一种改进的H.264运动估计信息隐藏算法

王 炜[1,2] 林夕杰[1,2] 李孝琴[3]

(解放军信息工程大学计算机学院 郑州 450001)[1] (数学与先进计算国家重点实验室 郑州 450001)[2]

(78020部队 昆明 650229)[3]

摘 要 基于H.264特有的1/4像素精度运动估计,提出一种改进的mH.264运动估计信息隐藏算法。通过修改宏块中每个分割块的最佳匹配位置,利用分割块匹配位置与待嵌二进制信息之间的映射规则,将信息隐藏到分割块的匹配位置中。信息的提取基于解码器中亮度像素内插过程,不需原始视频参与,属于盲提取机制。实验结果表明,改进的H.264运动估计信息隐藏算法在不明显降低视频质量的前提下提高了隐藏容量,降低了系统开销,具有较高的整体性能。

关键词 信息隐藏,H.264,运动估计,1/4像素

中图法分类号 TN919.81 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.6.033

Improved Information Hiding Algorithm Based on Motion Estimation of H.264

WANG Wei[1,2] LIN Xi-jie[1,2] LI Xiao-qin[3]

(School of Computer Science and Technology, PLA Information Engineering University, Zhengzhou 450001, China)[1]

(State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China)[2]

(Unit 78020, PLA, Kunming 650229, China)[3]

Abstract An improved information hiding algorithm based on H.264's specific motion estimation with quarter-pixel precision was proposed. By modifying every sub-block's best matching position of macroblock and using the mapping rule between sub-block's position and binary information, the information is embedded in sub-block's position. Information extraction based on the interpolation process of luma pixels in decoder of H.264 doesn't need the original videos and belongs to the blind extraction mechanism. The experiment results demonstrate that the improved information hiding algorithm in motion estimation of H.264 increases the hiding capacity and decreases the system cost without decreasing the video quality conspicuously, which has higher integer performance than other algorithms.

Keywords Information hiding, H.264, Motion estimation, Quarter-pixel

网络及传输技术的发展,为人类社会的信息传递带来巨大便利的同时,也使得信息在网络传输中容易被不可信的第三方所监控、窃取及篡改。因此,如何提高秘密信息的安全性,成为当今信息安全研究的热点。信息隐藏技术的出现和发展改变了秘密信息在网络传输中的不利局面。基于视频的信息隐藏技术具有隐藏容量大、隐蔽性好等优点,在信息隐藏领域占据重要地位。

作为一种被广泛使用的视频标准,H.264采用多参考帧预测、亚像素运动估计、可变尺寸块帧间预测以及自适应熵编码等独特技术,使得压缩后的视频具有高压缩比、低码率、高质量等优点[1]。另外,由于H.264采用复杂编码技术,对编码过程中的变化非常敏感,因此基于H.264的信息隐藏具有很大的挑战性。

基于H.264进行信息隐藏的方法包括基于帧内预测的隐藏算法[2-4]、基于帧间预测的隐藏算法[5-7]、基于离散余弦变换(DCT)的隐藏算法[8]、基于熵编码的隐藏算法[9,10]等,其中,基于帧间预测的隐藏算法主要利用运动矢量特征(如运动矢量的幅值大小、水平和垂直分量相角大小等)进行信息隐藏。朱洪留等[11]利用H.264特有的1/4像素精度运动估计过程,通过改变运动矢量水平和垂直分量的奇偶性实现信息隐藏,具有较高的隐藏容量,并且对视频质量及码率的影响较小。但为减小对视频码率的影响,该算法未在skip模式的宏块中隐藏信息,而视频编码P帧中存在较多skip宏块,限制了隐藏容量的增加,并且算法对运动矢量的修改及对skip模式的判断在运动估计结束之后,需再次利用运动估计过程并对运动矢量进行修改,使得运动矢量水平和垂直分量的奇偶性满足要求才能实现信息隐藏,大大增加了视频编码时间,增大了系统开销。

为提高基于1/4像素精度运动估计信息隐藏的容量,并降低系统开销,对文献[11]中的算法进行改进,将每个分割块具有的1/4像素精度匹配位置与待嵌二进制信息相映射,通过修改每个分割块的最佳匹配位置,使分割块的最终匹配位置与待隐藏的信息相吻合。为减小对码率的影响,需选择使宏块最小代价函数变化量最小的匹配位置。由于不考虑宏块

到稿日期:2014-05-19 返修日期:2014-11-24 本文受信息工程大学未来基金(1201)资助。

王 炜(1975-),男,博士,讲师,CCF会员,主要研究方向为计算机体系结构、信息安全,E-mail:wang.wei@meac-skl.cn;林夕杰(1987-),男,硕士,主要研究方向为信息安全;李孝琴(1975-),女,硕士,工程师,主要研究方向为信息安全。

是否为 skip 模式，因此可用于隐藏信息的宏块数目高于文献[11]中的算法，提高了隐藏容量，并且整个信息隐藏过程直接在运动估计过程中实现，提高了信息隐藏效率。实验结果表明，本文提出的基于 H.264 1/4 像素精度运动估计的信息隐藏改进算法，能够提高信息隐藏容量，降低系统开销，并且没有对视频质量造成明显影响，具有较高的整体性能。信息提取算法简单、高效，属于盲提取机制，即信息提取时不需要原始视频的参加，只需解码部分码流就能提取信息，并且利用自校验机制保证信息提取的正确性。

1 H.264 运动估计

为提高视频压缩效率，H.264 在已有视频编码标准的基础上增加了 1/4 像素精度的运动估计、可变尺寸快的运动补偿以及多参考帧等新技术[1]。

1.1 运动估计

在一个视频序列中得到分割块运动矢量的过程叫作运动估计[1]，H.264 宏块运动估计时先依次按照 16×16，16×8，8×16的模式将宏块进行分割，所有分割块按一定顺序进行搜索匹配；然后进入 P8×8 模式，依次按照 8×8，8×4，4×8，4×4的模式对每个 P8×8 块进行分割，每个分割块按照一定的顺序进行搜索匹配。H.264 编码器采用 Lagrangian 优化算法实现每个分割块的最佳位置匹配，并得到最佳运动矢量和最佳分割模式。式(1)为 Lagrangian 代价函数表达式[1]。

$$J(\boldsymbol{mv},\lambda_{motion})=SAD\{s,c(\boldsymbol{mv})\}+\lambda_{motion}R(\boldsymbol{mv}-\boldsymbol{p}) \quad (1)$$

其中，s 为当前帧编码数据，c 为参考帧编码数据，$\boldsymbol{mv}$ 为运动估计得到的运动矢量，$\boldsymbol{p}$ 表示通过对相邻宏块预测得到的运动矢量，λ_{motion} 为 Lagrangian 乘数，$R(\boldsymbol{mv}-\boldsymbol{p})$ 为编码运动矢量预测残差所需要的比特数，失真度 SAD 可利用式(2)计算得到，其中，B_x，B_y 的取值可以为 16，8 或 4。

$$SAD\{s,c(\boldsymbol{mv})\}=\sum_{x=1}^{B_x}\sum_{y=1}^{B_y}|s[x,y]-c[x-\boldsymbol{mv}_x,y-\boldsymbol{mv}_y]| \quad (2)$$

1.2 H.264 1/4 像素精度运动估计

有时相邻帧间运动物体进行运动估计时不以整像素为基本单位进行搜索匹配，可能是 1/2 像素、1/4 像素甚至 1/8 像素，这个精度越高，搜索匹配位置越精确，得到的预测值与实际值的误差就越小，码率就越低，即压缩比就越高[12]。

图 1 为一个视频序列分别采用 1/2、1/4 和 1/8 像素精度进行运动估计得到的压缩视频的信噪比与码率的关系。可以发现，1/4 像素精度的运动估计相比 1/2 像素精度具有较高的编码效率；但 1/8 像素精度的运动估计并没有比 1/4 像素精度的编码效率高多少，而且其计算复杂度较高，这对编解码器的编解码能力提出更高的要求。因此，在 H.264 编码标准的制定过程中，只保留了 1/4 像素精度的运动估计[1]。

图 2 为宏块或分割块进行运动估计的模型图。两个整点像素位置(如图 2 中的 0 像素)之间以 1/4 像素为单位分割，中间位置叫做半像素位置(如图 2 中的 1—8 像素)，半像素与整点像素的中间位置叫做 1/4 像素位置(如图 2 中的 a—h 像素)。H.264 运动估计的块搜索匹配过程分为 3 步：首先进行整点像素位置的搜索匹配，根据式(1)运算得到分割块在整点像素位置所具有的运动矢量 $\boldsymbol{mv}$ 及代价函数值 J；然后进行半像素位置的搜索匹配；最后是 1/4 像素位置的搜索匹配。最终使得 Lagrangian 代价函数最小的运动矢量具有 1/4 像素精度，使得 H.264 相比于早先的视频压缩标准能获得较高的预测精度，从而获得更好的视觉质量。

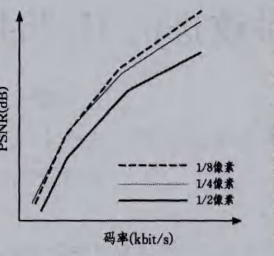


图 1 运动估计精度与编码效率关系

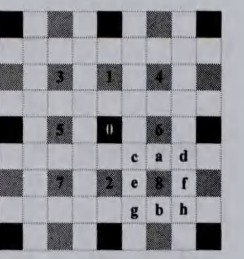


图 2 运动估计的模型图

2 基于 1/4 像素精度运动估计的隐藏算法

本文首先建立 1/4 像素精度运动估计过程中每个分割块的匹配位置与二进制信息之间的映射关系，通过修改分割块的最佳匹配位置，将信息隐藏到修改后的匹配位置中，每个帧间预测宏块中的分割块都隐藏信息，整个隐藏过程在亮度块的匹配位置搜索过程中完成，即只将信息隐藏在亮度块中。由于仅在水平或垂直方向移动分割块 1/4 像素距离，因此能最大限度减少对编码特征的影响。

2.1 二进制信息映射规则

分割块的最佳匹配位置与二进制信息之间的映射规则如图 3 所示。以图 3(a)中 P 点(整像素点或半像素点)为中心，联合周围的 8 个 1/4 像素点组成方形区域，将 0 和 1 间隔分配，如图 3(b)所示。

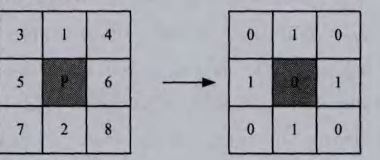


图 3 信息映射模型

将每帧图像以 1/4 像素为单位进行分割，并按照图 3 的映射规则，对整幅图像进行 0 和 1 的间隔分配，如图 4 所示。可以发现，整像素点和半像素点处映射的信息相同，其周围的数据都按照同样的规律分配，即任何分割块的任何匹配位置都能映射一个数据。这种信息映射规则有利于简单、快速、高效地隐藏信息。

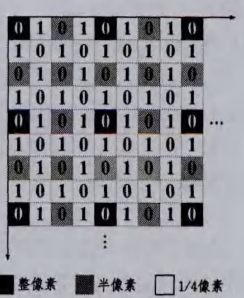


图 4 1/4 像素精度的信息映射规则模型

2.2 隐藏规则

H.264 的运动估计过程结束后，分割块的最佳匹配位置落在图 3(a)组成的点集{P,1,2,3,4,5,6,7,8}的某一点。设待嵌二进制信息为 B，信息隐藏算法如下。

输入：隐藏信息 B

输出：隐藏信息后分割块的最佳匹配位置 best_pos、最佳运动矢量 **mv** 以及最小代价函数 mcost(由式(1)得到)

(1)最佳匹配位置映射信息与隐藏信息相等时：

```
1. set best_pos[9]={P,1,2,3,4,5,6,7,8};
2. if best_pos ∈{1,2,5,6} and B=1 then
3.      output best_pos,mv,mcost;
4. end
5. else if best_pos ∈{P,3,4,7,8}and B=0 then
6.      output best_pos,mv,mcost;
7. end
```

(2)最佳匹配位置映射信息与隐藏信息不相等时：

```
1. set best_pos[9]={P,1,2,3,4,5,6,7,8};
2. int best_pos_changed,mv_changed;
3. if best_pos ∈{1,2,5,6} and B=0 then
4.        best_pos=best_pos_changed;
5.        mv=mv_changed;
6.      output best_pos,mv,mcost;
7. end
8. else if best_pos ∈{P,3,4,7,8} and B=1 then
9.        best_pos=best_pos_changed;
10.       mv=mv_changed;
11.     output best_pos,mv,mcost;
12. end
```

当最佳匹配点 *best_pos* 映射信息与隐藏信息 B 相等时，不修改 *best_pos* 及运动矢量 ***mv*** 的初始值；当 *best_pos* 映射信息与 B 不相等时，用修改后的匹配位置 *best_pos_changed* 和修改后的运动矢量 *mv_changed* 代替初始的 *best_pos* 和 ***mv***。其中，*best_pos_changed* 的值与表 1 中的 *pos_changed* 值相关，*pos_changed* 代表 *best_pos* 可能修改为的值。结合图 3(a)可发现，*pos_changed* 的值都是由最佳匹配点在水平或垂直方向移动 1/4 像素距离得到的。

由表 1 可知，*pos_changed* 有 2 个、3 个或 4 个值可以选择，根据式(3)、式(4)选择一个最佳 *pos_changed* 作为 *best_pos_changed*。

$$\Delta mcost_{changed} = mcost_{pos\_changed} - mcost \tag{3}$$

表 1 *best_pos_chaged* 取值规则

隐藏信息	best_pos	best_pos_changed
B=0	P	
	1	P 或 3 或 4
	2	P 或 7 或 8
	3	
	4	
	5	P 或 3 或 7
	6	P 或 4 或 8
	7	
	8	
B=1	P	1 或 2 或 5 或 6
	1	
	2	
	3	1 或 5
	4	1 或 6
	5	
	6	
	7	2 或 5
	8	2 或 6

$$best\_pos\_changed = \{pos\_changed \mid \min\langle \Delta mcost_{changed} \rangle\} \tag{4}$$

其中，$mcost$ 为最小代价函数，$mcost_{pos\_changed}$ 为 *pos_changed* 处的代价函数，$\Delta mcost_{changed}$ 为代价函数之差。由式(4)可得，使得 $\Delta mcost_{changed}$ 最小的 *pos_changed* 值作为 *best_pos_changed*，即选择最接近最小代价函数的匹配位置隐藏信息，从而对预测精度等编码特征造成的影响达到最小。

由上述可知，通过改变分割块的 *best_pos* 值来实现信息隐藏，改变了 ***mv***，据式(1)和式(2)可知，宏块的分割模式也可能发生改变：从普通的帧间宏块转变为帧内 I4 或 I16 宏块，或从 I4 或 I16 宏块变为普通的帧间预测块。本文将在2.3节中讨论利用 *get_block*()函数提取信息，*get_block*()仅对帧间宏块进行像素内插，并不对帧内 I4、I16 宏块进行像素内插，因此，若宏块经过隐藏过程后变为 I4 或 I16 宏块，则表示信息隐藏不成功，需选择下一个宏块继续对该信息进行隐藏；若宏块不为 I4 或 I16，则表示信息隐藏成功。最终信息隐藏流程如图 5 所示。

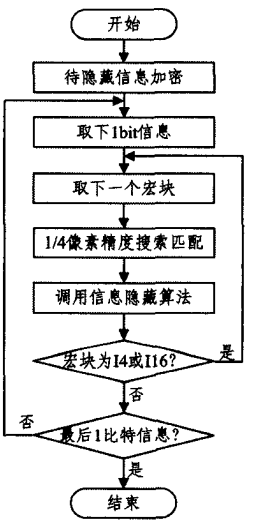


图 5 信息隐藏流程图

隐藏信息随码流经过网络传输，可能受到网络环境的干扰，使得提取的信息与原始信息不同，从而破坏信息的正确性。为了解决这一问题，在信息隐藏过程中，将一个宏块中的所有分割块的最佳匹配位置修改到映射相同二进制数的位置，即一个宏块表示隐藏 1 比特数据信息，从而将信息的自校验功能融入到信息隐藏过程。

为了提高隐藏信息的安全性，可首先对信息进行加密处理，然后利用 Logistic 混沌映射算法[13] 随机地选择隐藏信息的宏块，将信息分散到整个视频空间中，因为在一般情况下，要隐藏的信息量相对于整个视频信息量来说非常小。这样可以保证视频质量不会因为局部的失真而影响整体，又可以提高隐藏信息的隐蔽性和抗分析能力。Logistic 的数学表达公式如下：

$$x_{n+1} = x_n \times \mu \times (1 - x_n) \tag{5}$$

其中，$\mu \in [0,4]$为 Logistic 参数，当 $n=0$ 时，x_0 为初始值。当 $x \in [0,1]$时，Logistic 的映射为混沌随机状态。当 μ 非常接近 4 时，迭代生成的 x_n 处于随机状态且平均分布在整个 0 到 1 的区间内。因此，为使得宏块选择更具随机性，选取的 μ 越接近 4 越好。

2.3 基于亮度像素内插过程的隐藏信息提取

根据已提出的隐藏算法，利用解码器中亮度像素内插函

数 *get_block*()的工作原理[14]提取信息。亮度像素内插是解码器根据参考图像对解码图像的帧间预测宏块进行重建的过程，以4×4亮度块为单位，同样具有1/4像素精度。通过对每个4×4亮度块的重建位置的确定，即可提取出重建位置所对应的二进制信息。

信息提取过程如下：

Step1　解码得到宏块中4×4亮度块的运动矢量 ***mv***(以1/4像素为单位)。

Step2　根据运动矢量 ***mv***，利用式(6)、式(7)计算得到4×4亮度块分别在水平和垂直方向相对于整像素的偏移位置 x_pos 和 y_pos。其中，$\boldsymbol{mv}[x]$ 和 $\boldsymbol{mv}[y]$ 分别为运动矢量 ***mv*** 的水平和垂直分量。

$$x\_pos=\boldsymbol{mv}[x]\&3 \tag{6}$$

$$y\_pos=\boldsymbol{mv}[y]\&3 \tag{7}$$

Step3　根据偏移位置坐标(x_pos，y_pos)确定重建像素在图6中的位置，然后根据图7的信息映射关系图，记录每个4×4亮度块内插位置所对应的信息。如图7所示，当 $x\_pos=3$、$y\_pos=2$ 时，根据图6映射的信息为1。以此类推，将每个宏块中的16个4×4亮度块映射的16个数据存放在数组 $B[16]$ 中。

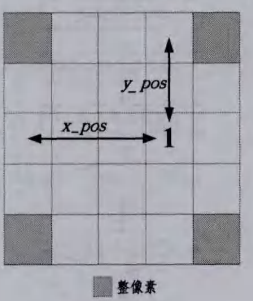


图6　信息提取模型

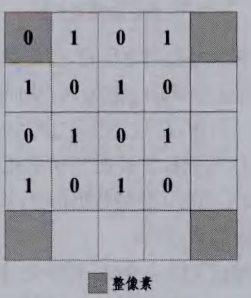


图7　亮度像素内插位置与二进制信息映射关系

为保证一个宏块经网络传输后信息提取的正确性，要求一个宏块中的所有分割块在信息隐藏时都映射到同一个二进制信息，即数组 $B[16]$ 代表一个比特数据 B。因此，利用以下规则进行信息的提取。

若 $B[16]=\underbrace{\{1,1,1,\cdots\cdots,1,1,1\}}_{16个1}\Rightarrow B=1$

若 $B[16]=\underbrace{\{0,0,\cdots\cdots,0,0,0\}}_{16个0}\Rightarrow B=0$

3　性能分析

本算法利用 H.264/AVC 参考软件 JM8.6[15]进行仿真实验。实验中对4个 QCIF 格式的视频测试序列(foreman，soccer，city，ice)进行了信息隐藏和提取，测试视频序列分辨率都为176×144。编码器的一些主要参数设置如表2所列。

表2　JM 重要编码参数设定

参数名称	设定值
编码档次	Baseline
帧率	30帧/秒
Y:U:V	4:2:0
量化参数(QP)	28
FrameSkip	0
编码帧视频结构	IPPP
编码视频帧数	6

实验主要从视频客观质量 PSNR、P 帧编码数 PTB(P-TotalBits)、码率 BR(Bit-Rate)、运动估计时间 MET(Motion Estimate-Time)和视频编码时间 ET(Encoding-Time)等方面对隐藏算法进行评估。

3.1　实验结果

实验中设置6帧编码帧，由于编码视频结构为 IPPP，则第1帧为 I 帧，后5帧为 P 帧。因此，只有后5帧可以进行信息隐藏。

定义1　R_1 为视频隐藏信息后相对于原始视频关于 PSNR 的变化率：

$$R_1=\frac{PSNR_{hidden}-PSNR_{original}}{PSNR_{original}}\times 100 \tag{8}$$

其中，$PSNR_{original}$ 为原始视频信噪比，$PSNR_{hidden}$ 为隐藏信息的视频信噪比。Y、U 和 V 分别为 PSNR 的亮度分量和两个色度分量。视频序列各分量信噪比如图8所示。

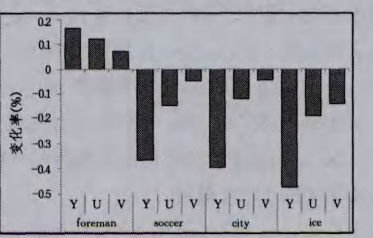


图8　各视频序列 Y、U、V 分量信噪比

由图8可知，隐藏信息后，soccer、city 和 ice 视频序列的 Y 分量信噪比变化率在0.3%～0.5%之内，U 分量和 V 分量信噪比变化在0.2%以内，表明信息隐藏对视频信噪比影响很小。foreman 视频中各个分量的信噪比有所升高，说明信息隐藏后视频的质量提高了。这是因为宏块最小代价函数的计算主要利用了 SAD 原则，分割块进行搜索匹配，具有局部最优的特点，而非全局最优。故存在优于 H.264 运动估计得到的搜索匹配，且具有较高的视频质量。

定义2　R_2 为视频隐藏信息后相对于原始视频关于 PTB、BR、MET 和 ET 的变化率：

$$R_2=\frac{M_{hidden}-M_{original}}{M_{original}}\times 100 \tag{9}$$

其中，$M_{original}$ 为原始视频的各参数值，M_{hidden} 为隐藏信息视频的各参数值。由图9可知，foreman 序列隐藏信息后，PTB 的值减小，说明在这一视频序列中隐藏信息后得到整体的匹配精度高于隐藏信息之前的匹配精度，从而需要更少的比特数去编码 P 帧，这与图8中 foreman 序列隐藏信息后信噪比增加相吻合。而其他视频 PTB 的值增加，表明信息隐藏导致整体匹配精度降低，需更多的比特数去编码 P 帧。由于限定了帧率，因此 BR 随着 PTB 的变化而变化。所有视频序列的 MET 都增加，增加的部分主要集中在算法对分割块最佳匹配

位置映射信息的判断和对最佳匹配位置的修改，变化率在1%左右；但对整体的视频编码时间影响较小，变化率在0.5%左右。

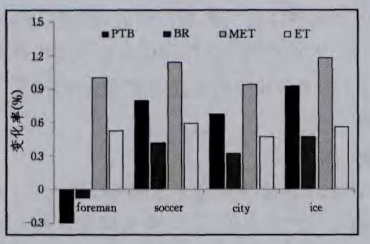
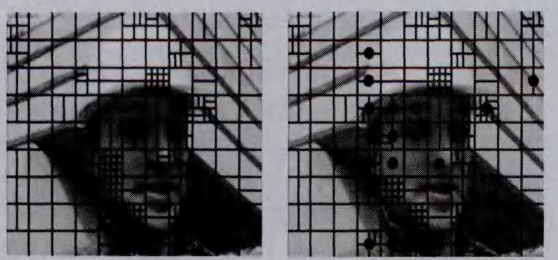


图9　信息隐藏后各个编码参数的变化率

每个宏块的分割模式与最小代价函数相关。信息隐藏改变分割块的最佳匹配位置，则宏块的失真度和宏块需要编码的比特数发生变化，根据式(1)可知，宏块原有的最小代价函数也随之改变，若修改后的代价函数小于其他分割模式的代价函数，则将改变宏块的分割模式。图10为foreman序列第2编码帧的宏块分割模式在信息隐藏前后的比较，圆点标注的为分割模式改变的宏块。结果表明，分割模式改变的宏块集中在图像精细区域。因为精细区域需要更为精细的划分，需要更高的预测精度，宏块的各种分割模式的代价函数更接近，故修改精细区域分割块最佳匹配位置更易改变宏块的分割模式。

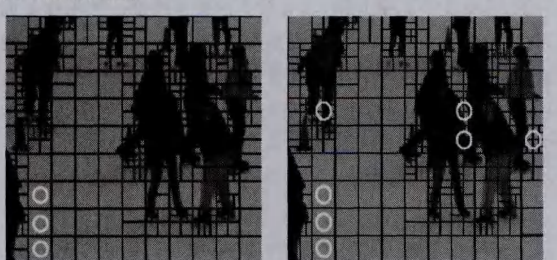


(a)原始视频　　(b)隐藏信息视频

图10　隐藏信息前后的宏块分割模式的变化

3.2　信息隐藏容量分析

图11为视频ice第2帧在信息隐藏前后的I4和I16宏块的变化情况，白色圆圈标注的为I4或I16宏块。其中，图11(a)为信息隐藏前的3个I16宏块，图11(b)为信息隐藏后的3个I16宏块和4个I4宏块。说明：有些帧间预测宏块经信息隐藏所引起的最佳匹配位置的改变，转变为I4或I16宏块。因此，图11(b)中的7个帧内宏块不能用于隐藏信息。为验证视频序列能够隐藏的最大信息量，本算法在所有的帧间宏块中隐藏信息。



(a)原始视频　　(b)隐藏信息视频

图11　信息隐藏前后I4和I16宏块的变化

图12为视频序列的提取信息数。表2中设置的视频总帧数为6帧，视频帧编码结构为：IPPPPP，因此，所有P帧的宏块数为((6−1)×(176×144)/(16×16))=495。理想状态下P帧中无I4和I16宏块，视频序列提取信息应为495bit。但信息隐藏过程后P帧中存在一定数目的I4和I16宏块，故提取信息少于495bit。为避免出现宏块数目不足的情况，隐藏信息之前应尽可能选择较大的视频序列。

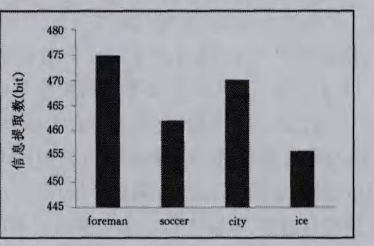


图12　视频提取信息数

3.3　性能比较

本文方法与文献[11]中的算法具有不同的信息隐藏机制和隐藏过程，因此两种算法具有不同的性能，例如信息隐藏后视频的信噪比PSNR、码率BR、隐藏容量(Hiding Capacity，HC)及编码时间ET等。

定义3　K表示本文算法与文献[11]中的算法的相对性能百分比：

$$K=\frac{\alpha}{\beta}\times 100 \tag{10}$$

其中，α和β分别为本文算法和文献[11]中的算法得到的PSNR、BR、HC和ET的值。

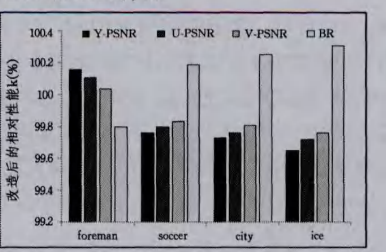
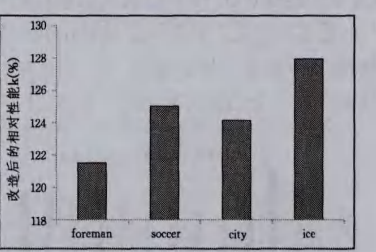
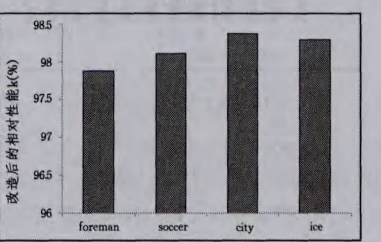


图13　改进前后信噪比与编码效率



(a)隐藏容量



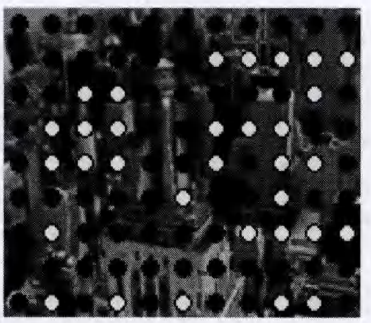
(b)编码时间

图14　改进前后的隐藏容量与编码时间

如图13所示，视频序列soccer，city和ice利用两种算法进行隐藏信息后的信噪比相差不大，K在99.7%左右，说明基于1/4像素精度运动估计隐藏信息对视频质量影响非常

小。但本算法的信噪比稍低，因为隐藏容量与视频客观质量之间具有相互制约的关系，隐藏容量越大，则视频质量越低，即信噪比越低[1]。本文算法隐藏容量较大，需要修改的分割块最佳匹配位置较多，因此帧间预测产生的误差较大，从而需要更多的比特数去编码 P 帧，而帧率一定，因此本文算法的码率较高。而序列 foreman 信噪比的 K 超过 100%，并且码率下降，与图 8 中的 foreman 序列信噪比变化相对应。

如图 14(a)所示，改进后算法所有视频序列的隐藏容量提高了 23%左右，因为在 skip 宏块中也进行信息的隐藏，而文献[11]中的算法未利用 skip 宏块。视频编码 P 帧中含有大量的 skip 宏块，如图 15(city 序列第 2 帧)所示。因此，P 帧中 skip 宏块数目越多，本算法相对于文献[11]中的算法的隐藏容量越大。



●非 skip 宏块　　○skip 宏块

图 15　city 序列 P 帧宏块类型分布

如图 14(b)所示，视频编码时间 ET 降低了 2%左右。因为文献[11]中的算法进行运动估计，然后判断宏块类型，再次利用运动估计过程进行信息隐藏，而本文算法直接在运动估计过程中实现信息隐藏，因此效率更高，所需编码时间更少。

为了衡量隐藏算法的整体性能，定义整体性能 ρ 如下：

$$\rho=\frac{P(\mathrm{dB})\times C(\mathrm{bit})}{R(\mathrm{kbit/s})\times T(\mathrm{s})} \tag{11}$$

其中，P 和 C 分别代表信息隐藏后的信噪比和隐藏容量，数值越高说明算法性能越好；R 和 T 分别代表信息隐藏后的码率和编码时间，数值越低说明算法性能越好；ρ 代表算法整体性能，数值越高，则算法整体性能越高。图 16 为两算法整体性能比较，其中参数 P 分别取信噪比的 Y、U、V 分量。

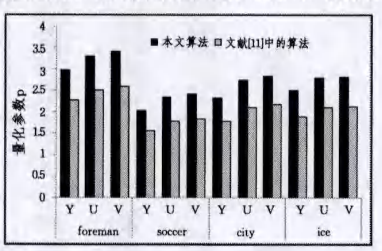


图 16　两算法整体性能比较

由图 16 可见，本文算法的量化参数 ρ 大于文献[11]中的算法，本算法的整体性能较高，主要原因在于增加隐藏容量和降低系统开销两方面，并且保持对视频客观质量和码率较小的影响，视觉效果没有变化。

3.4　安全性描述

信息在隐藏之前先经过加密处理，将加密后的密文隐藏到视频中。另外，在一般情况下，要隐藏的信息量相对于整个视频信息量来说非常小，因此，可以充分利用视频空间和随机选择机制，随机地选择用于隐藏信息的帧间预测块，将隐藏的信息分散到整个视频空间域，而不是将信息隐藏到局部的小空间中，即将极小嵌入率的信息分散到较大的视频流中，从而提高了隐藏信息分布的随机性，并可以保证视频质量不会因为局部的失真而整体受到影响。因此，隐藏信息的安全性取决于隐藏前的加密密钥的选择和信息隐藏随机性的特征和规律，即利用信息加密和隐藏位置的随机选择，提高了隐藏信息的隐蔽性和安全性。

3.5　鲁棒性测试

对于 H.264 码流在网络中传输出现的丢包、丢帧现象从而破坏隐藏信息的完整性，本算法利用重复嵌入机制将每比特信息段重复嵌入到不同的帧间预测宏块中，当隐藏信息的某些宏块因丢包、丢帧而丢失时，可用备份信息所在的宏块恢复丢失信息。

以 football 序列为例，待隐藏信息为 50bit，分别将其重复嵌入 1 次、2 次和 3 次，根据文献[16]中的丢包算法，测试在不同丢包率下的信息恢复率，如图 17 所示。

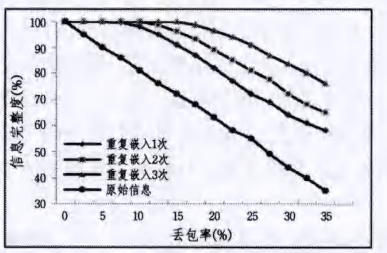


图 17　丢包率及信息完整度

由图 17 可知，对于未经过重复嵌入的隐藏信息，随着丢包率的升高，信息完整度成线性趋势快速下降。重复嵌入 1 次后，当丢包率小于 5%时，信息恢复率能够保持 100%，依然能够保持提取信息的完整性；重复嵌入 2 次后，当丢包率小于 10%时，信息恢复率能够保持 100%；重复嵌入 3 次后，当丢包率小于 15%时，信息回复率能够保持 100%。因此，重复嵌入次数越高，隐藏信息具有的鲁棒性越高。所以，重复嵌入也可以抵御一定的剪切攻击。

结束语　基于 H.264 的信息隐藏技术在信息安全领域发挥着越来越重要的作用，国内外对其展开了相当广泛的研究。在 H.264 特有的 1/4 像素精度运动估计过程中隐藏信息具有多方面优势，如隐藏容量大、对视频质量影响小等。本文在已有隐藏算法的基础上对其进行改进，将每个分割块具有的 1/4 像素精度匹配位置与待嵌二进制信息相映射，通过修改每个分割块的最佳匹配位置，使分割块的最终匹配位置与待隐藏信息相吻合，从而实现信息隐藏。提高了信息隐藏容量，并减少了系统开销，具有较高的整体性能。利用重复嵌入机制，在一定程度上能提高隐藏信息的鲁棒性。信息的提取算法基于解码器中亮度像素的内插过程，对宏块中所有 4×4亮度块的重建位置进行分析即可提取信息，方法简单、快速，属于盲提取机制。

参 考 文 献

[1]　毕厚杰，王健. 新一代视频压缩编码标准——H.264/AVC[M]. 北京：人民邮电出版社，2009：15-96
Bi Hou-jie, Wang Jian. A New Standard for Video Compression Coding——H.264/AVC[M]. Beijing: Posts & Telecom Press. 2009: 15-96

[2]　Bouchama S, Hamami L, Aliane H. H.264/AVC Data Hiding

Based on Intra Prediction Modes for Real-time Applications [OL]. 2012-04http://www.iaeng.org/publication/WCECS2012/WCECS2012_pp655-659.pdf

[3] Mehmood N, Mushtaq M. Blind Watermarking Scheme for H.264/AVC Based on Intra 4x4 Prediction Modes[J]. Lecture Notes in Electrical Engineering, 2012, 179: 1-7

[4] Yang Gao-bo, Li Jun-jie, He Ying-liang, et al. An information hiding algorithm based on intra-prediction modes and matrix coding for H.264/AVC video stream[J]. AEU-International Journal of Electronics and Communications, 2011, 65(4): 331-337

[5] Ko Man-Geun, Hong Jang-Eui, Suh Jae-Won. H.264/AVC error detection scheme using fragile data hiding in motion vector set [C]// 2012 IEEE International Conference on Consumer Electronics (ICCE). 2012: 237-238

[6] Su Yu-ting, Zhang Cheng-qian, Zhang Chun-tian. A video steganalytic algorithm against motion-vector-based steganography [J]. Signal Processing, 2011, 91(8): 1901-1909

[7] Jing Hui-yun, He Xin, Han Qi, et al. Motion Vector Based Information Hiding Algorithm for H.264/AVC against Motion Vector Steganalysis[J]. Intelligent Information and Database Systems Lecture Notes in Computer Science, 2012, 7197: 91-98

[8] 刘争艳. H.264/AVC视频流的信息隐藏技术研究[D]. 长沙: 湖南大学, 2009
Liu Zheng-yan. Researches on H.264/AVC Video Streaming Information Hiding Technology[D]. Changsha: Hunan University. 2009

[9] Kim Sung-Min, Kim Sang-Beom, Hong Youp-yo, et al. Data Hiding on H.264/AVC Compressed Video[J]. Lecture Notes in Computer Science, 2007, 4633: 698-707

[10] L in Xi-jie, Li Qing-bao, Wang Wei, et al. Information Hiding Based on CAVLC in H.264/AVC Standard[C]// 2012 Fourth International Conference on Multimedia Information Networking and Security (MINES). 2012: 900-904

[11] Zhu Hong-liu, Wang Rang-ding, Xu Da-wen. Information hiding algorithm for H.264 based on the motion estimation of quarter-pixel[C]// 2010 2nd International Conference on Future Computer and Communication(ICFCC). 2010: 423-427

[12] 马兰, 沈笑云, 等. 精通 Visual C++视频/音频编码技术[M]. 北京: 人民邮电出版社. 2008: 42-45
Ma Lan, Shen Xiao-yun, et al. Proficient in video/audio coding technology with VisualC++[M]. Beijing: Posts & Telecom Press. 2008: 42-45

[13] Logistic 混沌映射[EB/OL]. 2008-03-10. http://blog.sina.com.cn/s/blog_4511c21f01008p0k.html
Logistic Chaos Map [EB/OL]. 2008-03-10. http://blog.sina.com.cn/s/blog_4511c21f01008p0k.html

[14] jm86之get_block() 1/4亮度像素内插详述[EB/OL]. 2007-02-04. http:// blog.csdn.net/zhangji1983/article/details/1502035
The detail of the 1/4 pixels interpolation with get_block() in jm86[EB/OL]. 2007-02-04. http:// blog.csdn.net/zhangji1983/article/details/1502035

[15] JVT Reference Software version JM8.6[EB/OL]. 2012-04. http:// iphome.hhi.de/suehring/tml/download/old_jm/

[16] 唐贵进, 朱秀昌. H.264 JM模型中丢包算法的研究及改进[J]. 数字视频, 2011, 35(13): 6-8, 27
Tang Gui-jin, Zhu Xiu-chang. Research and Improvement of Packet Loss Algorithm in JM Model of H.264[J]. Digital Video, 2011, 35(13): 6-8, 27

(上接第144页)

[9] Ahmadi H, Pham N, Ganti R, et al. Privacy-aware regression modeling of participatory sensing data[C]// Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems. ACM, 2010: 99-112

[10] 朱晓玲. VANET安全和隐私保护机制研究[D]. 合肥: 合肥工业大学, 2013
Zhu Xiao-ling. Security and Privacy Preservation Mechanisms in Vehicular Ad Hoc Network[D]. Hefei: Hefei University of Technology, 2013

[11] Sweeney L. K-anonymity: a model for protecting privacy. International Journal on Uncertainty[J]. Fuzziness and Knowledge-based Syatems, 2002, 10(5): 557-570

[12] Camenisch J, Groth J. Group Signatures: Better Efficiency and New Theoretical Aspects[C]// SCN 2004, LNCS 3352, 2005: 120-133

[13] Agrawal R, Evfimievski A, Srikant R. Information Sharing Across Private Databases[C]// Proceedings of SIGMOD. 2003: 86-97

[14] Xie Q, Hengartner U. Privacy-Preserving Matchmaking for Mobile Social Networking Secure Against Malicious Users[C]// Proceedings 9th Int. Conf. on Privacy, Security. 2011: 252-259

[15] Li S D, Wang D S, Lou P, et al. Symmetric Cryptographic Solution to Yao's Millionaires' Problem and an Evaluation of Secure Multiparty Computations[J]. Information Sciences, 2008, 178(1): 244-255

[16] Liu R X, Lin X D, Liang X H, et al. Secure Handshake with Symptoms-matching[C]// The Essential to the Success of mHealthcare Social Network. BodyNets 2010, Corfu Island, Greece, September 2010: 10-12

[17] Chaum D. Blind signature system[C]// Advances in cryptology. Springer US, 1984: 153-153

[18] Bloom B. Space/time trade-offs in hash coding with allowable errors[J]. Comm. ACM, 1970, 13(7): 422-426

[19] Broder A, Mitzenmacher M. Network applications of bloom filters: A survey[J]. Internet Mathematics, 2005, 1(4): 485-509

[20] Zhao Y, Wu J. B-sub: A practical bloom-filter-based publish-subscribe system for human networks[C]// 2010 IEEE 30th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2010: 634-643

[21] Gremillion L L. Designing a Bloom filter for differential file access[J]. Communications of the ACM, 1982, 25(9): 600-604

[22] Qiu Ling, Li Ying-jiu, Wu Xin-tao. An Approach to Outsourcing Data Mining Tasks While Protecting Business Intelligence and Customer Privacy[C]// Proc. of the 6th IEEE International Conference on Data Mining. Hong Kong, China: [s.n.], 2006: 551-558

[23] Zhang Y, Ren K. DP2AC: Distributed Privacy-Preserving Access Control in Sensor Networks[C]// INFOCOM 2009. IEEE, 2009: 1251-1259

[24] Sun J, Zhang R, Zhang Y. Privacy-preserving spatiotemporal matching[C]// 2013 Proceedings IEEE INFOCOM. IEEE, 2013: 800-808