要：为了克服传统指纹 Fuzzy Vault 算法在应用上的不足，提出了一种改进的 Fuzzy Vault 算法。主要采用一种新的多项式构造函数的方法来实现待保护密钥和指纹细节特征点的绑定，进而解决了传统 Fuzzy Vault 算法中密钥长度决定解码时需要匹配细节点个数的问题。实验结果表明，此方法进一步增强了算法的安全性和灵活性。

关键词：模糊密钥绑定；生物特征加密；密钥绑定；生物特征；指纹细节点
中图分类号：TP391　　**文献标志码**：A　　**文章编号**：1001-3695(2012)06-2208-03
doi:10.3969/j.issn.1001-3695.2012.06.054

Improved Fuzzy Vault fingerprint encryption scheme

TAN Tai-zhe, ZHANG Hong-yan
(*Faculty of Computer, Guangdong University of Technology, Guangzhou* 510006, *China*)

Abstract: In order to overcome the deficiency of the traditional Fuzzy Vault algorithm in the application, this paper proposed an improved Fuzzy Vault algorithm. It mainly adopted a novel polynomial constructor method to realize the cryptographic key and fingerprint minutiae data binding together, and then solved the problem that the matching minutiae number was determined by the length of secret in the traditional Fuzzy Vault algorithm in key decoding phase. The experiment results show that this method further enhances the security and flexibility of the algorithm.

Key words: Fuzzy Vault; biometric encryption; key binding; biometric; fingerprint minutiae

0　引言

随着生物特征识别技术应用的愈加深入，其本身固有的一些缺陷和不足也逐渐暴露出来，最主要的就是生物特征所涉及到的人的隐私以及由此而带来的安全性问题。因此，一种生物特征识别技术与密码技术相结合的生物特征加密技术应运而生，成为未来信息安全领域一个重要发展趋势。生物特征加密技术[1]，简单来讲，就是将生物特征与密码技术相结合，产生可撤销且不可逆的生物特征模板，以此来达到保护人们的生物特征安全和隐私的目的，是一种通过使用生物特征来保护密码的密钥系统。根据密钥的使用方式或者产生方式的不同，又可以分为密钥释放(key release)、密钥绑定(key binding)和密钥生成(key generation)三种具体方法[2]。在密钥绑定[3]方法中，用户将待保护的密钥与自己生物特征绑定在一起，只有当生物特征匹配成功时密钥才能被相应算法提取出来，密钥得以释放；否则为非法用户，系统输出一个拒绝信号。

模糊密钥绑定(Fuzzy Vault)算法是由 Juels 等人[4]于2002年提出，其最大的特点就是它的模糊性。很多研究者的工作都是基于这个算法的。之后，Clancy 等人[5]在 Juels 等人工作的基础上提出了 Fingerprint Vault 的概念，并给出了指纹域多项式次数的具体描述。与此同时，Uludag 等人[6]提出了更为实用的 Fuzzy Vault for Fingerprint，并首次提出使用辅助信息(helper data)在加密域内进行指纹配准，进一步增强算法安全性。在 Fuzzy Vault 算法方案中，其主要思想是将待保护的密钥隐藏于多项式系数中，而后将生物特征数据作为此多项式的变量得到相应的多项式值，同时生成大量不在此多项式上的杂凑点，形成了 Vault。解码时通过从 Vault 中分离出真实特征点，然后再通过多项式重构出加密密钥。

在现有的指纹 Fuzzy Vault 方案中，密钥的长度决定了构造多项式函数的幂次，从而最终决定解码时查询指纹与注册模板指纹中匹配的细节点数目。从理论上来说，密钥长度与匹配的细节点数目两者之间应该没有必要的联系[7]。这种关系限制了 Fuzzy Vault 算法的灵活性。在文献[8]中也指出，验证一个人的身份至少需要 12 个细节点。因此，当密钥长度过短时，需要验证的指纹细节点个数较少，这样加密系统就会存在安全隐患。针对上面提出的 Fuzzy Vault 算法的不足，本文提出了一种“多减少补”的原则来构造多项式函数，从而解除了密钥长度决定解码时匹配的细节点数目的这种联系，避免系统产生安全隐患问题，使得 Fuzzy Vault 算法使用起来更为灵活。

1　Fuzzy Vault

该算法包括两个步骤：a)用户 Alice 将密钥 S 放置到保险箱(Vault)中，并用无序集 A 加以锁定；b)用户 Bob 使用无序集 B 尝试打开保险箱 Vault 并访问 S。只有当无序集 B 中的元素与无序集 A 中的元素大多数重合时，用户 Bob 才能打开保险箱 Vault 并访问 S。

1)密钥绑定(图1)阶段　首先使用一种纠错码技术对密钥 S 进行处理，在密钥 S 尾部加上以上纠错码技术产生的检验

收稿日期：2011-11-11；**修回日期**：2011-12-14　　**基金项目**：国家自然科学基金资助项目(60974019)；广东省自然科学基金资助项目(9451009001002686)

作者简介：谭台哲(1970-)，男，山东莱阳人，副教授，硕导，主要研究方向为生物特征识别、图像处理等；章红燕(1986-)，女，硕士研究生，主要研究方向为生物特征加密(xiaozhang8712@126.com).

码,形成 sc,然后使用 sc 按照一定规则构造多项式函数 P。另一方面,提取用于加密指纹模板图像的细节点特征(x,y,θ);再级联横坐标和纵坐标,即 $x|y$,找到 $x|y$ 在 P 上的投影点$(x|y,P(x|y))$;然后随机添加一组不在 P 上且距离真实细节点一定距离的杂凑点到保险箱 Vault 中,将真实细节点与杂凑点进行置乱,这样就构成了最终的保险箱 Vault。

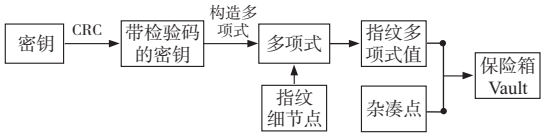

图1　密钥绑定

2)密钥解码(图 2)阶段　首先提取用于比对的指纹图像查询细节点位置(x,y,θ),然后寻找保险箱 V 中与它们匹配的点,找到若干候选点,如果是合法用户,这一步将会过滤掉大量的杂凑点。之后,在找到的若干候选点中使用 Lagrange 插值方法重构相应的多项式,用加密阶段使用的检验技术来检验纠错码,确定哪些为真实细节点,最终重新获取密钥 S。下面是这个算法的一个简单例子。

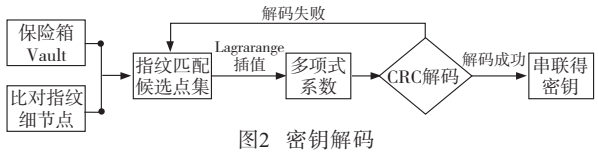

图2　密钥解码

假设用户 A 根据密钥选择的多项式函数 P 是 $2X^2-X+1$。如果无序集合 A 是$\{-1,-2,2,4\}$,获得的多项式投影是$\{(A,P(A))\}=\{(-1,4),(-2,11),(2,7),(4,29)\}$。再加上两个干扰点 $C=\{(0,2),(2,0)\}$,这两个点不在多项式上。因此最终的点集 V 为$\{(-1,5),(-2,11),(3,1),(2,-1),(0,2),(1,0)\}$。现在如果 Bob 能够从 V 中分离出基于 P 的至少三个点对,则能重构得到多项式函数 P,也就是能够得到代表密钥信息的多项式系数$(2,-1,1)$;否则,将得不到正确的多项式函数 P。

2　改进的指纹 Fuzzy Vault 加密方案

在指纹 Fuzzy Vault 算法中,密钥长度的设定是一个关键问题,直接影响到后面所构造的多项式函数。由于加密密钥构成了多项式函数的系数,即密钥的长度决定了构造多项式函数的幂次,从而最终决定解码时查询指纹与注册模板指纹中匹配的细节点数目。对于长度为 L 的加密密钥,在运算域 GF(2^{16})上可以确定解码需要匹配的细节点数目为 $L/16+1$;而在文献[8]中指出,验证一个人的身份至少需要 12 个细节点,当 $L/16+1$ 不大于 12 时,指纹 Fuzzy Vault 算法将存在安全隐患。从理论上来说,密钥长度与匹配的细节点数目两者之间应该没有必要的联系。这种关系限制了 Fuzzy Vault 算法的灵活性。为此,本文提出了一种新的构造多项式函数的方法,避免了这种联系的存在给系统带来的安全隐患。

令解码阶段要匹配的细节点数目记为 N,加密密钥 S 的长度记为 L,整个加/解密过程在运算域 GF(2^{16})上进行。改进的 Fuzzy Vault 算法的主要思想是,按"多减少补"的原则构造多项式函数。由于验证一个人的身份至少需要 12 个细节点,即假设匹配的细节点数目 $N=12$,下面根据 N 与密钥长度 L 的比较来实现多项式函数的构造。

2.1　"补"原则构造多项式函数的指纹细节点加密

如果 $L/16+1<12$,则采用"补"的方法构造多项式函数,即补 p 位的伪随机数,使得 $L/16+p/16+1=N$,使用对称加密算法 AES 生成密钥。假设生成的密钥 L 长度为128 bit的串,匹配的细节点数目 $N=12$,算法的加/解密过程如下:

a)密钥绑定过程。其中关键一步是构造多项式函数 $p(x)$,整个加/解密过程是在运算域 GF(2^{16})上进行。密钥 L 长度为128 bit,由于 $L/16+1<12$,因此采用"补"的方法构造多项式函数。在构造多项式的过程中,先将密钥 L 顺序分割成 8 个长度均为16 bit的串,作为多项式系数一部分,分别记为 r_8、r_7、…、r_1;然后用 16bit 的伪随机数发生器生成$P=48$ bit的随机数,分割成 3 个长度均为 16 bit 的串,作为多项式系数的另一部分,记为 r_{11}、r_{10}、r_9,使得 $L/16+P/16+1=N$;最后使用循环冗余校验(CRC)对整个多项式的系数 R 进行处理,即计算系数 R 的 16 bit 的 CRC-16 纠错码,记为 r_0,则构造的多项式函数 $P(x)$ 为

$$P(x)=r_{11}x^{11}+r_{10}x^{10}+\cdots+r_1x^1+r_0 \tag{1}$$

然后提取指纹细节点。对于指纹细节点特征一般需要记录其如下属性:特征点坐标值(x,y);特征点所在纹线方向 θ;种类 T(是端点还是分叉点)。设指纹模板细节点特征记为 m_i,其细节点特征表示为(x,y,θ)。由于指纹特征点所在纹线方向与细节点横纵坐标有很强的相关性,因此本文仅使用指纹细节点的横纵坐标来进行加密。提取出指纹细节点特征记为 $m_i=[(x_i,y_i)|i=1,2,\cdots,N]$,其中 N 为模板细节点个数。串联横纵坐标 x_i 与 y_i,x_i 与 y_i 分别用8 bit的串表示,串联后为 16 bit,记做 M_i。将串联后的细节点特征值投影到多项式函数 $P(x)$ 上,可以得到集合 $A=\{M_i,p(M_i)|i=1,2,\cdots,N\}$。接着随机生成大量杂凑点集合 $C=(u_j,v_j|j=1,2,\cdots,J,J>>N,v_j\neq p(u_j))$,从杂凑点集中可以看出,杂凑点的数量远大于细节点数量,生成的杂凑点不在多项式函数 $P(x)$ 上,且杂凑点距离真实细节点有一定的距离。将集合中的点进行置乱,生成最终的保险箱 $V=\{A\cup C\}$。

b)密钥解码过程。首先将要进行比对的指纹进行处理,提取出相应的细节点特征值,组成特征点集合 $m'_k=[(x''_i,y''_i)|k=1,2,\cdots,K]$,其中 K 为采样指纹的特征点数目。对于保险箱 Vault 中的任意元素(a,b),把 a 拆分为两个8 bit的字符串,分别记为 x'_i 与 y'_i,这样就构成另一集合 $C'=(x'_i,y'_i)$,该集合由 N 个真实点和 J 个杂凑点信息共同组成。再逐个比对采样指纹特征点集合 m'_k 与 C' 集合中的元素,如果采样指纹来自真实用户,上一步通过比对可以滤除大量的杂凑点,得到一组候选点集合 $V'=\{(x_i,f_i)|i=1,2,\cdots,W\}$。为了重构出多项式 $P(x)$,候选点集合的数目 W 应该满足 $12\leqslant W\leqslant N+L$。再从候选点集合中任选出 12 个候选点作为一组进行 Lagrange 插值,用 Lagrange 插值法重构出多项式函数 $P(x)$,直到找到其系数 r'_0 满足 CRC-16 校验码的那组候选点组合为止。最后串联各系数得到密钥 S。其中 Lagrange 插值法公式为

$$\begin{cases}P'(x_1)=r'_{11}x^{11}+r'_{10}x_1^{10}+\cdots+r'_1x_1+r'_0=f_1\\P'(x_2)=r'_{11}x_2^{11}+r'_{10}x_2^{10}+\cdots+r'_1x_2+r'_0=f_2\\\cdots\\P'(x_{11})=r'_{11}x_{11}^{11}+r'_{10}x_{11}^{10}+\cdots+r'_1x_{11}+r'_0=f_{11}\end{cases} \tag{2}$$

化简式(2),即可求出 $P'(x)=r'_{11}x^{11}+r'_{10}x_1^{10}+\cdots+r'_1x_1+r'_0$,将 r'_{11}、r'_{10}、…、r'_1、r'_0 分别表示为16 bit的形式。并计算 r'_{11}、r'_{10}、…、r'_1的 CRC-16 校验码,如果得出的结果为 r'_0,则表示解码成功,比对指纹为合法用户,串联 r'_8、r'_7、…、r'_1,得出最终密钥结果;如果校验结果不为 r'_0,则表示恢复密钥失败,为非法用户的访问。

2.2 “减”原则构造多项式函数的指纹细节点加密

如果 $L/16+1 \geqslant N$,则采用“减”的方法来构造多项式函数。使用对称加密算法 AES 生成密钥,假设生成的密钥 L 长度为 256 bit的串,匹配的细节点数目 $N=12$,以下是算法构造多项式函数过程描述:

a)密钥绑定阶段。密钥长度 L 为256 bit。在构造多项式的过程中,先将密钥 L 顺序分割成 16 个长度均为16 bit的串,分别记为 r_{16}、r_{15}、…、r_{11}、r_{10}、…、r_1。由于要匹配的细节点数目 $N=12$,即构成的多项式函数为 $P(x)=r_{11}x^{11}+r_{10}x^{10}+\cdots+r_1x^1+r_0$,其中 r_0 为多项式系数 $R=\{r_{11},r_{10},\cdots,r_1\}$ 的 CRC-16 纠错码,将提取的指纹细节点特征值投影到此多项式函数 $P(x)$ 上,得到集合 $A=\{M_i,p(M_i)\mid i=1,2,\cdots,N\}$,将余下的系数 r_{12}、r_{12}、r_{14}、r_{15}、r_{16}与六个字节全为 0 的数顺序连接在一起,组成分组为128 bit的 R',使用对称加密算法 AES 进行加密。将 R'作为明文,得到相应密文 $B=E(R',k)$,其中 k 为对称加密算法的密钥。接着随机生成大量不在此多项式函数上且距离真实细节点一定距离的杂凑点,得到集合 $C=(u_j,v_j\mid j=1,2,\cdots,J,J>>N,v_j\neq p(u_j))$,因此生成最终的保险箱 $V=\{A\cup B\cup C\}$。

b)密钥解码阶段。首先提取用于比对的指纹图像中的细节点位置(x,y);然后寻找保险箱 V 中与它们对应的点。如果采样指纹来自真实用户,上一步通过比对可以滤除大量的杂凑点,得到若干组候选点集合。从候选点集合中任选出 12 个候选点作为一组进行 Lagrange 插值,用 Lagrange 插值法重构出多项式函数 $P(x)$,直到找到其系数 r'_0 满足 CRC-16 校验码的那组候选点组合为止。同时使用 AES 算法,应用密钥绑定阶段中使用的密钥 k 对密文 B 进行解密,即 $R''=D(B,k)$,得到另一部分密钥算 r'_{12}、r'_{13}、r'_{14}、r'_{15}、r'_{16},与上一步重构出的多项式系数一起,重新串联组合成密钥,得到最终密钥结果。

3 实验结果与分析

本文实验是在 FVC2004 的 DB1 指纹库上应用 MATLAB 工具进行仿真编程测试。在 DB1 指纹库中,有 100 个人的指纹,每人 8 幅图像,每幅图像的大小是 640×480 像素。首先对 FVC2004 的 DB1 数据库中的所有指纹图像进行预处理,然后提取出指纹细节特征点,将提取得到的细节特征点再进行去伪处理,使得到的细节特征点更为真实。对于每个指纹的 8 幅图像,选择细节点最多的那幅图像作为模板,利用它来生成 Vault,其他图像作为样本。考虑到密钥越长,运行速度就越慢,实验分别基于 64、96以及 128 bit 的密钥,其中需要匹配的细节点数目分别设为 8 和 13,应用本文方法和文献[6]中提出的方法分别进行测试,并将测试结果进行对比。对比结果如表 1、2 所示。

表 1 匹配的细节点数目 $N=13$

密钥长度/bit	本文方法($N=13$)		文献[6]方法	
	拒真率(FRR)/%	认假率(FAR)/%	拒真率(FRR)/%	认假率(FAR)/%
64	15.82	1.25	3.53	14.12
96	16.56	0.68	11.35	6.35
128	18.13	0	21.92	0

表 2 匹配的细节点数目 N 不同

密钥长度/bit	本文方法($N=8$)		文献[6]方法($N=13$)	
	拒真率(FRR)/%	认假率(FAR)/%	拒真率(FRR)/%	认假率(FAR)/%
64	12.64	1.73	15.82	1.25
96	15.18	0.92	16.56	0.68
128	17.61	0.11	18.13	0

从表 1 可以看出,相对文献[6]中介绍的方法,本文方法拒真率相对稳定,变化幅度不大,同时,认假率不高。这是因为在文献[6]中,由于密钥长度决定解码时需要匹配细节点数目,对于较短长度的密钥,解码需要匹配的细节点数目较少,所以认假率较高。而本文方法中事先给出了需要匹配的细节点数目,当密钥长度较短时,可以采用“补”原则构造多项式函数进行细节点加密,所以认假率相对较低,因此进一步增强了算法的安全性。从表 2 可以看出,在本文方法中,在密钥长度不变的情况下,需要匹配细节数目越多,其拒真率就越大,即随着解码时所需匹配的细节点数目的增大,其拒真率增大。同时在匹配的细节点数目确定的情况下,其拒真率随密钥长度的增大而增大,但增大幅度相对稳定。这是因为预先给出了密钥恢复时所需匹配的细节点数目,也就确定了编码多项式的幂次,使密钥长度与编码多项式的幂次没有必然的联系,所以对于不同长度的密钥均具有相对稳定的性能及更好的灵活性。

对上述结果分析可知,本文方法拒真率相对稳定,且认假率为1/1 000量级,在可接受范围内。相对于文献[6],本文方法解决了传统 Fuzzy Vault 算法中加密密钥长度较短时安全性不足的问题,同时也解除了密钥长度决定解码时匹配的细节点数目的这种联系,因此本文方法具有相对较高的安全性及更好的灵活性。

4 结束语

本文分析了现有指纹 Fuzzy Vault 方法的不足,针对 Fuzzy Vault 算法中密钥长度决定 Fuzzy Vault 编码多项式幂次的问题,提出一种“多减少补”的构造多项式函数的方法,避免了密钥长度决定解码阶段匹配细节特征点数目而存在的安全隐患,使得改进的指纹 Fuzzy Vault 算法的使用更为安全和灵活。

参考文献:

[1] 田捷,杨鑫. 生物特征识别理论与应用[M]. 北京:清华大学出版社,2009:358-360.

[2] 姚琳,范庆娜,孔祥维. 基于生物加密的认证机制[J]. 计算机应用研究,2010,27(1): 268-270.

[3] LIU Hai-lun, SUN Dong-mei, XIONG Ke, *et al*. Is fuzzy vault scheme very effective for key binding in biometric cryptosystems[C]//Proc of International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery. 2011:279-284.

[4] JUELS A, SUDAN M. A fuzzy vault scheme[C]//Proc of International Symposium on Information Theory. [S. l.]:IEEE Press, 2002:408-426.

[5] CLANCY T C, KIYAVASH N, LIN D J. Secure smartcard-based fingerprint authentication[C]// Proc of ACM SIGMM Multimedia, Biometrics Methods and Applications Workshop. New York:ACM Press, 2003:45-52.

[6] ULUDAG U, JAIN A. Securing fingerprint template: fuzzy vault with helper data[C]// Proc of CVPR Workshop on Privacy Research in Vision. Washington DC:IEEE Computer Society, 2006:163-169.

[7] BRINDHA V E. Biometric template security using fuzzy vault[C]// Proc of the 15th International Symposium on Consumer Electronics. 2011:384-387.

[8] MERKLE J, NIESING M, SCHWAIGER M, *et al*. Provable security for the fuzzy[C]//Proc of the 5th International Conference on Internet Monitoring and Protection. Washington DC:IEEE Computer Society, 2010:65-73.