基于 H.264 的视频存储加密方案①

Video Store Encrpytion Scheme Based on H.264

闫　磊　蔡　勉　(北京工业大学 电子信息与控制工程学院 北京 100022)

贾　佳　(华北计算技术研究所 北京 100083)

摘　要： 随着视频和互联网技术的发展，视频信息的存储安全问题越来越紧迫。本文针对新一代国际视频压缩编码标准 H.264 的编码特点，提出一种将流密码与 H.264 熵编码相结合的视频加密算法，通过加密码字索引达到保密目的。实验证明，此算法在安全性、计算复杂度、压缩比以及率失真等方面性能良好，适用于政治、军事、商务机要视频信息以及 PC 机、移动存储设备中的私密视频信息的安全存储管理。

关键词： 视频安全存储 H.264 视频加密 CAVLC 流密码

1　引言

随着视频技术和网络通信技术的发展，视频信息安全已成为日益迫切的问题。视频信息安全大体可分为视频存储安全和视频传输安全[1]。视频信息的存储安全包括视频的存储环境安全、存储介质安全、存储管理安全等方面；视频信息的传输安全针对视频传输过程，需要应对传输中的信息非法拦截、信息泄露、信息篡改等入侵手段。视频传输过程的安全研究固然重要，但安全存储这一环节也不能忽视。现代社会中，人们越来越多地将重要、私密的视频文件存储在个人电脑、手机、移动存储设备中，一旦这些设备被他人获得，如果没有足够的文件保密措施，个人隐私将得不到保障。此外，在政治、军事、商务等机要视频文件的管理和存档过程中，如在视频存储上不加以安全手段，也会产生重大隐患。因此，如何设计一种行之有效的视频文件保密方案，具有非常重要的研究意义和实际用途。

目前最新视频压缩编码国际标准是 H.264，与以往压缩标准相比，它具有更高效的压缩性能和更好的网络亲和力。随着 H.264 标准的推出和发展，基于 H.264 标准的产品纷纷面市，其应用范围涵盖了视频存储播放、实时视频通信、广播电视、网络视频点播(VOD)等多个领域，具有广阔的应用前景。而其安全问题也必将成为视频安全算法研究的热点。

目前市面上的视频保护技术多采用用户权限机制，视频内容本身未被加密，这为盗取者提供了在视频存储过程中非法拷贝、盗印、伪造、篡改的机会。而当前大多数视频加密技术也主要局限于最常用的 MPEG 视频，因此研究适合于 H.264 的视频内容加密算法是必要的。本文通过对 H.264 熵编码特点进行研究，提出一种适用于视频安全存储的加密方案。并通过其安全性、计算复杂度、压缩比保持性等性能的分析，以及与其它视频加密算法性能的对比，证明此方案的可行性。

2　基于熵编码的视频加密方案

根据加密过程与编码过程结合方式的不同，现有的视频加密算法主要可分为三种[2]：完全加密算法、选择性加密算法、基于熵编码的加密算法。目前视频安全的研究主要集中在基于 DCT 的选择性加密算法上，此算法实时性较好，可操作性较强，但加密 DC 系数会改变残差数据的统计特性，对 DCT 变换之后的熵编码压缩性能有一定影响。然而视频数据的安全存储对于安全性和压缩编码性能等方面具有很高的要

① 基金项目:北京市自然科学基金项目(4062007);北京工业大学研究生科技基金科研项目(ykj-2006-154)
收稿时间:2008-09-22

求，这就需要加密算法能更好地与视频压缩编码过程相结合，而基于熵编码的加密算法由于其高效、安全和较好的压缩性能，必将成为视频安全未来研究的热点和重点。

基于熵编码的加密算法，前人已有一定研究。Jingtao Wen 和 Mike Severa[3]提出了一种格式兼容性加密框架，将加密区域分为定长编码和可变长编码，给码表的每一个码字分配了一个索引序号，对串联索引序号进行加密，而不是对码字本身内容进行加密，然后将加密后的索引序号映射回原来的码表中，由此完成码流的加密，同时又保证了格式的相容性。

Chunping Wu 和 C.-C. J Kuo 提出一种 MHT 算法[4]，通过变异树训练 2K 个不同的 huffman 编码表，产生随机变量作为密钥选择不同的码表从而完成对原始数据流的加密。Dahua Xie[6]在 MHT 的基础上提出一种加强型 MHT 方案，采用 hush 函数仿制跳跃密钥来对抗选择性明文攻击，提高了算法的安全性。

在熵编码的过程中融入加密过程的构想虽已付诸研究，但是正如前面所介绍的，已有的基于熵编码的视频加密算法大多是针对 MPEG 视频的，而 H.264 的熵编码与 MPEG 熵编码原理不同，因此如何设计一种适合于 H.264 熵编码特点的加密算法是我们要解决的关键问题。

2.1 H.264 熵编码特点

H.264 提供了两种熵编码方法：基于上下文的自适应变长编码(CAVLC)和基于上下文的自适应二进制算术编码(CABAC)[6]。其中 CAVLC 容易实现，计算简单，可通过查询变长码表实现，对于比特误码具有很好的鲁棒性。CABAC 可以自适应于视频内容的变化，编码效率很高，但运算复杂度较高。由于 H.264 的基本档次支持可视电话、会议电视、无线通信等多种视频通信场合，涵盖范围比较广泛，而主档次所支持的数字广播电视并非现代视频应用的主流，在未来一段时间还有待开发，因此本论文仅针对基本档次所支持的 CAVLC 进行研究，而对主档次所支持的 CABAC，由于其编码复杂性暂且不作考虑。

H.264 的 CAVLC 可分为以下五个步骤：

(1)编码非零系数总数 TotalCoeffs 和拖尾系数(TrailingOnes)；

(2)编码所有 TrailingOnes 的符号；

(3)编码除了拖尾系数以外非零系数幅值(Levels)；

(4)编码最后一个非零系数前的零总数(TotalZeros)；

(5)编码每个非零系数前的零数目(Runbefore)。

这五个过程中都有相应的码表，我们的加密将利用到 CAVLC 编码码表。下面将介绍本文所给出的算法。

2.2 本文建议的视频加密方案

本文在借鉴已有熵编码加密算法的基础上，给出一种适合于 H.264 熵编码特点的加密方案。如图 1 所示，在 CAVLC 过程中，利用流密码算法加密码字索引，得到一个新的索引号，然后根据这个新的索引号在原码表中找到其对应的新的码字作为码流输出。通过这种加密方式打乱码字在表中的位置，得到与原码字不同的新码字。

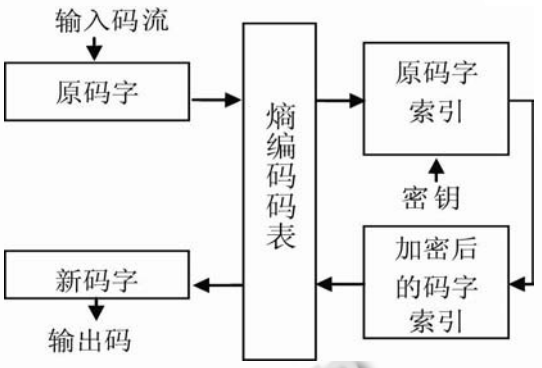


图 1　方案流程图

CAVLC 五个步骤中，每一步都有自己相对应的码表。但是将加密过程植入编码过程时，并不适合在每一步都进行加密：由于非零系数的幅值编码涉及前缀(level_prefix)编码和后缀编码(level-suffix)，比较复杂，即使微小的改动都可能导致无效码字的产生，所以不对此部分进行加密；另外，拖尾系数符号(Trailingones)在码流中所占比例相当小，对此部分的加密对于提高算法的安全性无太大意义，因此，本算法采用在 TotalCoeffs、4×4 block TotalZeros、chroma DC 2×2 block TotalZeros 和 Runbefore 的码表查询过程中对码字索引作保密性处理。

在加密过程中可能会遇到这样的问题：原码表中有些区域内的码字是无效的，但对索引序号加密后可能会出现新索引号落在码表无效区域的情况，因此要对加密后的索引进行判断，若索引号所对应的码字无效，则进行下一步处理。

以 Run_before 码表为例，只有在 row<=7&&column<=15 和 row<=6&&column<=row(row、column 分别代表码字所在码表的行和列)的范围内码字是有效的。假设原码字 P 在码表中原始位置索引为(m,n)，加密后的码字为 C(m’,n’)，其加密处理流程如下所示：

Input original codeword P(m,n)

m’ = encrypt m

n’ = encrypt n

If (n’ >15) then

n’ = n’ mod 15

Else if (m’ < =6) then

If (n’ > m’) then

n’ = n’ mod (m’ + 1)

Else m’ = m’ mod 7

Else keep m’ and n’ as they are

Map the index (m’,n’) back to the Runbefore table

Output the new codeword C(m’,n’)

3 实验结果和性能分析

与视频传输、播放等应用不同，视频存储更注重安全保密性、压缩性能、加密成本等方面的性能特点，而对传输实时性和格式兼容性要求不高。本方案在 H.264 标准模型 JM98 的软件平台上完成加密仿真，通过实验结果和测试对以下性能进行分析。

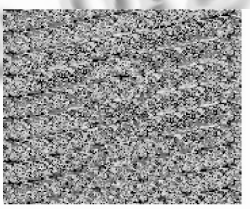
3.1 安全性能

本方案的加密模块可采用经典密码学算法，如 DES、AES、RC4 等。由于视频文件具有数据量大、编码结构复杂的特点，其密码破译难度远大于一般文本文件，因为本方案选用安全度较高但简单易行的流密码算法 RC4。RC4 用 1-256 字节的可变长密钥初始化一个 256 字节的状态表，然后用生成的伪随机流与输入的码字索引相异或。RC4 密钥至少应为 128 位，这就意味着它一次产生的密钥空间大小为 2128。而我们用密钥流对码表执行四次加密操作，很显然在每执行一次加密中都会产生一个新的密钥流。因此对于每一个码字索引来说，其密钥空间大小至少为 2512。同时，本方案所加密的部分涉及视频亮度和色度数据的编码，且主要集中在低频位，没有明显的统计特性，对此部分进行加密不易破译。因此本方案的安全性是可以保证的。

我们对以下几种视频安全方案进行了测试：加密直流系数 DC 和运动矢量 MV、对整个视频文件进行加密、基于熵编码的加密。测试序列为 qcif 格式的 Claire.yuv 序列，取其中某一帧，加密效果如图 2 所示。

(a)原始视频的第 50 帧

(b)加密 DC 和 MV



(c)加密整个视频文件

(d)基于 CAVLC 的加密

图 2　加密效果图

3.2 计算复杂度

本加密方案的计算复杂度主要由流密码和查表时间所决定。从加密码字索引号到映射码表，加密过程所需时间几乎等同于 CAVLC 编码查表过程，对整个编码过程耗时量影响甚小。另外，RC4 算法简单高效，密钥生成和分发所需时间也很小，而且仅仅是码流的逐位异或操作，总的说来算法的计算复杂度比较低。

3.3 压缩性能

本方案加密过程位于码表查询过程，加密的只是码字索引而并不是码字，加密后送入 NAL 单元的码字仍位于码表内，因此对熵编码的统计特性的影响很小。我们用压缩比变化率 Δr 考察算法的压缩性能，Δr 定义为

$$\Delta r = \frac{|r_1 - r_2|}{r_1} \times 100\%$$

其中 r_1 表示进行加密前的视频数据量，r_2 表示进行加密后的视频数据量。表 1 是对几个视频序列进行的压缩性能测试(视频格式：qcif，编码帧数：100)。

3.4 率失真性能

率失真性能是衡量编码质量的标准，它对于码率控制和解密和解码后的图像重建很重要。在率失真曲线(RD 曲线)中，通常以 x 轴代表比特率(bit rate)，以 y 轴代表峰值信噪比(PSNR)。在同一比特率，曲线点

越高意味着图像重建质量越好。以 Claire 序列为例(视频格式：qcif，码流频率：30Hz，量化步长：28)。

表 1 加密前后的压缩比变化

视频序列 (4:2:0)	加密前 Total bits r_1	加密后 Total bits r_2	压缩比增量 Δr (%)
Claire	138192	143520	3.856
Akiyo	149280	154728	3.650
Foreman	507792	526000	3.586

三种加密方案的色度分量率失真曲线如图 3 所示。基于熵编码的加密方案显示了较好的编码质量，这主要是由于加密位置为码字索引而非码字本身，且使用流密码加密可以很好地保持码流控制精度。

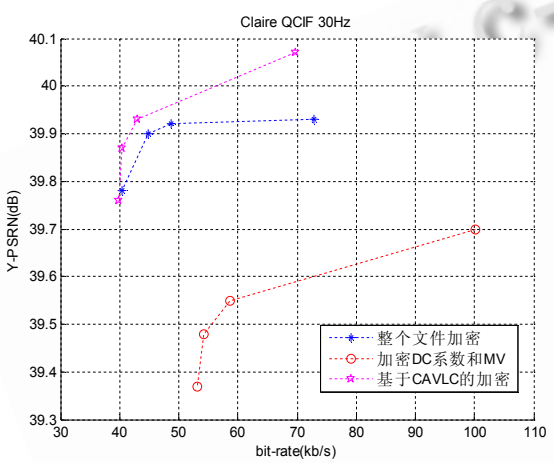


图 3 RD 曲线图

4 结语

本文在研究新一代视频压缩标准 H.264 的基础上设计了一种针对 H.264 熵编码特点的视频加密算法。此算法具有较好的安全保密性，简单高效，对压缩比的影响较小，率失真性能良好。适用于政治、军事、商务机要视频信息以及 PC 机、移动存储设备中的私密视频信息的安全存储管理。

参考文献

1 Riedel E, Kallahalla M, Swaminathan R. A framework for evaluating storage system security. Proceedings of the 1st Conference on File and Storage Technologies, Monterey, CA, January 2002.

2 廉士国,孙金生,王执铨.视频加密算法及其发展现状. 信息与控制, 2004,33(5):560－566.

3 Wen JT, Severa M, Zeng W J. A Format-Compliant Configurable Encryption Framework for Access Control of Multimedia. Multimedia Signal Processing, 2001 IEEE Fourth Workshop on 3－5 Oct. 2001:435－440.

4 Wu C P, Kuo J.C.-C. Efficient Multimedia Encryption via Entropy Codec Design. Proceedings of SPIE International Symposium on Electronic Imaging, 2001, (San Jose, Ca, USA), Jan. 2001,4314.

5 Xie DH, Kuo J.C-C. Enhanced multiple huffamn table(MHT) encryption sheme using key hopping. Circuits and Systems,2005,ISCAS 2005. IEEE International Symposium on 23-26 May 2005,6:5533－5536.

6 余兆明,查日勇,黄磊,周海骄.图像编码标准 H.264 技术.北京:人民邮电出版社, 2005.