

## 低复杂度的 H.264 帧间编码的模式判决算法

张冬明<sup>1,2</sup> 沈燕飞<sup>1</sup> 林守勋<sup>1</sup> 张勇东<sup>1</sup>

<sup>1</sup>(中国科学院计算技术研究所 北京 100080)

<sup>2</sup>(中国科学院研究生院 北京 100049)

(dmzhang@ict.ac.cn)

### Low Complexity Mode Decision for H.264 Inter Frame Encoding

Zhang Dongming<sup>1,2</sup>, Shen Yanfei<sup>1</sup>, Lin Shouxun<sup>1</sup>, and Zhang Yongdong<sup>1</sup>

<sup>1</sup>(Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080)

<sup>2</sup>(Graduate University of Chinese Academy of Sciences, Beijing 100049)

**Abstract** The new video coding standard, H.264 gives a better encoding performance than previous video standards at the cost of expensive computation since it allows motion estimation performing on tri-tree structured macroblock partitioning and multiple reference frames. In this paper, a fast algorithm FIMDA is proposed to accelerate mode decision of inter macroblock. FIMDA makes full use of valuable cues provided by the previous searching reference frames, such as mode, rate distortion cost etc., to eliminate unnecessary modes and reference frames in the following searching. FIMDA can effectively reduce encoding complexity and the quality degradation compared with full search can be ignored. Simulation results show that average complexity reduction exceeds 85% and average quality degradation is only about 0.07dB.

**Key words** H.264; multiple coded modes; multiple reference frames; mode decision

**摘要** H.264 是最新的国际视频编码标准,它能够提供更高效的编码效率,但其编码实现也非常复杂,三叉树结构的多编码模式和多帧参考技术是复杂度增加的两个重要因素.针对这两个因素提出一种用于帧间编码的快速模式判决算法 FIMDA,它利用编码过程中获得的模式、码率开销等编码信息来减少那些对编码性能影响不大的参考帧和模式的搜索过程,从而降低编码复杂度.实验结果表明,对常用的 5 帧参考,与全搜索相比, FIMDA 可将编码复杂度降低平均 85% 以上,而 PSNR 下降仅在 0.07dB 左右.

**关键词** H.264; 多编码模式; 多帧参考; 模式判决

中图法分类号 TP391

最新的视频编码标准 H.264<sup>[1]</sup>综合使用了多种编码新技术:可变形块、1/4 像素精度的运动矢量、4×4 整数变换、自适应的环路滤波、内容自适应的变长编码/算术编码等编码技术,有效提高了编码性能,与 H.263、Mpeg2 标准相比, H.264 可以在图像质量相同的前提下节省 50% 以上的码率<sup>[2]</sup>. 不过这些新技术同时也导致了编码复杂度的迅速增加,这制约 H.264 在实时性要求较高场合中的应用,比

如视频会议、视频电话等等.

文献[3-4]研究单参考帧的模式预判决算法,通过分析已编码宏块的信息减少搜索的模式,并引入经验阈值建立中途停止搜索机制(half-stop),以降低单参考帧的模式搜索的复杂度.在我们的先前工作中,通过分析当前宏块纹理特征和周围宏块运动特征来预选编码模式<sup>[5]</sup>,它们对于降低单参考帧的模式判决复杂度都是富有成效的.

针对多帧参考,文献[6]提出了一种通过预选参考帧降低运动估计复杂度的方法,它通过搜索极小范围内的整像素点确定最优参考帧,之后在该参考帧中进行更详细的搜索.该算法的缺点是所有子块必然使用相同的参考帧,不能充分利用 H.264 标准所允许每个  $8 \times 8$  块可使用不同参考帧的特性;此外,预搜索范围的选定很难适合所有序列,可能会造成最优参考帧误选,因而,该算法会导致编码性能较大幅度的下降.文献[7]中提出的算法由近而远地搜索参考帧,搜索过程中利用已有的编码信息,通过设定不同阈值,判定是否继续搜索,该算法充分考虑了视频的不同特性,有较好的编码性能.然而部分阈值需要手工调整,这不利于算法的实际应用.我们对多帧参考与编码复杂度的关系进行了研究,并设计了一种基于多帧参考的复杂度可分级的编码算法<sup>[8]</sup>.

本文分析了 H.264 帧间编码时间复杂度,重点分析了多帧参考和多模式编码对编码复杂度的影响以及与编码性能的关系;在此基础上总结出编码模式随参考帧选择的变化规律;进而设计了快速模式判决的结构和算法 FIMDA(fast inter-frame mode decision algorithm),可以加速单帧参考和多帧参考编码中的模式判决过程.实验结果证明,FIMDA 可以有效降低编码复杂度,且不会引起视频质量的明显下降.

## 1 帧间编码的复杂度分析

帧间编码用于消除视频信号的时域冗余,是视频编码的主要部分.它主要包括模式判决、变换编码和熵编码,其中模式判决是复杂度最高的部分.H.264 帧间编码模式判决过程包括运动估计和帧内预测,目标是寻找与当前编码块(块形状可变)差别最小的预测块. $SAD$ (sum of absolute difference)值是度量这种差别最常用的手段,其定义如下:

$$SAD = \sum_{j=1}^{B_x} \sum_{i=1}^{B_y} |C(j, i) - R(j, i)|,$$

其中, $C(j, i)$ 表示当前块的像素值; $R(j, i)$ 表示预测块的像素值; $B_x, B_y$ 表示块的大小; $|\cdot|$ 表示取绝对值操作.因此也常用  $SAD$  的计算次数来度量模式判决算法的复杂度.假定其搜索范围为  $R$ ,则全搜索需要检测的整像素点数为  $(2 \times R + 1)^2$ ,如果只检测  $16 \times 16$  模式,每个宏块包含 16 个  $4 \times 4$  块,以  $4 \times 4$  块的  $SAD$  为单位,则每个搜索点需要 16 次  $SAD$ ,共计需要进行  $16 \times (2 \times R + 1)^2$  次  $SAD$  计算.

以下我们将借助典型测试序列来具体分析多参考帧和多编码模式两大因素与编码复杂度和编码性能的关系,以寻求降低帧间编码复杂度的方法.值得说明的是参与测试的序列均是 JVT<sup>[1]</sup>提供的标准测试序列,包括运动复杂和运动简单的典型情况,其中,Foreman 和 Carphone 中人物运动和摄像头运动速度较慢;News 和 Salesman 仅有前景人物的缓慢运动;Stefan 中摄像头和人物的运动均较剧烈.这些序列具有较好的代表性.

### 1.1 多帧参考

与单帧参考相比,多帧参考技术能够更有效地提高 3 种情况的预测精确度:一种是块中含有对象的遮挡或出现,块中对象在最近参考帧中找不到匹配的参考块,而在更远的参考帧中可以找到;一种是块中对象的周期性运动,对象在更远的参考帧中可以找到与当前更相似的运动状态,比如飞鸟翅膀的往复性振动;还有一种是对象的分像素级运动,运动对象在紧邻参考帧中处于分像素位置,而更远的参考帧中有整像素位置的参考块,在这类情况下,多帧参考不但可以提高预测精度,还可能提高编码性能,因为编码分像素精度的运动矢量需要更多的比特.然而,多帧参考能在多大程度上提高编码效率,以及使用多少个参考帧最合适,完全依赖于视频本身的特性,如果序列中这 3 类情况的比重很低甚至没有,那么多帧参考所带来的编码增益非常有限,而编码复杂度却总是线性增加的:如果使用  $N$  帧参考,全搜索的运动估计需要计算  $SAD$  的次数增加为  $N \times 16 \times (2 \times R + 1)^2$  次.表 1 中列出了使用 5 帧参考时实际参考帧的分布情况.其中 ref1 为最近的参考帧,ref2,ref3,ref4,ref5 离开当前帧的距离依次增加.结果显示,平均 90% 的参考帧来自最近的参考帧.

Table 1 Distribution of Reference Frame

Sequence	表 1 参考帧的分布					%
	Ref1	Ref2	Ref3	Ref4	Ref5	
Foreman	85.3	6.3	3.7	2.6	2.1	
Carphone	83.9	4.7	5.2	3.7	2.6	
Stefan	82.2	8.3	6.3	1.4	1.8	
Salesman	99.5	0.2	0.1	0.1	0.1	
News	98.9	0.4	0.3	0.2	0.1	
Average	90.0	4.0	3.1	1.6	1.3	

此外,实验中我们还发现,多帧参考技术的编码增益主要来自宏块内子块使用不同的参考帧,就是说,多帧参考在子块使用不同参考帧时更有效,如果

所有子块使用相同的参考帧,该参考帧往往就是最近参考帧,此时多帧参考技术不会带来任何编码增益.这也是预选参考帧算法<sup>[5]</sup>导致编码性能降低的原因之一.

## 1.2 多模式编码

H.264 的帧间宏块可以划分为不同块形状,它采用如图 1 所示的三叉树结构进行块形状划分.任何一个帧间宏块的形状可以是  $16 \times 16$ ,  $16 \times 8$ ,  $8 \times 16$ ,  $8 \times 8$  当中的任意一种 - 对应的前向预测模式分别为  $P16 \times 16$ ,  $P16 \times 8$ ,  $P8 \times 16$  和  $P8 \times 8$ , 而且每个  $8 \times 8$  的子块可以进一步化分为  $8 \times 8$ ,  $8 \times 4$ ,  $4 \times 8$  和  $4 \times 4$ . 因此,一个帧间宏块可以有  $4^4 + 3 = 259$  种块形状划分模式,其中  $4^4$  表示的是  $8 \times 8$  以下形状的模式组合,3 是指  $16 \times 16$ ,  $16 \times 8$  和  $8 \times 16$  等 3 种划分模式.如果宏块内包含多个运动不同的对象或纹理比较丰富,这种划分就可能提高宏块的预测精度.多编码模式的设计导致了 SAD 计算次数迅速增加,仍以  $4 \times 4$  块的 SAD 为单位,  $P16 \times 16$ ,  $P16 \times 8$  和  $P8 \times 16$  分别需要 16 次,  $P8 \times 8$  中,每个  $8 \times 8$  块有 4 种模式,需要 16 次.除此以外, H.264 的帧间编码宏块可以使用帧内模式,因此帧间宏块的编码模式还包括  $I4 \times 4$  和  $I16 \times 16$ ,  $I4 \times 4$  有 9 个预测方向,  $I16 \times 16$  有 4 个预测方向,每个方向的预测需要 16 次 SAD 计算.因此每个宏块需要进行  $(3 \times 16 + 4 \times 4 \times 4) \times N \times (2 \times R + 1)^2 + 16 \times 3 = 112 \times N \times (2 \times R + 1)^2 + 208$  次 SAD 计算,其中  $N$  为参考帧数,  $R$  为搜索范围.

宏块编码模式的选择主要由宏块内对象的运动和纹理情况决定,如果宏块运动情况一致,则选择  $16 \times 16$  模式;如果左、右分属不同的运动情况,则选择  $8 \times 16$  模式;如果上、下分属不同的运动情况,则选择  $16 \times 8$  模式;更复杂的运动情况选择  $8 \times 8$  模式,每个  $8 \times 8$  子块再根据子块不同的运动情况进行类似的划分.图 2 显示了 4 个序列中宏块模式的分布状况,结果表明,模式判决与序列的运动特征和图像的纹理特性存在密切联系:对于基本静止的序列,选择大块的模式占绝大多数,如图 2(d)所示;在运动剧烈的序列中,小块的分布概率较高,如图 2(a), (b)所示;纹理丰富或运动的宏块也偏向于选择较小块的编码模式,这在 4 个序列中均有体现.

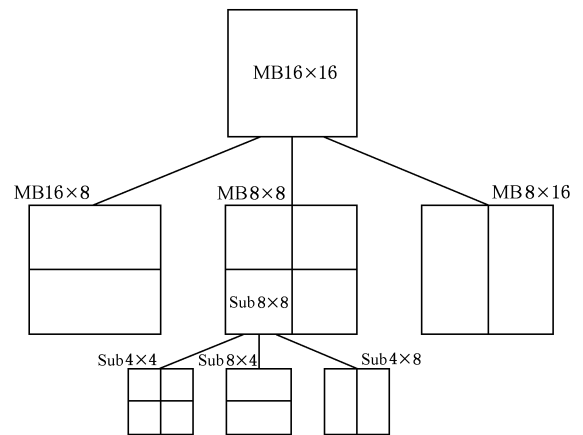


Fig. 1 Tri-tree structured macroblock partitioning in H.264.

图 1 H.264 中三叉树状结构的宏块划分

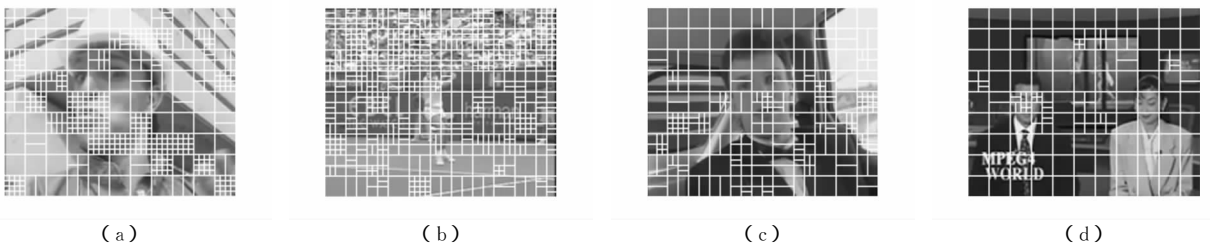


Fig. 2 Relation between coded modes and macroblock's texture and motion. (a) Foreman; (b) Stefan; (c) Carphone; and (d) News.

图 2 编码模式的选择与纹理和运动特征的关系. (a) Foreman; (b) Stefan; (c) Carphone; and (d) News.

我们对编码模式的分布与参考帧的变化关系也进行了统计和分析.表 2 列出各种模式的分布情况,其中每一种模式中左列表示单帧参考时选中该模式的概率,右列表示使用 5 帧参考之后模式发生转变的概率.

可以看出,对于运动简单的序列,如 News, Salesman, 选中  $16 \times 16$  大块的概率很高,而且在使

用 5 帧参考之后发生转变的概率也很低;对于运动较复杂的序列,选中大块的概率明显降低,运动最剧烈的 Stefan,选中  $16 \times 16$  大块的概率已经降到了 36%,其模式的转换概率也很高,达到了 22%.该测试结果揭示了多帧参考对模式判决的影响,同时佐证了上面对宏块模式分布与序列运动和纹理特性的关系分析的结论.

Table 2 Distribution of Coded Modes

表 2 编码模式分布

Mode	P16×16	P16×8	P8×16	P8×8	I4×4	I16×16	%					
Foreman	45	14	11	29	13	22	29	14	1	35	1	4
Carphone	53	11	9	28	11	27	22	17	2	39	3	12
Stefan	36	22	10	24	6	35	43	9	4	45	2	16
News	76	2	3	14	5	15	15	5	1	25	0	4
Salesman	82	1	2	8	2	16	14	3	0	21	0	0
Average	58	10	7	20	7	23	25	10	1	33	1	7

在统计实验中,我们还注意到,宏块帧内模式的提前检测有利于后续参考帧中模式判决,因为块的大小说明了块的纹理特性,大块( $I16 \times 16$ )表明纹理简单,小块( $I4 \times 4$ )表明纹理丰富.因此我们首先检测帧内模式,在后续的模式判决中,如果最优模式是帧间模式,首先确定最优模式,再根据帧内模式选择次优的候选模式:如果是  $I4 \times 4$  则偏向于选择形状较小的模式,否则选择形状较大的模式.

## 2 快速模式判决算法设计

根据以上统计数据和分析结果,可以总结出编码模式与参考帧之间的变化规律.为方便描述,将全局最优编码模式记为  $MODE_{opt}$ ,第 1 参考帧中最佳编码模式记为  $MODE_{first}$ .

① 如果  $MODE_{first}$  为  $P16 \times 16$ ,则  $MODE_{first}$  往往就是  $MODE_{opt}$ ,最优参考帧就是第 1 帧,可以停止搜索;

② 如果  $MODE_{first}$  为帧内模式,则  $MODE_{opt}$  就取  $MODE_{first}$ ,停止搜索其他参考帧;

③ 在帧内模式中,如果  $I4 \times 4$  较好,后续的模式检测中,偏向于选择小块的帧间模式;否则偏向于选择大块的帧间模式.

根据以上结论,我们设计了如图 3 所示的快速模式选择编码结构,具体步骤和算法描述如下:

步骤 1. 检测当前编码的两种帧内模式: $I4 \times 4$  和  $I16 \times 16$  选择较好的帧内模式,记为  $BetterIntraMode$ .

步骤 2. 用快速算法搜索最近参考图像,寻找最优帧间模式,并与帧内模式比较,最优模式记为  $FirstBestMode$ ,该模式下的 RD 开销记为  $BestMode\_RDCost$ ,同时置  $Current\_RDCost$  为  $BestMode\_RDCost$ .快速模式检测算法是采用文献[5]中对于宏块的纹理特征和局部运动特性特征进行的模式选择,包括计算像素的变化特征、水平纹理特征、垂直纹理特征和运动矢量相关性分析,根据这些特征信息可以避

免检测那些选中的概率很低的模式.

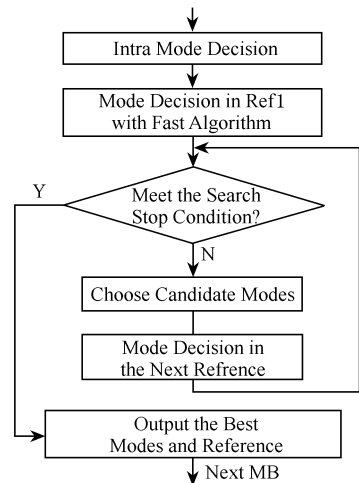


Fig. 3 Process of fast mode decision.

图 3 快速模式判决算法流程

步骤 3. 判断是否满足搜索停止条件,如果满足,转到步骤 5;不满足则用  $Current\_RDCost$  更新  $BestMode\_RDCost$ .搜索停止条件是满足以下条件中的任意一个:①搜索完第 1 参考帧  $ref1$  后,如果( $FirstBestMode = BetterIntraMode$ )或者( $FirstBestMode == P16 \times 16$ );②( $Current\_RDCost \geq BestMode\_RDCost$ );③已经搜索完所有参考帧;

步骤 4. 根据最优模式  $FirstBestMode$  和较好的帧内模式  $BetterIntraMode$  选择一种模式或几种模式作为下一个参考帧的候选模式.具体过程如下:

如果  $FirstBestMode$  是  $P16 \times 8$ ,则  $P16 \times 8$  作为候选模式;此时如果  $BetterIntraMode$  是  $I4 \times 4$ , $P8 \times 8$  也作为候选模式;

如果  $FirstBestMode$  是  $P8 \times 16$ ,则  $P8 \times 16$  作为候选模式;此时如果  $BetterIntraMode$  是  $I4 \times 4$ , $P8 \times 8$  也作为候选模式;

如果  $FirstBestMode$  是  $P8 \times 8$ ,则  $P8 \times 8$  作为候选模式;此时如果  $BetterIntraMode$  是  $I16 \times 16$ , $P16 \times 16$  也作为候选模式;

步骤 5. 在下一个参考图像中检测候选模式得到最优模式,并更新  $FirstBestMode$ ,记录该模式下的 RD 开销  $Current\_RDCost$ ,返回步骤 3;

步骤 6. 输出最优模式,如果是帧间模式,同时输出最优参考帧.

## 3 实验结果

所有实验在 PC 平台上执行,CPU 运行频率为 2.4GHz,内存为 768MB,实验平台为 JM7.3<sup>[9]</sup>,

相关编码参数设置如表 3 所示. 图 4 是各序列使用快速模式判决算法 FIMDA 与全搜索(full search, FS)的性能对比, 图中曲线为它们的编码性能曲线. 结果显示: 使用快速模式判决算法后编码性能没有明显下降, 对于运动平滑、简单的序列, 编码性能与全搜索基本一致, 比如 Salesman 和 News 中, 两条曲线基本重合; 甚至对运动剧烈的 Stefan 序列也显示了很好的编码性能, 造成该现象的原因在于运动特别剧烈之后, 帧内模式预测性能优于帧间模式, 而 FIMDA 是优先考虑帧内模式的, 对其他两个运动相对复杂的序列编码性能则稍有下降.

Table 3 Parameter Settings of Encoding

表 3 编码参数设置

Parameters	Setting
Image Format	QCIF (176 × 144)
QP	28, 32, 36, 40
Search Range	16
Reference Number	5
Entropy Mode	CABAC
RDO	Enable

表 4 是按照文献 [10] 中描述的方法(将码率和失真折算为统一的量来衡量编码性能) FIMDA 与 FS 的编码性能的比较结果.  $\Delta PSNR$  表示统一用 PSNR 衡量性能时 FIMDA 比 FS 增加的值, 负号表示降低;  $\Delta bit-rate$  表示统一用码率衡量编码性能时 FIMDA 比 FS 增加的码率.

Table 4 Encoding Performance Comparison of FIMDA and FS

表 4 FIMDA 与 FS 的编码性能比较

Sequence	Foreman	Carphone	Stefan	News	Salesman
$\Delta PSNR$ (dB)	-0.13	-0.15	-0.06	-0.01	-0.005
$\Delta bit-rate$ (%)	2.95	3.01	1.19	0.18	0.13

Table 5 Complexity Comparison of FIMDA and FS (kilo-cycles/MB)

表 5 FIMDA 与 FS 的编码复杂度比较(千周期/宏块)

Sequence	Foreman	Carphone	Stefan	News	Salesman	Average
FS	6164.3	6181.1	6244.8	6145.8	6272.4	6201.7
FIMDA	628.8	1193.0	1099.1	540.8	420.2	776.4
Complexity Reduction (%)	89.8	80.7	82.4	91.2	93.3	87.5

编码每个宏块平均所需的 CPU 周期数作为衡量复杂度的尺度. 结果表明, 快速模式判决算法对降低编码复杂度的效果是明显的, 平均降低在 85% 以上.

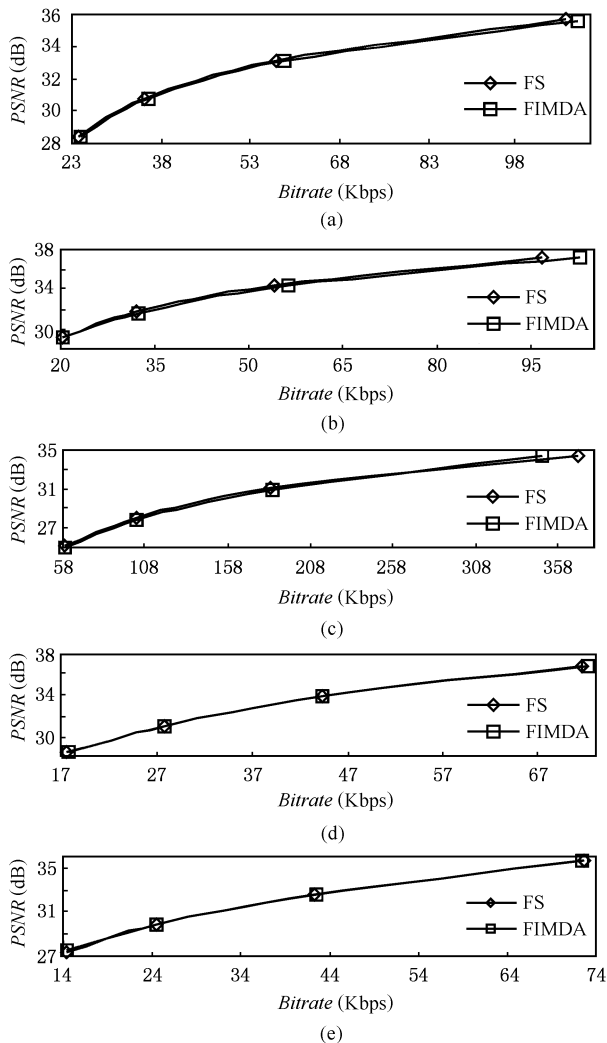


Fig. 4 Encoding performance curves of FIMDA and FS. (a) Foreman 30fps; (b) Carphone 30fps; (c) Stefan 30fps; (d) News 30fps; and (e) Salesman 30fps.

图 4 FIMDA 与 FS 的编码性能曲线对比. (a) Foreman 30fps; (b) Carphone 30fps; (c) Stefan 30fps; (d) News 30fps; and (e) Salesman 30fps.

表 5 是 FS 和 FIMDA 的编码复杂度的比较结果. 为了避免图像大小引起的复杂度变化, 我们以

## 4 结 论

本文针对影响 H.264 帧间编码复杂度的两个

**重要因素:**多参考帧和多编码模式,研究快速模式判决算法。借助典型测试序列编码的统计结果,分析了多参考帧和多编码模式对编码复杂度和编码性能的影响,据此设计了快速模式判决算法 FIMDA。实验结果表明,本文提出的快速算法可以有效降低使用多帧参考的 H.264 的编码复杂度,同时基本不会降低编码的视频质量,复杂度平均降低在 85% 以上,因此而引起的 PSNR 下降在 0.07dB 左右,折算为码率增加仅为 1.5% 左右。算法性能明显优于参考帧预选算法<sup>[6]</sup>,也优于我们先前的工作<sup>[5,8]</sup>。与文献<sup>[7]</sup>中的算法编码性能相当,复杂度更低,而且避免了手动调整阈值。FIMDA 能够显著降低编码复杂度,并保持较高的编码性能,其根本原因在于它充分利用了视频序列的参考帧分布特性、块模式的相关性,借助编码过程中获得的信息以及对序列乃至宏块特征的分析来指导整个编码过程,避免搜索那些对编码性能影响不大的参考帧和编码模式。FIMDA 的实用性强,适用于运算能力低的平台或实时性要求较高的的视频编码。

### 参 考 文 献

- [1] Draft ITU-T Recommendation and Final Draft Int'l Standard of Joint Video Specification (ITU-T ec. H.264/ISO/IEC 14496-10 AVC) [S]. ISO/IEC MPEG and ITU-T, JVTG050, 2003
- [2] T Wiegand, J Gary, G Bjontegaard, *et al.* Overview of the H.264/AVC Coding Standard [J]. IEEE Trans on Circuit System and Video Technology, 2003, 13(7): 560-575
- [3] A Chang, O C Au, Y M Yeung. A novel approach to fast multi-block motion estimation for H.264 video coding [C]. 2003 Int'l Conf on Multimedia and Expo, Baltimore, Maryland, 2003
- [4] Peng Yin, Hye-Yeon Cheong, Tourapis, A M Tourapis, *et al.* Fast mode decision and motion estimation for JVT/H.264 [C]. 2003 Int'l Conf on Image Processing, Barcelona, 2003
- [5] Yanfei Shen, Dongming Zhang, Chao Huang, *et al.* Fast mode selection based on texture analysis and local motion activity in H.264/JVT [C]. IEEE 2004 Int'l Conf on Communications, Circuits and Systems, Chengdu, 2004
- [6] C W Ting, L M Po, C H Cheung. Center-biased frame selection algorithms for fast multi-frame motion estimation in H.264 [C]. The 2003 IEEE 2003 Int'l Conf on Neural Networks and Signal Processing, Nanjing, 2003
- [7] Y W Huang, B Y Hsieh, T C Wang, *et al.* Analysis and reduction of reference frames for motion estimation in MPEG-4 AVC/JVT/H.264 [C]. IEEE 2003 Int'l Conf. Acoustics, Speech, and Signal Processing, Hong Kong, 2003
- [8] Dongming Zhang, Chao Huang, Shouxun Lin, *et al.* Complexity scalable motion estimation based on modes pre-selection in H.264 [C]. IEEE Int'l Conf on Signal Processing, Beijing, 2004

- [9] JVT. JVT Reference Software JM73 [OL]. <http://bs.hhi.de/~suehring/tml/download/JM73.zip>, 2003
- [10] G Bjontegaard. Calculation of Average PSNR Differences Between RD-Curves. JVT Document VCEG-M33, 2001



**Zhang Dongming**, born in 1977. Ph D candidate in multimedia processing at the Institute of Computing Technology, the Chinese Academy of Sciences. His main research interests include video coding, design of video codec system on chip and video surveillance.

张冬明, 1977 年生, 博士研究生, 主要研究方向为数字视频编码。



**Shen Yanfei**, born in 1976. Master and assistant professor. Currently working in the Institute of Computing Technology, the Chinese Academy of Sciences. He focuses on research of digital multimedia signal

processing.

沈燕飞, 1976 年生, 硕士, 助理研究员, 主要研究方向为多媒体数字信号处理。



**Lin Shouxun**, born in 1948. Ph D professor and Ph D supervisor in the Institute of Computing Technology, the Chinese Academy of Sciences. Received his Ph D

degree from the Beijing University of Technology, Beijing, China, in computer science, in 1998, senior member of China Computer Federation. His current research interests include of video coding, video analysis and retrieval, universal media access, measurement and evaluation of human language technology.

林守勋, 1948 年生, 博士, 研究员, 博士生导师, 中国计算机学会高级会员, 主要研究方向为多媒体技术及应用系统。



**Zhang Yongdong**, born in 1973. Ph D and associate professor in the Institute of Computing Technology, the Chinese Academy of Sciences. Received his Ph D

degree from Tianjin University, Tianjin, China, in Electrical Engineering, in 2002, senior member of China Computer Federation. His current research interests include video coding and transcoding, video analysis and retrieval, universal media access.

张勇东, 1973 年生, 博士, 副研究员, 中国计算机学会高级会员, 主要研究方向为视频内容分析、视频数据压缩等。

## Research Background

H.264/AVC is a state-of-the-art video coding standard. It gives better coding performance than previous standards, but the cost is intensive computation. Current experiments show that the computation of H.264 encoding amounts to 3~4 times MPEG-2. Multiple frames reference and multiple coded modes are the main contribution to the high coding complexity. The existing literatures focus on the fast algorithm for multiple frames reference or multiple modes individually. This work builds a general fast mode decision framework to study multiple frames and coded modes together, and the proposed algorithm can reduce coding complexity greatly with negligible encoding performance loss. This work is mainly supported by the National Natural Foundation of China under the grant number 60302028.

## 《计算机研究与发展》简介

《计算机研究与发展》是中国科学院计算技术研究所和中国计算机学会联合主办、科学出版社出版的学术性刊物、中国计算机学会会刊。主要刊登计算机科学技术领域高水平的学术论文、最新科研成果和重大应用成果。读者对象：各行业、各部门从事计算机研究与开发的研究人员、工程技术人员、各大专院校计算机专业及其它相关专业的师生和研究生。

《计算机研究与发展》于1958年创刊，是我国第一个计算机刊物，现已成为我国计算机领域知名度较高的学术期刊之一。并历次被评为我国计算机类核心期刊及国务院学位办指定的评估学位与研究生教育的“中文重要期刊”。此外，还被《中国学术期刊文摘》、《中国电子科技文摘》、《中国科学引文索引》及“中国科学引文数据库”、国家科委“中国科技论文统计源数据库”等国家重点检索机构列为引文刊物；并成为美国工程索引(EI)检索系统、日本《科学技术文献速报》和俄罗斯《文摘杂志》收录的期刊。此外本刊历届被我国权威评估机构评为“百种中国杰出学术期刊”。

为了方便广大作者和读者，从1997年开始我编辑部已实行数据库管理、网络投稿、网络审稿、网络查询等全部自动化管理。为了扩大大刊的影响，从1998年开始，期刊的中英文摘要已全部上网；从2005年开始，实现网上全文检索。

欢迎订阅，欢迎投稿。

来函请寄：100080 北京 2704 信箱《计算机研究与发展》编辑部

国内邮发代号：2-654；国外发行代号：M603

国际标准刊号：ISSN1000-1239

国内统一刊号：CN11-1777/TP

电话（010）62620696；62600350

Email：crad@ict.ac.cn

http://crad.ict.ac.cn