H.264 标准中基于 DCT 的视频加密研究

曹 奕 张 荣 刘政凯
(中国科学技术大学电子工程与信息科学系,合肥 230027)

摘 要 随着信息科学和计算机技术的发展,互联网上视频的商业应用,如视频点播、在线电影等变得越来越普及,同时对安全的要求也越来越迫切。为此,针对目前最新的视频压缩标准 H.264 的特点,提出了若干基于 DCT(离散余弦变换)的视频加密算法,采用两个标准视频序列进行了算法仿真,并对各个算法的性能进行了比较分析。本文中所有的试验结果都在为 H.264 专门编写的 JM(联合模型,版本 7.3)软件平台上得到了验证。实验结果表明,这些算法基本不影响压缩效率,且用较低的复杂度得到了较好的加密效果。

关键词 H.264 加密 JM(联合模型) 变换
中图法分类号:TP309.7 TP391.41 **文献标识码**: A **文章编号**: 1006-8961(2005)08-1047-05

Research on DCT-based Video Encryption under H.264

CAO Yi, ZHANG Rong, LIU Zheng-kai
(*Department of Electronic Engineering and Information Science, University of Science & Technology of China, Hefei* 230027)

Abstract With the development of information science and the computer technology, the application of video in commerce on the Internet, such as video on demand or video on line has become more and more common. At the same time, the demand for security becomes more and more urgent. According to the characteristics of the latest video compression standard H.264, this paper proposes several DCT-based encryption algorithms, conducts simulation on two standard video sequences, and analyzes and compares the quality of those proposed algorithms. The experimental results show that the proposed algorithms have no impact on the compression efficiency and have gained good encryption results with low complexity. The simulation results are all tested on JM(joint model, version 7.3) reference software, which is especially programmed for testing the coding standard of H.264.

Keywords H.264, encryption, JM(joint model), transform

1 引 言

近年来,数字视频技术迅猛发展,从视频会议到视频点播,从数字电视到多视点电视,相关的应用层出不穷。可以预见,在不久的将来,数字视频一定会应用于社会的各个行业,走进千家万户。然而,从视频供应商的角度出发,我们面临着一个新的问题:如何保证只有已付费的或已授权的用户才可以接受指定质量的数字视频服务?即安全性问题,也就是视频的加密问题。由于视频信号的数据量巨大,而网络带宽有限,因此它总是以压缩的形式进行传播的。这样,就需要结合压缩标准来设计加密方案,一方面要使加密后的数据与解压格式兼容,另一方面要保证不能影响到数据的压缩效率和有效传输。

传统的多媒体加密技术(如 DES(data encryption standard), VEA (vide encryption algorithm), IDEA (international data encryption algorithm)等)通常要面对两方面的困难:其一是多媒体文件(如视频、语音等等)通常含有很大的信息量;其二是多媒体文件需要通过网络进行实时传输和播放。在大多数多媒体加密系统中,这些多媒体信息被视为纯比特流,采

基金项目:国家自然科学基金重点项目(60333020)
收稿日期:2004-08-30;**改回日期**:2004-12-20
第一作者简介:曹奕(1983 ~),男。2001 年进入中国科学技术大学电子工程与信息科学系攻读学士学位。现主要研究方向为图像处理、多媒体压缩以及多媒体加密。E-mail:yicao@mail.ustc.edu.cn

用经典的 DES 算法[1]加密,加密在压缩编码之后进行。由于 DES 算法复杂,会带来巨大的运算负荷和时延,不能满足多媒体实时传输播放的要求[2,3]。为了尽量减少计算量,采用选择性加密,只对 I 帧加密,P 帧和 B 帧则不加密[2]。但 P 帧和 B 帧中会有帧内预测的宏块,还是存在安全隐患。Tang[4]把加密同压缩过程相结合,使加密在压缩的过程中完成。他把变换后的 64 个系数随机置乱,因此只增加了极少的负荷,但这种置乱破坏了 DCT 系数的统计特性,降低了压缩率。VEA[5]通过于密钥相应位的异或来改变 AC(交流)和 DC(直流)系数的符号位,此法速度很快,适合实时传输,但易受明文攻击,不能提供很高的安全性,只适合代价较低的多媒体应用。Shi[6]用异或只改变帧内宏块的 DC 系数的符号位和运动矢量的符号位,同样很快,但也不能提供很可靠的安全性。Zeng[7]在变换之后量化之前进行置乱加密,在内部路由器内进行转码(transcoding)、解压缩(熵解码和反量化)和再编码(recoding)无需密钥。他置乱 I 帧及 P 帧和 B 帧内的帧内宏块(intra MB),且对运动矢量的符号加密。Kankanhalli[8]把文献[7]中的置乱技术应用到变长编码的 Huffman 表,减少了开销,且没有降低压缩率,提供了合理的安全等级,适用于低开销的置乱系统。

目前最新的压缩编码标准是 H.264,提出这一标准的根本出发点就是在视觉质量保持不变的基础上尽可能多地增大压缩比,以便于窄带网络的多媒体应用。因此,对于 H.264 的加密也不能忽略这一点。在加密过程中,需要尽量简化加密算法,减少运算量,减小密钥的开销,降低对计算机有限资源的占用,减少因加密造成的延迟,避免时间的过多耗费。这样,H.264 对加密算法提出的新要求与多媒体信息的自身特点构成了很难调和的矛盾。然而,对于多媒体信息这一类特殊的信息来说,以上的要求并非不能实现。因为多媒体信息与一般的电子信息(如银行信息、分类文献等)相比:一是价值更低,比特率更高,攻击者不会花很大代价去破译密码,因为破解密码的代价很可能远远超过该多媒体服务的售价;二是由于商业的需要,有些多媒体文件需要有一定的透明度,以吸引未缴费的观众购买该产品。因此,对 H.264 的加密目标应该是在适当的安全性下,使得加密的开销尽可能达到最低。

针对 H.264 的特点,提出一种基于 DCT 系数的加密方案,其包括以下 3 个方面:(1)对 DCT 系数符号的翻盘;(2)对 4×4 块的随机洗牌;(3)高低频系数之间洗牌。试验结果表明了此方案的正确性。

2 加密方案

加密方案如图 1 所示,把加密过程取在量化之后熵编码之前,正是出于尽量简化加密算法,减少运算量,减小密钥的开销且不影响压缩效率的考虑。首先,如果按照传统的方法在量化之前对 DCT 系数加密(如对系数符号的翻盘),那么等于 0 的系数很少,势必需要很多的密钥来分配;然而经过量化以后,很多系数会变成 0,那么分配给这些系数的密钥就完全浪费掉了。其次,量化之前的加密,出于不破坏 DCT 系数自身统计特性和 zigzag 特性(以便于后面的熵编码,且不会对比特率造成很大影响)的考虑,只能在同阶次的系数之间置乱(直流与直流之间,同阶交流之间置乱)。由 DCT 变换的特性知道,这种同阶的置乱效果不是最理想的。然而,量化之后(特别是在 zigzag 排序之后)的置乱就可以在高频和低频系数之间进行,因为熵编码是根据系数的概率进行的,这样就能破坏 DCT 系数的统计特性而不会对压缩比和码率有较大影响,因此能获得非常好的加密效果。另外密钥要经过 DES 加密后传到解密端。

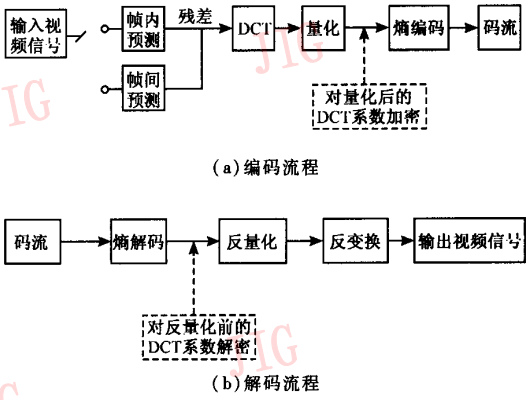

图 1 对 H.264 的加密方案

Fig. 1 Encryption scheme for H.264

H.264 的 DCT 变换对象是原始像素值与预测值的残差。这样,如果由于没有密钥而在解码端重建出一个出错的宏块,那么这一帧内,甚至是另一帧(P 帧或 B 帧)中用这个宏块进行后续预测的块的预测值首先就是错的,由于没有密钥,在预测值上又加上了出错的残差值,这样,解码器还原出来的视频

与原始视频的差别会越来越远。容易想像,H.264 用很少的密钥就能达到很好的加密效果。这种类似“雪崩效应”的连锁反应正是 H.264 环环相扣的编码流程决定的。

算法具体如下:

(1)对 DCT 系数符号的翻盘

假设经变换量化之后宏块中的系数为 $C_{ij}(0 \leqslant i, j \leqslant 15)$,随机生成的密钥为 $K_{ij}(0 \leqslant i,j \leqslant 15, K_{ij} = 1, -1)$,则加密后的系数为

$$D_{ij} = E(C_{ij}) = C_{ij} \times K_{ij}$$

尝试不同的密钥分配方式,如每个 8×8 块公用一个密钥,每个 4×4 块公用一个密钥,每一个非零系数分配一个密钥。虽然符号的翻盘算法没有改变 DCT 系数的统计特性,安全性不高,但它算法简单,易于实现,对时间以及计算机资源耗费很少,对于编码的信噪比、码率没有丝毫影响,所以在需要一定透明度的加密领域其是一种很好的算法。

(2)对 4×4 块的随机洗牌

把一个宏块中每一个 4×4 块作为一个基本单位(H.264 是基于 4×4 的 DCT 变换)进行随机置乱,而 4×4 块内的系数的相对位置不变。

虽然对 4×4 块的随机洗牌也没有改变 DCT 系数的统计特性,但由于系数位置的巨大变动,总体效果要优于符号的翻盘,然而这种优化的效果是以更多加密时间和计算机资源的耗费为代价的。与符号翻盘的另一不同点是,随机洗牌对信噪比和码率有所影响,而且这种影响是随机的,即可能优化信噪比和码率,也可能使其变差。好在这种洗牌是在量化之后,对于熵编码来说,它的影响是微不足道的,即对信噪比和码率的影响都是很小的。

(3)高低频系数之间洗牌

把同一个 4×4 块,8×8 块或 16×16 宏块中的 DCT 系数在高频与低频之间随机置乱,即系数之间的相对位置完全改变。

这种方法完全破坏了 DCT 系数的统计特性,使系数处于及其无序的混乱状态,其效果在理论上与前两种相比是最优的。为了减少密钥的开销,降低对比特率的影响,这种随机置乱只在非零系数之间进行。尽管这样,由于随机置乱的算法并不简单,该法在编码端对时间及计算机资源的占用较大,密钥开销也大,而且置乱范围越大,开销越大。并且在解码器反量化之前的解密也要比前两种加密算法复杂,时间的延迟大。然而对于一些高度机密的视频文件(如会议录像等等)则可以采用该法。

3　实验结果与分析

选取 NL1_Sony_B.yuv 和 NL1_dance.yuv 这两个标准视频序列,在 840MHz PentiumIII 处理器和 256MB RAM 上测试以上涉及的加密算法对信噪比和码率的影响,以及不同算法之间的差别比较。

图 2 为不同加密方案下的加密效果。其不同的加密方案如下:

(1)每一个 8×8 的块公用一个密钥(sign 8)

(2)每一个 4×4 的块公用一个密钥(sign 4)

(3)每一个非零系数有一个密钥 (sign all)

(4)4×4 块的随机洗牌 (block shuffle)

(5)在 4×4 块范围内对系数进行高低阶置乱 (coef shuffle 4)

(6)在 8×8 块范围内对系数进行高低阶置乱 (coef shuffle 8)

(7)在 16×16 块范围内对系数进行高低阶置乱 (coef shuffle 16)

表 1 为对视频序列 NL1_Sony_B.yuv 测试所得的信噪比、码率和运行时间,表 2 为对视频序列NL1_dance.yuv 测试所得的信噪比、码率和运行时间。

表 1　对序列 NL1_Sony_B.yuv 的测试结果

Tab.1　Testing results of the sequence NL1_Sony_B.yuv

	NL1_Sony_B.yuv		
	信噪比(dB)	码率(kb/s)	时间(s)
original	37.36	851.04	22.472
sign 8	37.36	851.04	22.522
sign 4	37.36	851.04	22.993
sign all	37.36	851.04	22.453
block shuffle	37.23	834.88	22.682
coef shuffle 4	37.22	864.24	22.572
coef shuffle 8	37.52	865.84	22.543
coef shuffle 16	37.49	865.92	23.053

表 2　对序列 NL1_dance.yuv 的测试结果

Tab.2　Testing results of the sequence NL1_dance.yuv

	NL1_dance.yuv		
	信噪比(dB)	码率(kb/s)	时间(s)
original	34.69	846.24	19.989
sign 8	34.69	846.24	20.068
sign 4	34.69	846.24	20.029
sign all	34.69	846.24	20.029
block shuffle	34.68	819.36	20.128
coef shuffle 4	34.68	859.92	20.139
coef shuffle 8	34.61	856.80	20.129
coef shuffle 16	34.65	860.56	20.149

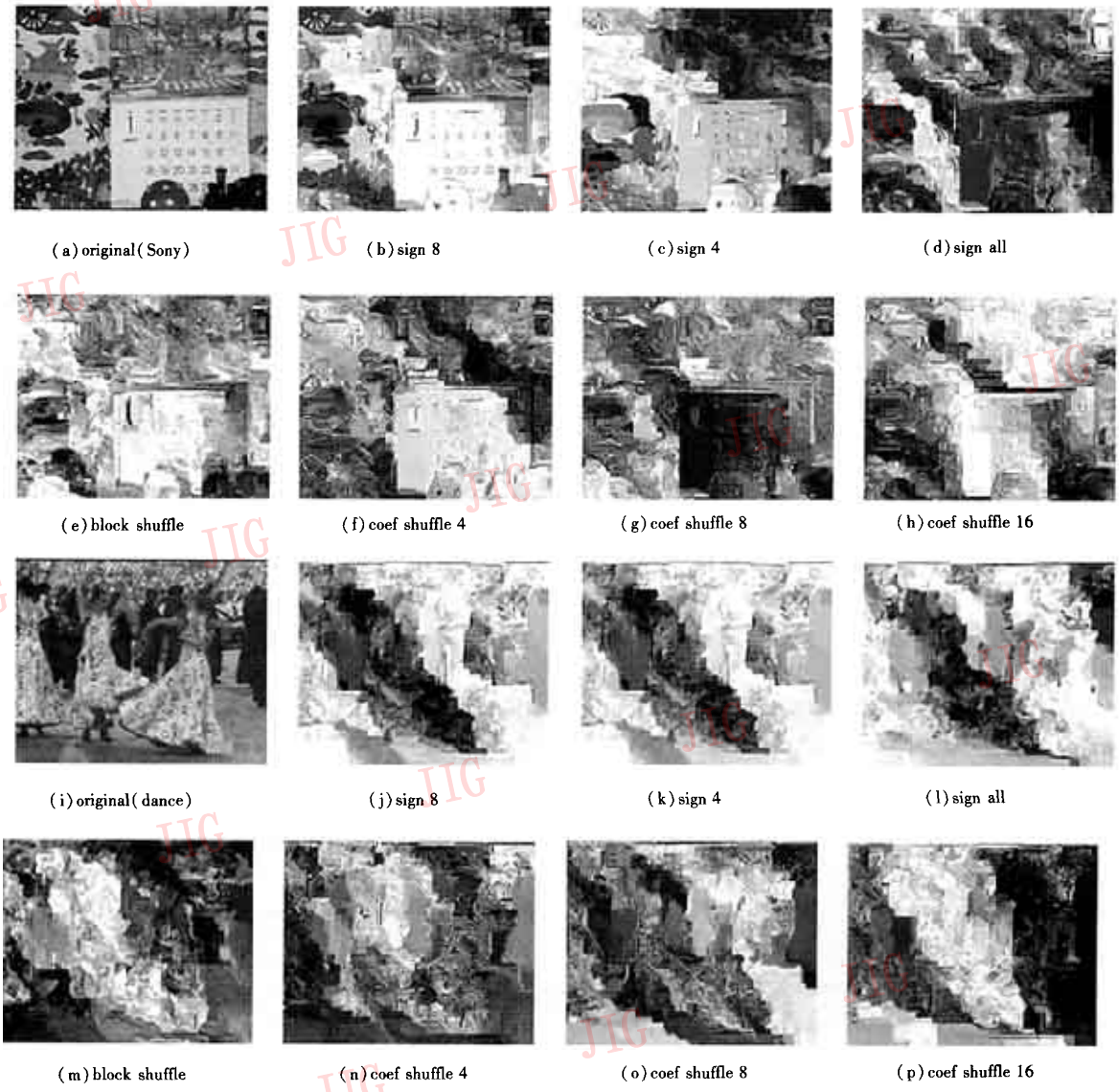

图 2 各种加密方案下的加密效果

Fig. 2 Encryption results of the mentioned encryption schemes

从视觉效果来看,对 NL1_Sony_B. yuv,sign 8 和 sign 4 的效果都不好,而 sign all 就有了明显改善,而且系数的置乱效果明显优于符号的翻盘,这说明好的加密效果要以更大的时间和资源的开销为代价。对于 NL1_dance. yuv 来说,sign 8 的效果已经很好了,后面更复杂的算法的优势就不那么明显了。所以说符号的翻盘不仅简单,易于实现,有些情况下(如画面比较复杂多变)也能获得良好的加密效果。但对于景物单一的画面,还应采用系数置乱的办法以获得预期的安全性。

从压缩效率的影响来看,表 1 和表 2 中的数据显示,对符号的变动不会对信噪比和码率造成任何影响。而效果更优的系数置乱则多少会对其有影响,虽然影响较小。而且可以注意到,在高低阶之间置乱往往会使比特率增加,所以应谨慎使用。

从解密的复杂度来看,对于 sign 8,每一个宏块最多只需 $2^4 = 16$ 次尝试即可破解,安全性最低。对于 sign 4,只需 $2^{16} = 65\,536$ 次,也不安全。加上存在全零的块,实际破解的次数会更少。而对于 sign all,假设量化后的非零系数个数仅占总数的 1/3,一个宏块只

有 85 个非零系数,那破译也需 $2^{85} \approx 3.87 \times 10^{25}$ 次,安全性较高。对于 block shuffle,需要 $16! \approx 2.09 \times 10^{13}$ 次尝试,安全性较高。对于 coef shuffle 4,一个4×4中非零系数约为 $16 \times 1/3 = 5$ 个,整个宏块需要尝试 $16 \times 5! = 1\,920$次,安全性较低,而 coef shuffle 8 需要尝试 $4 \times 21! \approx 2.04 \times 10^{20}$, coef shuffle 16 则需要 $85! \approx 2.8 \times 10^{128}$,安全性都很高。

4 结　论

针对目前最新的视频编码标准 H.264,提出了几种加密算法,并分析比较了它们各自的加密性能。可以看出 H.264 区别于以往编码标准的特点:在视觉质量保持不变的基础上尽可能多的增大压缩比。如果再与运动矢量加密和熵编码加密等相结合,相信会取得更好的加密效果。随着 H.264 的日益普及,相信在不久的将来,对 H.264 的加密应用将会更加广泛。

参考文献(References)

1 Stinson Douglas R. Cryptography, Theory and Practice[M]. New York:CRC Press Inc, 1995.

2 Maples T B, Spanos G A. Performance study of a selective encryption scheme for the security of networked, real-time video[A]. In: Proceedings of the 4th International Conference on Computer Communications and Network[C], Las Vegas, NV, USA, 1995: 2~10.

3 McCanne Steven, Jacobson Van. A flexible framework for packet video[A]. In: Proceedings of the ACM Multimedia95[C], San Francisco, CA, USA, 1995:511~522.

4 Tang Lei. Methods for Encrypting and decrypting MPEG video data efficiently[A]. In: Proceedings of the ACM Multimedia96[C], Boston, USA, 1996:219~229.

5 Shi Changgui, Bhargava Bharat. A fast MPEG video encryption algorithm[A]. In: Proceedings of the Sixth ACM International Conference on Multimeda[C], Bristol, United Kingdom, 1998: 81~88.

6 Shi C, Bhargava B. An efficient MPEG video encryption algorithm[A]. In: Proceedings of the Seventeenth IEEE Symposium on Reliable Distributed Systems[C], West Lafayetle, Indiana, USA, 1998:381~386.

7 Zeng Wenjun, Lei Shawmin. Efficient frequency domain video scrambling for content access control[A]. In: Proceedings of the Seventh ACM International Conference on Multimedia (Part 1)[C], Orlando, Florida, USA, 1999:285~294.

8 Kankanhalli M S, Teo Tian Guan. Compressed domain scrambler/descrambler for digital video[J]. IEEE Transactions on Consumer Electronics, 2002, **48**(2):356~365.

9 LIAN Shi-guo, SUN Jin-sheng, WANG Zhi-quan. Quality analysis of several typical MPEG video encryption algorithms[J]. Journal of Image and Graphics, 2004, **9**(4):483~490.[廉士国,孙金生,王执铨.几种典型视频加密算法的性能评价[J].中国图象图形学报,2004, **9**(4):483~490.]