

摘 要

H. 264/AVC 是新一代的视频编码标准, 具有压缩性能好, 图像质量清晰和良好的网络交互性能等多种特性, 因而受到了诸多科研机构及多媒体工业界的高度重视。在 H. 264/AVC 标准中, 码率控制技术是其中的关键部分。码率控制技术的主要工作是在编码前对帧级、宏块级的量化参数进行计算、分配比特数, 这些工作对图像质量具有重要影响。本文主要研究码率控制在低带宽下的应用, 包括以下两个方面:

一、在带宽较窄的情况下, 视频整体质量并不能完全的满足人们的需求。在这种情况下, 人们开始希望通过改善感兴趣区域的视频质量以其达到主观要求。在本文中提出了一种新的关于人脸感兴趣区域的码率控制算法, 该算法建立在一种有效的 ROI 提取方法上, 改进了一种帧级配码率分配策略和基本单元层的 MAD 预测方法, 调整了不同宏块的 QP。为了更好的强化感兴趣区域的质量, 算法对感兴趣区域的比特数进行了增强。实验结果表明, 这种改进算法提升了感兴趣区域的 PSNR 和感兴趣区域的主观质量。

二、本文分析了 H.264 中 MAD 线性模型预测中的不足, 对此提出了改进策略。该算法不仅利用了时间的信息也利用了空间的信息进行预测 MAD 的计算。在分别计算得出利用空间信息和时间信息的预测 MAD 值后根据两者的一贯以来的精确程度进行比较、确定。在此基础上, 本文在原来的 JVT-G012 采用方法上提出了新的帧级比特分配方案, 这种新的分配方法结合了缓冲区的充溢度, 能够根据编码的复杂度自适应的分配了比特。实验结果表明, 这种改进算法可以较好的提高 PSNR 值。

关键词: H.264; 码率控制; 感兴趣区域; MAD 预测; 码率分配

Abstract

H.264/AVC video coding standard which is the newest video coding standard is paid close attention with its higher compression efficiency, better image quantity and network compatibility. Rate Control as a very important part of the video compressing encode, can compute QP and has responsibility to allocate bit at the frame-level and macroblock-level. The paper mainly researches on Rate control algorithm of H.264/AVC at the low bandwidth include the following two areas:

First, the video quality can not meet the need at the low Bandwidth, so researchers make some methods to increase the quality of ROI and hope to meet people's need using this method. A rate control algorithm based Region of interest is proposed in this paper, the algorithm which establishes an effective method in the ROI extraction, works at frame level and the basic unit level. The algorithm improves the tactics to allocate the bit at the frame level and improves the MAD prediction at the basic-unit level. The algorithm also enhances the bit in the region of interest to raise the quality. The results shows the algorithm increase PSNR and improves the subjective quality.

A rate control algorithm is proposed based on analyzing the lack of MAD prediction linear model. The algorithm predicts MAD using the information of time and space, and this way is more precise than the original way which only use the information of space. The paper improves the tactics of bit-allocation at frame-level based on the new MAD prediction model. The new way of bit-allocation can adaptively allocate the bit combining the buffer fullness and the coding complexity. The result shows the algorithm can increase PSNR.

Key words: H.264; Rate Control; Region of Interest; MAD prediction; Bit allocation

西南交通大学

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权西南交通大学可以将本论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复印手段保存和汇编本学位论文。

本学位论文属于

1. 保密 ，在 年解密后适用本授权书；
2. 不保密 ，使用本授权书。

(请在以上方框内打“v”)

学位论文作者签名: 朱柳

日期: 2010.6.5

指导老师签名:



日期:

2010.6.7

西南交通大学硕士学位论文主要工作（贡献）声明

本人在学位论文中所做的主要工作或贡献如下：

1. 改进了针对人脸感兴趣区域的码率控制算法。该算法建立在有效的 ROI 提取方法上，该算法在帧级采用了新的比特分配策略，改进基本单元层中 MAD 预测算法，对不同宏块的 QP 值进行了调整同时对感兴趣区域的比特数进行增强。
2. 分析了 H.264 中帧级 MAD 线性模型中的不足，并提出了改进策略。这种改进策略在预测 MAD 时，改变了原来 MAD 只利用时域信息进行计算的方法，在利用时域信息进行计算的同时也利用了空域的信息进行了计算。本文同时改进了原来 G012 方案中对帧级比特分配策略，这种新的分配方法结合了缓冲区的充溢度，能够根据编码的复杂度自适应的分配了比特。

本人郑重声明：所呈交的学位论文，是在导师指导下独立进行研究工作所得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中作了明确说明。本人完全了解违反上述声明所引起的一切法律责任将由本人承担。

学位论文作者签名：

宋柳

日期：2010.6.5

第 1 章 绪 论

1.1 课题背景及意义

当今世界是一个信息爆炸的世界，通信技术，计算机技术和网络技术不断发展，人类的生活方式和工作方式因此有了深刻的改变。在这个过程中，多媒体技术和视频技术成为了当前的一个研究热点，这些研究热点的不断发展和兴起会给人们生活带来更多的便利。

H.264/AVC^[1-5]视频压缩标准是联合视频工作组 JVT (Joint Video Team) 正式提出的新一代视频压缩标准，它具有抗误码能力强、多编码模式，编码参数自适应选择，上下文自适应熵编码，多参考帧选择等特点。H.264 这一系列的技术措施大大提高了编码和压缩效率，使其可以应用在数字电视，移动视频，可视电话，视频会议等。

码率控制作为视频编码的重要组成部分，对提高视频质量有举足轻重的作用。码率控制技术对不同的应用可以采取不同的策略，具有很强的灵活性，码率控制在有限的缓冲区容量和信道宽带下保证良好视频质量进行传输为最终目的。但是在已经广泛推广的数字电视广播和正处在新兴发展中的数字移动电视，都对码率有相对比较严格的要求^[1,2]，因此根据实际需要 H.264 码率控制部分进行修改也是一项非常有意义的课题。

在实际的生活中，由于人眼视觉特性人们总是对视频中的某些特定区域特别感兴趣而对其他区域部分之外的信息并不是特别重视。因此在带宽较窄的实际应用中，由于很难提高整体视频的质量，有学者开始利用码率控制等技术提高感兴趣区域 (ROI) 的视频质量来满足人们在低比特条件下对视频的主观要求。

1.2 国内外研究现状

在现有的研究中，各国学者都在根据各种应用和需求平衡视频编码中的两个关键问题—效率和失真。作为编码器中的重要一环码率控制是平衡这两个关键问题的一种解决方案。码率控制可以对天生具有波动的视频进行控制，结合现有的缓冲器容量估计网络的可用带宽后决定要分配给每帧多少比特数，也可以根据不同的应用调整编码参数、量化参数使视频质量达到最好的状态。现在的码率控制方案可按照应用信道的不同分为：恒定模式 (CBR) 和可变模式 (VBR)；或按照参考视频信息角度，分为前向控制、后向控制、联合前向控制和后向控制模式。一个典型的码率控制方案是由缓冲区策略、目标码率分配、率失真模型、量化参数选择等组成的。

在码率控制算法研究中，主要的算法有 2 个：

1. 在 JVT-F086 中提出，应用于 MPEG-2 中 TM8 版本改进的算法^[6]：主要分为帧比特数分配，第一次率失真计算，量化参数选择，再次率失真计算四部。

2. 另一种算法是在 JVT-G012^[7]提案中介绍的算法。它提出了根据流量往返模型分配比特数, 并且根据采用二次率失真模型函数计算宏块层的量化参数等创新点。

这两种算法采用了不同的算法解决了“蛋鸡悖论”, 其中 JVT-G012 提案中所提出的线性预测 MAD 的方法拟合度比较高, 效果比较好。现在最新的 H.264/AVC 采用的是这种算法。但是这两种算法还是具有不足之处: 例如计算复杂度高, JVT-F086 提案所提出的算法使用了二次模型计算的方法; 在 JVT-G012 提案每次码率控制中的 QP 计算也都需要修改二次模型, 这个计算的复杂度是相当可观的; 另外在 JVT-G012 提案中提出流量往返模型, 线性预测模型和二次率失真模型, 这些模型都需要不断的随着编码的过程对参数进行修改, 算法复杂度比较大。计算复杂度高将降低编码效率同时增加硬件负担。还有上述提案对于场景切换或者是运动平缓 and 运动剧烈场景交替出现时, 码流的平稳性控制不够好。在运动平缓和运动剧烈场景交替出现时, 编码器的码流将会出现剧烈的波动, 容易在接收端产生视频抖动, 对于视频质量有所影响。

现在对于码率控制算法也有其他的学者针对各种应用提出了相对应的改进。针对宏块级别比特分配不准确等缺陷, Milanis^[8]提出了一种低复杂度的码率控制算法, 这种算法发现量化后零系数所占比率和编码码率之间的近似线性关系, 可以较精确对宏块所需比特进行预测, 从而进行分配; 文献[9]提出平均绝对变换量化误差 (MATQD), 对宏块级码率控制算法采用了新型的加权预测算法, 产生了新的二次 R-D 模型; 李蕾^[10]提出了一种实时的比特分配和码率控制算法, 这种算法不需要二次编码且根据两帧的直方图信息作为帧编码复杂度的测量准则, 算法复杂度有所下降^[11]。黄莉^[12]则提出利用 MADratio 调整目标比特数和量化参数的新型码率控制算法。

同时由于量化参数(QP)的计算对实现精确的码率控制起着决定性作用, 也有学者致力于此方向的研究, 在以前的研究重点在于开发各种不同的 R-Q 模型(比特-量化等级), 有基于幂指数的、基于对数的、曲线的、多项式的(包括线形和二次)、指数的等等。

总而言之在码率技术的发展过程中, 对于码率控制算法的要求在不断地提高, 并据此可分为三个发展阶段。第一个阶段的目的在于精确地实现目标码率, 学者研究发现可以将确定量化参数的率失真模型作为重要因素应用于各种视频编码系统中。通过实践验证这种方法可以成功的精确实现目标码率。第二个阶段在精确控制码率的同时, 要求提高视频质量。此时衡量视频质量的标准是传统的 PSNR 等客观质量评价标准。第三个阶段以主观视觉质量为衡量标准, 在码率控制技术中充分考虑人眼视觉特性的因素, 力求在同等的码率下获得更加满足人的主观感受的视频画面或者用更少的存储空间或网络带宽得到同等主观质量的视频画面。从长远来看, 结合人眼视觉特性的视频编码技术是非常有发展前景的。

就感兴趣区域而言, 有一个重要的研究热点是如何提取感兴趣区域。现阶段的提取技术主要分为: 帧内的空域分割技术, 帧间的时域分割技术和时/空域联合分割技术。也有些算法^[13-15]利用视频序列的时间相关性, 利用当前图像和前, 后图像背景相同,

色度亮度信息基本相同只是物体的位置不同的特点,确定宏块的 QP 值。而人脸是一个非常重要的感兴趣区域。针对人脸这个 ROI 区域的提取方面,国内外的研究机构如:MIT,清华大学,中科院计算机技术研究所、中科院自动化研究所等,都有人员从事这项工作并且取得了一定的成果。

1.3 论文的研究内容和章节安排

1.3.1 论文的研究内容

本论文的主要研究内容有:

1、在可视电话,视频会议等低带宽视频应用中人脸区域的重要性,本文分析了视频中人脸检测技术,同时针对原有 JVT-G012 算法中的不足,改进了一种在低码率应用中的针对人脸的 H.264/AVC 码率控制算法。这种算法提出了一种新的帧级配码率分配策略,改进基本单元层的 MAD 预测,调整了不同宏块的 QP。为了更好的强化感兴趣区域的质量,算法对感兴趣区域的比特数进行了增强。

2、文分析了 H.264 中 MAD 线性模型预测中的不足,对此提出了改进策略。该算法不仅利用了时间的信息也利用了空间的信息进行预测 MAD 的计算。在分别计算得出利用空间信息和时间信息的预测 MAD 值后根据两者的一贯以来的精确程度进行比较、确定。在此基础上,本文在原来的 JVT-G012 采用方法上提出了新的帧级比特分配方案,这种新的分配方法结合了缓冲区的充溢度,能够根据编码的复杂度自适应的分配了比特。

1.3.2 论文的章节安排

本论文分五章展开。

第一章,绪论,首先介绍了本论文的研究背景及意义,接着介绍了相关技术的国内外研究现状,最后介绍了论文的主要工作和论文的章节安排。

第二章,介绍了视频编码码率控制原理、JVT-G012 及其他经典算法。

第三章,分析了感兴趣区域编码的意义,详细介绍了针对人脸感兴趣区域的码率控制算法,其中包括感兴趣区域的提取以及在提取后针对感兴趣区域的码率控制算法。在实现了基于人脸的 H.264 码率控制算法后给出了仿真结果。

第四章,分析了 H.264/AVC 的码率控制算法的不足。在这个基础上,改进了文献 [16]的帧层比特分配策略和帧层的 MAD 预测方法,并给出了仿真结果。

最后,给出了本文研究工作的总结,并分析了其中的不足,对下一步研究内容和方向进行了展望。

第 2 章 H. 264 码率控制基本原理和经典算法

2.1 码率控制技术基本原理及码率控制算法分类

2.1.1 码率控制技术基本原理

在目前的视频编码标准中由于所采用 I、P、B 帧的压缩原理不同，每种类型的帧消耗的比特是显著不同的，因此码率具有了天生的波动特性。在现实的应用中码流的强烈波动会导致系统无法正常工作，对视频通信系统产生非常严重的后果，因此必须进行必要的控制。码率控制使码流能够在网络上进行平稳的传输，满足实时信源解码的播放效果。码率控制的目标是在实际带宽的限制下，调整编码参数，对每个视频序列的编码单位中如图像组，帧，宏块间进行比较合理的比特分配，达到比较好的图像质量，满足各类应用需求。

控制系统结构中是采用反馈校正装置使得原有的系统缺陷得到校正。如图 2-1 所示在码率控制系统结构中采用的也是反馈校正装置使得整个性能指标满足给定的各项性能指标。从图示可以看出码率控制中的性能指标有码率，缓冲延迟，缓冲区状态等。其中在编码器和传输信道间的缓冲器，主要起到了平滑由码流使其平稳的进入传输信道。而码率控制器的作用是使得原来波动很大的原始码流以平稳的方式进入缓冲区。若选择的是恒定码率模式，则输出的码率将维持不变；若是可变模式，那么输出的码流将在一定的范围内波动。毫无疑问，当缓冲器足够大的时候可以满足恒定的输出码流，但是值得注意的是在实际应用中，缓冲器往往是有限的且由于缓冲器带来延迟会影响实时视频通信。在现在的码率控制中，也有不少学者研究的码率控制算法中^[17-20]考虑到了缓冲器的特性，使得在充分利用了缓冲器能平滑码率的有效特性同时也不会出现溢出的情况。

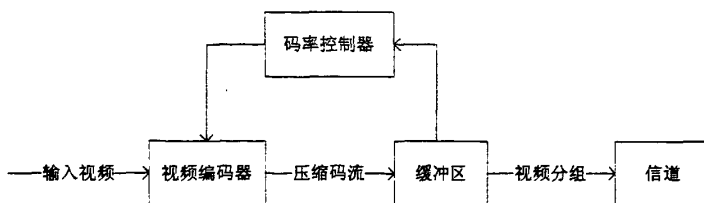


图 2-1 传输信道

2.1.2 码率控制算法分类

码率控制从用户网络接口选择的信道编码是否恒定分为以下两种模式：可变码率 VBR 模式和恒定码率 CBR 模式。两者的主要不同是：恒定码率模式实现方法比较简单，通过输入视频源和目标比特结合缓冲区大小、网络带宽的估计动态调整 QP，得到符合要求的码率。在 CBR 模式中，当图像复杂度比较低通过填充无用的比特达到设定码率，

当图像纹理信息比较多时,调整 QP 降低图像质量满足设定的码率需要。可变码率模式设定 QP 值,由于图像信息不同,码流也相应地产生变化[21]。可变码率模式可以根据不同的视频场景提供出好的视频质量。CBR 模式主要应用于网络视频会议,视频监控会议,数字视频点播,数字电视等可以认为传输信道恒定的应用中。VBR 模式主要应用在移动视频电话,移动视频电视,无线网络等由于传输信道不稳定,码率经常处在波动的情境中。

另外码率控制算法也可以从是否参考视频信息角度上大体上分为三类^[22]:第一是前向控制。前向控制先对输出码率进行预测或利用基于失真率模型的方法来估计各个编码单位的输出码率进而确定 QP。这种方法计算量大且需要考虑图像的纹理信息和人眼的视觉特性,但是这种方法较之后向预测可以更好的提高视频质量。第二类是依据信道速率和缓冲区容量来调整量化步长的后向控制。后向预测能有效地保证编码器输出码率与信道传输率相适应,但在低码率时图像质量波动较大。第三类是联合前向控制和后向控制方法,在图像层采用自适应后向控制,确定量化步长的基准值;在宏块层采用前向控制,细化修正量化步长的基准值,确定每个宏块的量化步长。

在码率控制算法中也可以采用调整相关编码参数的方法对进视频编码器进行有效的比特调整,达到最终的码率控制要求^[21-23]。

1 调整视频序列的每秒编码帧数,即帧率。当码率高于信道速率的时候,通过丢掉几帧来降低码率;当码率低于信道速率的时候,通过提高帧率来提高视觉的连续性。帧率控制方法只能针对视频信号时域冗余,而且需要注意这种方法通常是用在对单个图像质量要求高,不能通过减小每帧的编码比特数来达到降低码率的情况。

2 只对每个图像块的部分像素编码,如对角线系数编码或者只对图像的低频系数编码。由于整个像素块中 DC 系数的能量占有较大比例,只对像素块中的 DC 系数进行编码来保持图像质量,而对图像质量影响不大的 AC 系数少编码或不编码,这样可以大大减少输出比特率。这种方法主要是应用在图像细节较少,空间相关性较大时。这种方法可以在低码率下获得较好的图像质量,但是在图像纹理信息较多或者对图像质量要求很大的情况下,丢到太多的 AC 系数的方法并不适用。

3 调整图像大小。改变分辨率也可以调整码率。当码率高于信道速率时,调高图像的分辨率来降低码率,当码率低于信道速率,缩小图像尺寸满足信道传输。

4 调整量化参数 QP。当视频序列中有剧烈运动可以通过增加 QP 值获得较高的时域视频质量。在图像的空间细节采用较大的 QP 值量化,可以降低空间视频质量最终实现码率的平稳。QP 值越大 DCT 系数经过量化后的取值范围越小,在游程编码之前可获得的零系数编码越多,熵编码后压缩程度越高,最终输出的比特数小;相反,QP 值越小,量化后的取值范围越大,编码后的比特数越高。在 H.264 编码器中,可以通过对调整帧、片或者宏块的量化参数来实现不同级别的码率控制。

5 控制输出比特率的编码参数。设置这个参数可以确定预测帧(P 帧)中的宏块是编码还是跳过。如果门限值提高,编码器对运动敏感度降低,编码宏块的数目将减少。

相反,对较低的运动门限,运动敏感性提高,将有更多数目的宏块被编码,从而比特率也提高。同样,INTRA 或 INTER 判决门限的提高或降低也可用来控制 P 帧中编码宏块的输出比特率。更多的 INTRA 编码宏块会使得比特率增加,编码质量提高。

上述几种编码参数的调整可能会引起图像质量的变化,但是在编码过程中调整它们的取值可以有效的控制视频编码器的输出比特率,最终达到当前码率控制的要求。

2.2 H.264/AVC 码率控制算法

H.264/AVC 编码标准中码率控制部分的提案主要有 JVT-F086^[24],JVT-G012^[25],JVT-H017^[26],JVT-O016^[27]。其中比较具有重大突破意义的是 JVT-F086 和 JVT-G012 提案。JVT-F086 是中国科学院计算机研究所的马思伟博士在 2002 年提出的一种码率算法。这种算法在 TM5 的码率控制算法基础之上进行了修改,但是该算法的性能特点并不是十分的突出。在 JVT2003 年度的 Pattaya 会议中,JVT-G012 提案由新加坡的学者 Zhengguo Li 等提出。JVT-G012 提案是目前 H.264 码率控制算法中最具有影响力的一个算法。该算法提出了 MAD 的线性模型和基本单元层的概念较好的解决了蛋鸡悖论等传统难点,奠定了现在很多码率算法的基础。现在很多算法都在 JVT-G012 上进行修改并和它进行比较。

2.2.1 H.264 码率控制的基本概念

1 蛋鸡悖论

在 H.264 编码标准中,引入了率失真优化对模块的编码模式进行选择 and 运动预测同时率失真优化也会影响 QP 的计算。这将导致“蛋鸡悖论”。宏块的编码流程如下:



图 2-2 蛋鸡悖论

通过 2-2 图可以看出,QP 是通过码率控制的计算得到的。而此 QP 值将用来进行率失真优化和模式选择,在确定了编码模式并计算得出此模块的 MAD 值后将对该宏块进行编码。从上述的过程中可以看出,利用率失真优化进行宏块的模式选择必须先确定量化参数。而量化参数的计算是必须需要依靠得到当前宏块的 MAD 值,值得注意的是 MAD 值只能在率失真优化以后才可以确定得到。这样就产生了“蛋鸡”悖论。

2 基本单元层 (Basic Unit)

假设图像的每一帧是由 N_{mbpic} 个宏块组成,那么一个基本单元为一组由 N_{mbunit} 个邻近宏块组成,其中 N_{mbunit} 是 N_{mbpic} 的约数。如果一帧的基本单元的个数是 N_{unit} ,则有:

$$N_{unit} = \frac{N_{mbpic}}{N_{mbunit}} \quad (2-1)$$

基本单元可以是一个宏块、一片 (Slice)、一场 (field) 或一帧。假设有一个 QCIF

大小的视频序列, N_{mbpic} 为 99。依照定义, N_{mbunit} 可以是 1、3、9、11、33 或 99。 N_{unit} 为 99、33、11、9、3 和 1。在码率控制中使用基本单元作为码率控制的最小单元。

需要注意的是, 如果使用一个大的基本单元可以取得高 PSNR 值, 但是比特波动也大; 如果使用一个小的基本单元, 比特波动相应减少, 但是 PSNR 会明显降低。

3 MAD 线性预测模式

线性预测模型是用前一帧相应位置上基本单元的 MAD 值预测当前编码帧中基本单元的 MAD 值。用 MAD_{cb} 表示当前待编码帧中基本单元的 MAD 预测值, 而前一帧在相关位置上基本单元的 MAD 值用 MAD_{pb} 表示, 则线性预测模型表示如下:

$$MAD_{cb} = a_1 \times MAD_{pb} + a_2 \quad (2-2)$$

其中 a_1 和 a_2 这两个系数的初始值相应设为 1 和 0。当编码每一基本单元时它相应的进行更新。公式(2-2)所示的的线性模型可以解决“鸡与蛋”的悖论。

4 流量往返模型

流量往返模型的作用是计算当前编码帧的目标码率 (target bits)。将 N_{gop} 定义为一个图像组 (GOP) 总帧数, 其中 $n_{i,j} (i=1,2,\dots, j=1,2,\dots, N_{gop})$ 表示第 i 个图像组的第 j 帧, $B_c(n_{i,j})$ 表示编码第 j 帧以后虚拟缓冲区的占用情况。可以得到如下关系:

$$\begin{aligned} B_c(n_{i,j+1}) &= B_c(n_{i,j}) + b(n_{i,j}) - \frac{u(n_{i,j})}{F_r} \\ B_c(n_{1,1}) &= B_s / 8 \\ B_c(n_{i+1,0}) &= B_c(n_{i,N_{gop}}) \end{aligned} \quad (2-3)$$

这里的 $b(n_{i,j})$ 为第 i 个 GOP 中第 j 帧得到的比特数, $u(n_{i,j})$ 表示的是 VBR 或者 CBR 模式中的可用信道带宽, F_r 为预先设定的帧率, B_s 为缓冲区大小。

需要指出的一般初始缓冲区中的比特数量默认设置成 $B_s/8$, 它也可以根据应用需要设定成其他值。 B_s 是缓冲区的尺寸, 它的最大取值是根据不同的 profile 和 level 给出的。在码率波动比较小的情况下, 可以把初始的缓冲器设置的比较小。

5 假想参考解码器 (HRD)

给每帧的目标比特数加一个下限和一个上限是为了满足假想参考解码器(HRD)的要求。每一帧目标比特的下界和上界由假定参考解码器 (hypothetical reference decoder, HRD) 决定, 分别用 $L(n_{i,j})$ 和 $U(n_{i,j})$ 表示第 n 帧的下界和上界。根据 HRD 模型, 实际每帧的比特应介于 $[L(n_{i,j}), U(n_{i,j})]$ 之间, 用 $T_r(n_i, n_j)$ 表示第 i 个 GOP 的第 j 帧到达的时间, $be(t)$ 为时间 t 的等价比特, 转换因子为到达缓冲器时的码率^[28]。上界和下界的初始值用下式表示:

$$L(n_{i,1}) = T_r(n_{i,0}) + \frac{u(n_{i,0})}{F_r}$$

$$U(n_{i,1}) = (T_r(n_{i,0}) + be(t_r(n_{i,1}))) \times \varpi \quad (2-4)$$

其中 $T_r(n_{i,0})$ 是第 $(i-1)$ 个 GOP 残留的比特, 其中 $T_r(n_{1,0}) = 0$ 、 $\varpi = 0.9$ 。

$L(n_{i,j})$ 和 $U(n_{i,j})$ 由下式反复计算得到:

$$L(n_{i,j}) = L(n_{i,j-1}) + \frac{u(n_{i,j-1})}{F_r} - b(n_{i,j-1})$$

$$U(n_{i,j}) = U(n_{i,j-1}) + (\frac{u(n_{i,j-1})}{F_r} - b(n_{i,j-1})) \times \varpi \quad (2-5)$$

其中 B_s 为缓冲区大小, 根据不同的 LEVEL 和 PROFILE, 它的最大值不同。

2.2.2 H.264 的码率控制算法的具体步骤

在 H.264 中码率控制的基本步骤分为以下 5 个步骤:

- 1 通过流量往返模型(公式 2-3)和线性跟踪理论计算当前预编码帧的目标比特数;
- 2 将剩余比特数平均分配给当前预编码帧的未编码宏块;
- 3 通过线性模型(公式 2-2)预测出当前帧的 MAD 值;
- 4 通过 R-D 二次模型计算 QP 等相关参数;
- 5 利用已经得出 QP 值对每个基本单元执行率失真, 然后反复执行步骤 4。直至编码结束。

JVT-G012 提案默认适用于 GOP 层和基本单元层。当基本单元选择的是整帧时, 提案适用于 GOP 层, 帧层。当基本单元选择的是宏块时, 提案在适用于 GOP 层、帧层的同时还适用于基本单元层。

1、GOP 层的码率控制

GOP 层码率控制的目标是计算每个 GOP 中分配给未编码帧的比特数, 并决定每个 GOP 的初始量化参数。假设 GOP 结构是 IBBPBBP...P 或者 IPPP...P, GOP 的长度一般是 15-30。

(1) GOP 层总的比特数

在第 i 个 GOP 的开始, 分配给这个 GOP 的总比特数可用一下公式计算得出:

$$T_r(n_{i,0}) = \frac{u(n_{i,1})}{F_r} * N_{gop} - (\frac{B_s}{8} - B_c(n_{i-1, N_{gop}})) \quad (2-6)$$

其中 F_r 是帧率, N_{gop} 是第 1 个 GOP 中的总帧数, B_s 是缓冲区的大小, $B_c(n_{i-1, N_{gop}})$ 是编完第 i 个 GOP 后的缓冲区大小。从公式(2-6)可以看出, 后面的 GOP 的编码结果依赖于之前的 GOP。为了保证各个图像组有均衡的编码质量, 每个图像组只能使用分配给它的比特数。也就是说在编完每个图像组后缓冲器的占用值应该维持在初始缓冲区中的

比特数量 $\frac{B_s}{8}$ 。

因为信道带宽可能是时时变化着, T_r 是逐帧更新的:

$$T_r(n_{i,j}) = T_r(n_{i,j-1}) + \frac{u(n_{i,j}) - u(n_{i,j-1})}{F_r} (N_{gop} - j) - A(n_{i,j-1}) \quad (2-7)$$

在固定带宽情况下, $u(n_{i,j}) = u(n_{i,j-1})$, 公式(2-7)可以被简化为

$$T_r(n_{i,j}) = T_r(n_{i,j-1}) + \frac{u(n_{i,j}) - u(n_{i,j-1})}{F_r} (N_{gop} - j) - A(n_{i,j-1}) \quad (2-8)$$

(2) GOP 初始量化参数的计算

第一个 GOP 的 I 帧和第一个 P 帧的量化参数是一个已经设定好的 QP_0 。 QP_0 是根据可用的信道带宽和 GOP 长度来预定义的。通常, 如果可用信道带宽高较大, QP_0 应该相应的选择的较小, 使图像细节纹理显示的越充分; 如果信道带宽低, 则应该使用大的 QP_0 。在同样的带宽下, 如果 GOP 长度增加 15, 那么 QP_0 就相应减少 1。

$$QP_0 = \begin{cases} 40 & bpp < l1 \\ 30 & l1 < bpp < l2 \\ 20 & l2 < bpp < l3 \\ 10 & bpp > l3 \end{cases} \quad (2-9)$$

$$bpp = u(n_{(0,0)}) / F_r N_{pixel}$$

其中 N_{pixel} 是像素的个数。

表 2-1 $l1, l2, l3$ 取值

图像大小	$l1$	$l2$	$l3$
QCIF	0.1	0.3	0.6
CIF	0.2	0.6	1.2
大于 CIF	0.6	1.4	2.4

其余 GOP 的初始化量化参数 QP_{st} 按公式 (2-10) 计算:

$$QP_{st} = \frac{Sum_{PQP}}{N_p} - 1 - \frac{8T_r(n_{i-1, N_{gop}})}{T_r(n_{i,0})} - \frac{N_{gop}}{15} \quad (2-10)$$

N_p 为前一个 GOP 中的 P 帧总数, Sum_{PQP} 是前一个 GOP 中所有 P 帧的量化参数之和。

2、帧层码率控制

帧层码率控制分为前编码和后编码两个阶段。对于前编码的目标是为即将编码帧分配目标编码比特, 针对 B 帧和 P 帧分为两种不同的方法。后编码的任务是: 修改预测 MAD 模型中的两个参数; 修改 R-D 二次模型中的参数; 决定是否应该跳帧。

在前编码中, 针对 B 帧时计算量化参数是非常简便的。当编码帧为 P 帧时, 在帧

级比特控制中需要预测公式预测该帧的 MAD 值并根据分配的目标编码比特利用 R-Q 二次模型计算即将编码帧的量化参数;如果码率控制的基本单元不是一帧,该帧的第一个基本单元使用上一帧的平均量化参数进行编码,其余基本单元的量化参数通过基本单元层码率控制计算得到。

(1) B 帧量化参数计算

由于 B 帧不被用来进行帧间预测,为了为 I 帧和 P 帧节约一些比特数,它们的量化参数可以比与它相邻的 P 帧或者 I 帧大一些。另一方面,出于保持视频质量的平滑的愿意,两个临近帧之间的量化参数之差不应该大于 2。

假设两个 P 帧之间连续 B 帧的数量是 L,两个 P 帧的量化参数分别是 QP_1 和 QP_2 。第 i 个 B 帧的量化参数根据下面两种情况来计算:

情形 1 $L=1$, 即两个 P 帧之间只有一个 B 帧。B 帧 QP 按下式计算:

$$\tilde{QB}_1 = \begin{cases} \frac{QP_1 + QP_2 + 2}{2} & \text{如果 } QP_1 \neq QP_2 \\ QP_1 + 2 & \text{其它} \end{cases} \quad (2-11)$$

情形 2 $L>1$, 量化参数按公式 (2-12) 式计算:

$$\tilde{QB}_i = QP_1 + \alpha + \max\{\min\{\frac{QP_2 - QP_1}{L-1}, 2(i-1)\}, -2(i-1)\} \quad (2-12)$$

其中 α 是第一个 B 帧的 QP 和 QP_1 之间的差值,由公式 (2-13) 给出:

$$\alpha = \begin{cases} -3 & QP_2 - QP_1 \leq -2L - 3 \\ -2 & QP_2 - QP_1 = -2L - 2 \\ -1 & QP_2 - QP_1 = -2L - 1 \\ 0 & QP_2 - QP_1 = -2L \\ 1 & QP_2 - QP_1 = -2L + 1 \\ 2 & \text{其它} \end{cases} \quad (2-13)$$

最终的量化参数 QB_i 被调整为:

$$QB_i = \min\{\max\{\tilde{QB}_i, 1\}, 51\} \quad (2-14)$$

(2) P 帧量化参数计算

P 帧需要估计目标比特,通过二次模型得到量化参数。目标比特的计算分以下两步计算:

首先为每一个 P 帧计算目标比特。它又分为宏观控制和微观控制两个部分。

a 宏观控制(比特分配)

对每个 P 帧可以定义一个目标缓冲级(target buffer level, Tbl) 进行比特分配,通过目标比特来计算量化参数。第一个 P 帧的量化参数在 GOP 层给出,所以只需要对 GOP 中其余 P 帧预定义目标缓冲级。

在编码完第 i 个 GOP 的第一个 P 帧后,重新设置目标缓冲级的初始值为:

$$Tbl(n_{i,2}) = B_c(n_{i,2}) \quad (2-14)$$

$B_c(n_{i,2})$ 是在编码完第 i 个 GOP 中第一个 P 帧后实际的缓冲占用。

后面的 P 帧的目标缓冲级定义为：

$$Tbl(n_{i,j+1}) = Tbl(n_{i,j}) - \frac{Tbl(n_{i,2}) - B_s / 8}{N_p - 1} + \frac{\tilde{W}_p(n_{i,j})(L+1)u(n_{i,j})}{F_r(\tilde{W}_p(n_{i,j}) + \tilde{W}_b(n_{i,j})L)} - \frac{u(n_{i,j})}{F_r} \quad (2-15)$$

$\tilde{W}_p(n_{i,j})$ 是 P 帧平均复杂度因子， $\tilde{W}_b(n_{i,j})$ 是 B 帧平均复杂度因子， $Tbl(n_{i,j})$ 是目标缓存级。 \tilde{W}_p 和 \tilde{W}_b 计算如下：

$$\begin{aligned} \tilde{W}_p(n_{i,j}) &= \frac{W_p(n_{i,j})}{8} + \frac{7 * \tilde{W}_p(n_{i,j-1})}{8} \\ \tilde{W}_b(n_{i,j}) &= \frac{W_b(n_{i,j})}{8} + \frac{7 * \tilde{W}_b(n_{i,j-1})}{8} \\ W_p(n_{i,j}) &= S_p(n_{i,j})Q_p(n_{i,j}) \\ W_b(n_{i,j}) &= \frac{S_b(n_{i,j})Q_b(n_{i,j})}{1.3636} \end{aligned} \quad (2-16)$$

S_p 和 S_b 是编码相应帧产生的比特数， Q_p 和 Q_b 是相应帧的量化参数。在两个 P 帧间没有 B 帧的情况下，公式(2-16)可以被简化为：

$$Tbl(n_{i,j+1}) = Tbl(n_{i,j}) - \frac{Tbl(n_{i,2}) - B_s / 8}{N_p - 1} \quad (2-17)$$

如果缓冲器的实际占用率和预先所确定的一样的话可以保证每一个 GOP 使用它自己的预算。然而，因为 R-D 模型以及 MAD 线性预测模型并不能保证完全精确，缓冲器的实际占用率和目标缓冲级之间就经常存在差别。这就需要微观控制来计算每一帧的目标比特以减少实际缓冲值和目标缓冲级之间的差值。

b 微观控制

微观控制的目标是计算 P 帧的目标比特数。根据线性跟踪理论，下面几个因素决定了第 i 个 GOP 中第 j 帧分配的目标比特，其中包括：目标缓冲级、帧率、可用带宽、实际缓存占用。如下式所示：

$$\tilde{f}(n_{i,j}) = \frac{u(n_{i,j})}{F_r} + \gamma(Tbl(n_{i,j}) - B_c(n_{i,j})) \quad (2-18)$$

γ 是常数，典型值为没有 B 帧的情况下取值为 0.75，其它情况取值为 0.25。如果产生的实际比特数与目标比特数相近，很容易得到：

$$B_c(n_{i,j+1}) - Tbl(n_{i,j+1}) \approx (1 - \gamma)(B_c(n_{i,j}) - Tbl(n_{i,j})) \quad (2-19)$$

同时，在计算目标比特的还应该考虑剩余的比特数：

$$\hat{f}(n_{i,j}) = \frac{W_p(n_{i,j-1})T_r(n_{i,j})}{W_p(n_{i,j-1})N_{p,r}(j-1) + W_b(n_{i,j-1})N_{b,r}(j-1)} \quad (2-20)$$

目标比特是由 $\tilde{f}(n_{i,j})$ 和 $\hat{f}(n_{i,j})$ 两部分加权和组成:

$$f(n_{i,j}) = \beta * \hat{f}(n_{i,j}) + (1 - \beta) * \tilde{f}(n_{i,j}) \quad (2-21)$$

β 是常数, 没有 B 帧时取 0.5, 其它情况下取 0.9。

在目标比特数确定以后需要计算量化参数并进行 RDO。

当前 P 帧的量化参数 \hat{Q}_{pc} 是由前一个 P 帧的实际 MAD 值用二次 R-D 模型计算得来, 因此为了保持连续帧之间视频质量的平滑, 将量化参数调整成下述公式:

$$\tilde{Q}_{pc} = \min\{Q_{pp} + 2, \max\{Q_{pp} - 2, \hat{Q}_{pc}\}\} \quad (2-22)$$

其中, Q_{pp} 为前一个 P 帧的量化参数。最终的量化参数 Q_{pc} 需要进一步限制界限:

$$Q_{pc} = \min\{51, \max\{\tilde{Q}_{pc}, 1\}\} \quad (2-23)$$

在得到了量化参数后对每一个帧进行 RDO。

编码模式通过最小化下面的性能指标来选择:

$$D(s, c, MODE | QP) + \lambda_{\text{mode}} R(s, c, MODE | QP) \quad (2-24)$$

$$\begin{aligned} \lambda_{\text{mode}} &= 0.85 \times 2^{QP/3}, & \text{I, P 帧} \\ \lambda_{\text{mode}} &= 4 \times 0.85 \times 2^{QP/3}, & \text{B 帧} \end{aligned} \quad (2-25)$$

在进行帧层码率控制后, 可以得到当前帧的量化参数 QP , 但是如果需要更加精确的进行码率控制算法, 则需进行基本单元层的码率控制。

3、本单元层码率控制

如果选择基本单元不是帧层, 那还需要进行基本单元层的码率控制。基本单元层的码率控制是同帧层码率控制目标一样, 是为了各个基本单元层选择量化参数。在基本单元层的码率控制中, I 帧和 B 帧都使用统一的量化参数进行编码。其它的计算和帧层基本一致。不同的是, QP_1 和 QP_2 被相应的基本单元的量化参数平均值所取代。

对于 P 帧的码率控制, 由于当前帧的未编码单元的 MAD 是未知的, 故将剩余比特数平均分配给所有未编码的基本单元。以下将描述如何对所有 P 帧进行基本单元层码率控制。

(1) 计算当前基本单元的纹理比特数 $R_{i,j}$ (Texture Bits)。这步分为以下三个子步:

第一步 计算当前基本单元的目标比特

设 $f_{rb}(n_{i,j})$ 为当前帧中所有未编码基本单元的剩余比特数, 当前帧中未编码基本单元的个数为 N_{ub} , $f_{rb}(n_{i,j})$ 和 N_{ub} 的初始值分别为 $f(n_{i,j})$ 和 N_{init} 。当前基本单元的目标比特数为: f_{rb} / N_{ub} 。

第二步 计算所有已编码的基本单元产生的平均头比特数

$$\begin{aligned}\tilde{m}_{hdr,l} &= \tilde{m}_{hdr,l-1} \left(1 - \frac{1}{l}\right) + \frac{\hat{m}_{hdr,l}}{l} \\ m_{hdr} &= \tilde{m}_{hdr,l} \frac{l}{N_{unit}} + m_{hdr,l} \left(1 - \frac{l}{N_{unit}}\right)\end{aligned}\quad (2-26)$$

$\hat{m}_{hdr,l}$ 是当前帧中第 1 个基本单元产生的实际头比特数, $m_{hdr,l}$ 是前一帧中所有基本单元得出的估计值。

第三步 计算纹理比特数 $R_{i,l}$:

$$R_{i,l} = \frac{f_{rb}}{N_{ub}} - m_{hdr} \quad (2-27)$$

(2) 使用 MAD 线性模型预测当前基本单元的 MAD。

(3) 使用二次 R-D 模型计算当前基本单元的量化参数。分三种情况。

第一种情况 当前帧的第一个基本单元。

$$Q_{cb} = Q_{apf} \quad (2-28)$$

Q_{apf} 是前一帧中所有基本单元量化参数的平均值。

第二种情况 $f_{rb} < 0$ 。

这种情况下, 量化参数应该比前一个基本单元的大, 以使得产生的比特数之和接近 $f(n_{i,j})$, 也就是

$$\hat{Q}_{cb} = Q_{pb} + DQuant \quad (2-29)$$

Q_{pb} 是前一个基本单元的量化参数。为了减少块效应, 如果 N_{unit} 大于 8, 则 $DQuant$ 为 1; 否则, $DQuant$ 为 2。

为了维持视频质量的平滑, 对量化参数进一步限制:

$$Q_{cb} = \max\{1, Q_{apf} - \Delta, \min\{51, Q_{apf} + \Delta, \hat{Q}_{cb}\}\} \quad (2-30)$$

如果 N_{mbunit} 小于每一行的宏块数, $\Delta=3$; 否则 $\Delta=6$ 。

第三种情况其它情况。

首先使用二次 R-D 模型计算量化参数 \hat{Q}_{cb} 。然后为了减小块效应, \hat{Q}_{cb} 被限制在一定范围内:

$$\tilde{Q}_{cb} = \max\{Q_{pb} - DQuant, \min\{\hat{Q}_{cb}, Q_{pb} + DQuant\}\} \quad (2-31)$$

同时, 为了维持视频的平滑性, 进一步限定:

$$Dquant = \max(-2, \min(2, Dquant)) \quad (2-32)$$

(4) 对当前基本单元中所有宏块进行 RDO。

(5) 对当前帧中剩余比特数和未编码基本单元数进行更新。

(6) 在编码完整个帧后, Q_{apf} 被更新。

为了在平均 PSNR 和比特波动间取得权衡, 对于实时视频通信, JVT-G012 推荐

N_{mbunit} 为帧中一行中宏块的数, N_{unit} 推荐为 9。

2.3 率控制算法

码率控制在各种编码标准中都有所应用, 不同的应用需求也设定了不同的算法和模型。到现在为止, 经典的码率控制算法主要有: MPEG-2 的 TM5^[28]; MPEG-4 的 VM8^[29]、VM15^[30]; H.261 的 RM8^[31]; H.263 的 TMN8^[32]、TMN11 等。

2.3.1 MPEG-2 TM5 算法

MPEG-2 视频编码器采用的是 TM5 算法作为它的码率控制策略。TM5 算法主要引入了复杂度分析、比特分配、自适应量化等方法并且可以较好的达到率控效果。TM5 算法由三部分完成调整量化参数控制码率的过程:

首先是复杂度分析。TM5 算法在编码前, 根据预测的图像复杂度给 I、P、B 帧估计所分配的比特数。

$$\begin{cases} X_i = (160 * bit_rate) / 115 \\ X_p = (60 * bit_rate) / 115 \\ X_b = (42 * bit_rate) / 115 \end{cases} \quad (2-33)$$

X_i 、 X_p 、 X_b 分别为 I、P、B 帧的全局复杂度估计, bit_rate 为编码目标比特率。由于在未发生场景切换的情况下, 视频具有较好的时间相关性。因此编码帧的全局相关性可以从最近已编码帧的同类型帧确定。当某个类型的帧已经编码后, 可以通过实际所需要的比特数和该帧量化参数的平均值更新全局复杂度。

$$X_{i,p,b} = S_{i,p,b} * Q_{i,p,b} \quad (2-34)$$

S 表示的是编码后实际比特数 Q 表示的是平均量化参数。

第二步计算下一帧图像的目标比特数。

$$\begin{aligned} T_i &= \max \left\{ \frac{R}{1 + \frac{N_p X_p}{K_p X_i} + \frac{N_b X_b}{K_b X_i}}, \frac{bit_rate}{8 * frame_bit} \right\} \\ T_p &= \max \left\{ \frac{R}{N_p + \frac{N_b K_p X_b}{K_p X_p}}, \frac{bit_rate}{8 * frame_bit} \right\} \\ T_b &= \max \left\{ \frac{R}{N_b + \frac{N_p K_b X_p}{K_p X_b}}, \frac{bit_rate}{8 * frame_bit} \right\} \end{aligned} \quad (2-35)$$

T_i 、 T_p 、 T_b 分别为分配给当前相对类型帧的目标比特数; N_p 、 N_b 为在当前 GOP 中为编码的 P、B 帧数目。 K_p 、 K_b 是两个通用常量, 分别设置成 1.0 和 1.4。R 是剩余比特

数。在每编码一帧后，R 将进行更新：

$$R = R - S_{i,p,b} \quad (2-36)$$

其中 $S_{i,p,b}$ 为当前编码帧所用的比特数。

若第一个 GOP 进行编码前，则该 GOP 能分配到的比特数 R：

$$G + R = R \quad (2-37)$$

其中 $G = \text{bit_rate} * N_{p,b} / \text{frame_rate}$ ， $N_{p,b}$ 代表的是 P 帧和 B 帧数目。R 的初始值为 0。

在给每种类型个帧分配了比特数后，TM5 在宏块级根据缓冲区状态为每个宏块决定量化参数。当前宏块的分配比特公式是：

$$d_j^{i,p,b} = d_0^{i,p,b} + B_{j-1} \left[\frac{T_{i,p,b} * (j-1)}{MB_cnt} \right] \quad (2-38)$$

式中的 $d_j^{i,p,b}$ 为当前帧的第 j 个宏块编码后 Buffer 的充溢程度， $d_0^{i,p,b}$ 当前帧编码编码前 Buffer 的充溢程度， B_{j-1} 为前 j-1 个宏块实际编码比特数， $T_{i,p,b}$ 为当前帧目标比特数， MB_cnt 为宏块总数。当前帧最后的缓冲区的饱和度 $d_j^{i,p,b}$ ($j=MB_cnt$) 用于编码下一个同类型帧。当编码第一个帧时， $d_j^{i,p,b}$ 初始化为：

$$\begin{aligned} d_0^i &= 10 * r / 31 \\ d_0^p &= K_p * d_0^i \\ d_0^b &= K_b * d_0^p \end{aligned} \quad (2-39)$$

根据上面的公式，量化参数计算为：

$$Q_j = \left(\frac{d_j * 31}{r} \right) \quad (2-40)$$

其中 $r = 2 * \frac{\text{bit_rate}}{\text{frame_rate}}$

最后自适应量化

自适应量化根据计算当前宏块的帧结构和场结构的共 8 个 8×8 亮度子块的均方差，然后将所得最小方差值与平均方差值进行归一化，获得自适应的量化参数的调节系数。

这个算法选择了非常简单的 R-D 模型，容易实现，对恒定码率的码率控制显示了较好的效果。但是没有考虑场景切换时的视频压缩问题，对于同一个图像中的宏块是根据缓冲器的充溢程度进行比特分配，这将致图像上半部分被相对细量化，而下半部分被粗量化，使得图像上下部分失衡，从而影响图像整体质量。

2.3.2 MPEG-4 VM8 算法

VM8 算法作为 MPEG-4 的码率控制策略，由于 MPEG-4 与以往标准的一个重要区

别就是它是一个基于对象的视频压缩标准，因此 VM8 算法必须扩展到形状编码和多个视频对象编码 (VO)。下图是 VM8 的码率控制系统图。

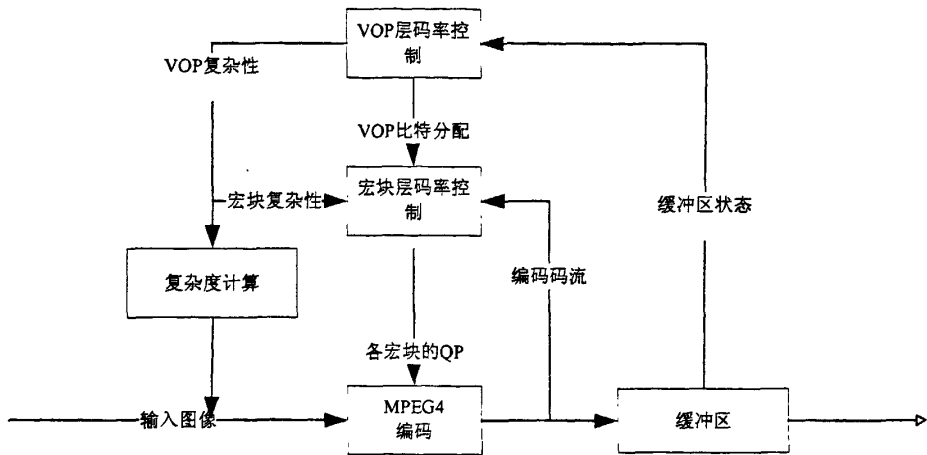


图 2-3 VM8 码率控制系统图

VM8 码率控制策略在很多地方继承了 TM5 算法，在码率分配方面基本相同。但是由于 VM8 采用了二次 R-D 模型，因此该模型可以提高目标比特率的估计准确程度。

在此主要介绍一下 VM8 中的 VOP 层码率控制。

VOP 层码率控制主要包括 5 个主要步骤：

1 初始化

需要初始化的有：初始化缓冲区大小为 $B_s/2$ ；初始化第一 I 帧的量化参数，一般设定为 15；给 R-D 模型系数分配预设的值。在第一帧编码完成后，所剩余的比特数目为： $B_r = B - B_f$ ，其中 B_f 是编码 I 帧所用的比特。

2 计算当前帧所需要的比特数

这个步骤又分为 2 个子步骤。第一需要根据上一帧的编码比特数和剩余比特数进行调整、同时每帧所分配的比特数又不得少于最低比特数 $B/(2 \times N)$

若前一帧的编码占用比特数比较多，剩余可用的比特数较少，当前帧的目标比特则确定为： $B_t = \max(\frac{B}{2N}, \frac{B_r}{N_f} \times \beta + B_f \times (1 - \beta))$ (2-41)

其中 β 是属于区间 $[0,1]$ 的权重系数；其数值可以根据实验结果确定； N_f 是一个 GOP 中剩余的待编码 P 帧的个数。

第二阶段是对所分配的比特进行微调：

$$B_t = B_f \frac{B_c + 2 \times (B_s - B_c)}{2 \times B_c + (B_s - B_c)} \quad (2-42)$$

其中 B_c 为当前缓冲区的饱和程度。当 B_c 大于 $B_s/2$ ，要减小 B_t ，反之则要增加 B_t 。

B_i 在确定之后, 仍然要为了避免缓冲区出现上溢或者下溢, 将上下限设置为 $0.9 B_s$ 和 $0.1 B_s$,

此时给每帧分配的比特数目为:

$$B_i = B_i \frac{B_c + 2 \times (B_s / 2 - B_c)}{2 \times B_c + (B_s - B_c)}$$

$$\begin{cases} B_i = \max\left(\frac{B}{2N}, 0.9 \times B - B_c\right) & B_i + B_c > 0.9 \times B_s \\ B_i = 0.1 \times B - B_c + R_p & B_i + B_c - R_p < 0.1 \times B_s \end{cases} \quad (2-43)$$

其中, R_p 是每帧编码后从缓冲区进入传输的比特数。

3 QP 值的计算并对当前帧编码

目标比特数确定以后, 通过率失真模型可以计算得出量化参数 Q :

该算法的不足之处是不适应宏块级的控制, 因此比特数的估计准确度不如 TMN8, 另外, 由于该模型成立的前提, 在场景变化快的情形非常容易导致 QP 估计的不准确。

$$Q = \begin{cases} \frac{2a_2 \times S}{\sqrt{(a_1 \times S)^2 + 4a_2 \times S(B_i - H)}} & a_2 = 0 \text{ 或 } (a_1 \times S)^2 + 4a_2 \times S(B_i - H) < 0 \\ \frac{a_1}{S \times (B_i - H)} & \text{else} \end{cases} \quad (2-44)$$

其中 S 是当前帧的编码复杂度, H 是头信息等其他非纹理信息所用的编码比特数目; Q 是计算的量化参数。

对于量化参数的计算最后就将计算得到的量化参数控制在 1-31 之间, 同时需要将此帧的量化参数限定在前一帧的 QP 量化值的正负 25% 以内, 以防视频质量的波动太大

$$Q = \max\{\min\{Q_{prev} \times 1.25, Q, 31\}, Q_{prev} \times 0.75, 1\}$$

$$\eta = \min\left(\frac{S}{S_{prev}}, \frac{S_{prev}}{S}\right) \quad (2-45)$$

$$W = \min\{W_{prev} + 1, \eta \times \max_sliding_window\}$$

其中 Q_{prev} 指的是前一帧的量化参数。

4 更新当前的模型参数

首先设定的滑动窗口和数据点的选择, 设:

$$\eta = \min\left(\frac{S}{S_{prev}}, \frac{S_{prev}}{S}\right) \quad (2-46)$$

S_{prev} 是前一帧的编码复杂度，且将滑动窗口的大小设定为：

$$W = \min \{W_{prev} + 1, \eta \times \max_sliding_window\} \quad (2-47)$$

$\max_sliding_window$ 的计算当中涉及两个常数其中

$$a_2 = \frac{W \sum_{i=1}^W \frac{T(i) - H(i)}{S(i)} - (\sum_{i=1}^W Q(i)^{-1}) (\sum_{i=1}^W Q(i) \frac{T(i) - H(i)}{S(i)})}{W \sum_{i=1}^W Q(i)^{-2} - (\sum_{i=1}^W Q(i)^{-1})^2} \quad \text{取值为 20;}$$

$$a_1 = \frac{\sum_{i=1}^W (Q(i) \frac{T(i) - H(i)}{S(i)} - a_2 Q(i)^{-1})}{W}$$

W 为当前滑动窗口的大小尺寸； W_{prev} 为前一帧滑动窗口的大小。

通过滑动窗口选中的 W 个数据点可以计算模型参数，编码器根据这些数据点确定了量化参数和实际比特消耗的统计值，根据最小二乘法，可以求取最佳的模型参数。

5 控制跳帧

如果缓冲区饱和度加上待编码帧的目标比特数大于预先设定的阈值（80%）就会跳过待编码帧以避免缓冲区出现溢出现象。

2.3.3 TMN8 码率控制算法

TMN8 是 H.263 视频编码中经典的码率控制算法。该算法采用二次率失真模型。下图为 TMN8 码率控制示意图：

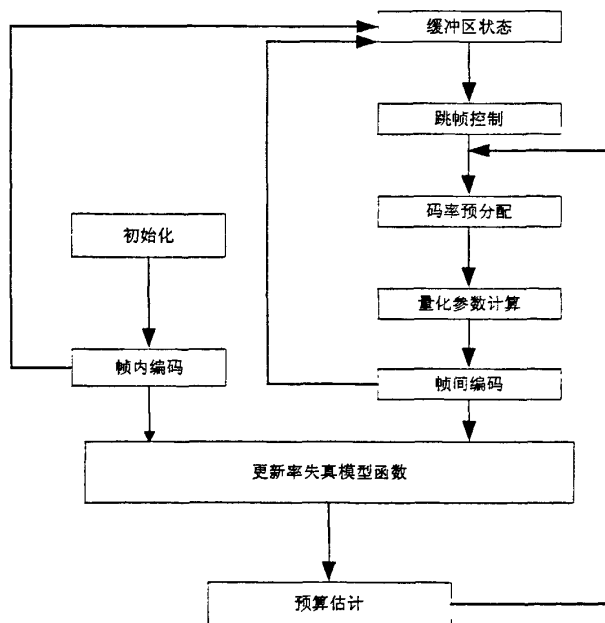


图 2-4 TMN8 码率控制示意图

从上图可知, TMN8 主要分为两个部分: 帧级码率控制和宏块级码率控制。帧级码率控制主要进行码率分配的工作, 它是根据带宽和现在缓冲区大小进行分配的。同时在帧级码率控制中还需要控制跳帧数目。宏块级别的码率控制是分配给各个宏块比特

1 帧级码率控制:

帧级码率控制中, 首先应该计算缓冲区饱和度 W :

$$W = \max(W_{prev} + B - R/F, 0) \quad (2-48)$$

B 为前一帧编码实际所用的比特数目; R 为信道或目标码率; F 为帧率; w_{prev} 为前一帧编码前缓冲区中的比特数。取最大值是为了避免缓冲区饱和度出现零值。

在 TMN8 中采用了跳帧算法避免缓冲区上溢的问题。当缓冲区超过了门限值 M , 进行跳帧。其中 $M = R/F$ 。

在计算得到缓冲区饱和度以后, 将要计算当前帧所需要的比特数 B : $B = R/F - \Delta$ F 为帧率, R 为目标比特率, Δ 为

$$\Delta = \begin{cases} W - 0.1 * R/F & (other) \\ W/F & (W > 0.1 * R/F) \end{cases} \quad (2-49)$$

2 宏块级码率控制

宏块级码率控制基于失真模型, 同时考虑了输出码率和图像质量, 使得在各个宏块总比特数于帧级所分配的比特数同时可以获得最小的失真。

编码每个宏块比特数 B_i 和量化参数 Q_i 之间存在下式所表述的数学关系

$$B_i = A(k \frac{\sigma_i^2}{Q_i^2} + C) \quad (2-50)$$

其中, A 为宏块的像素个数。 K 、 C 为可更新的常数。 K 是宏块中像素的统计规律, 满足 $0 \leq K \leq e/\ln 2$, C 是每个宏块包括头信息和运动矢量在内的比特数。 σ_i 是帧内宏块的亮和色度像素或者是运动补偿信息的标准差

$$\sigma_i = \sqrt{\frac{1}{16*16 + 8*16} \left(\sum_{j=1}^{16*16} (P_{li}(j) - \bar{P}_l)^2 + \sum_{j=1}^{8*16} (P_{ci}(j) - \bar{P})^2 \right)} \quad (2-51)$$

$P_{li}(j)$ 指的是第 i 个宏块中第 j 个像素的亮度值, $P_{ci}(j)$ 是第 i 个宏块中第 j 个像素的色度值, \bar{P} 指得是色度的平均值,

$$\bar{P}_l = \frac{1}{16*16 + 8*16} \left(\sum_{j=1}^{16*16} P_{lj}(j) + \sum_{j=1}^{8*16} P_{ij}(j) \right) \quad (2-52)$$

对一帧图像而言。该帧所需要的编码比特数 B 和失真函数模型为

$$B = \frac{1}{N} \sum_{i=1}^N B_i$$

$$D = \frac{1}{N} \sum_{i=1}^N \alpha_i^2 \frac{B_i^2}{12}$$
(2-53)

其中 N 为编码的宏块数目， α_i 为各个宏块的编码权重。

$$\alpha_i = \begin{cases} \frac{B}{16^2 N} (1 - \sigma_i) + \sigma_i & \frac{B}{16^2 N} < 0.5 \\ 1 & \text{其他} \end{cases}$$
(2-54)

上式中的 B 为图像的目标比特数。

根据上边介绍的公式可以得到量化参数

$$Q_i^* = \sqrt{\frac{256k}{L_i} \frac{\sigma_i}{\alpha_i} s_i}$$

其中 s_i 为剩余宏块的加权能量和， $L_i = B_i - 256N_i$ 为除去头信息，运动补偿向量以外的剩余可用比特数。

由于相邻宏块间的 QP 差值 D_{quant} 绝对值不能超过 2，因此

$$QP = \text{round}(Q_i^* / 2)$$

$$D_{quant} = QP - QP_{prev}$$

$$D_{quant} = \max(-2, \min(2, D_{quant}))$$
(2-55)

$$QP = QP_{prev} + D_{quant}$$

当 $L_i \leq 0$ 时，表示此时没有可用比特，于是令 $Q_i^* = 2(QP_{prev} + 2)$ 。

TMN8 是一种低码率情况下广泛使用的码率控制算法。该算法适用于 CBR 和 VBR 模式，结合了码率模型和率失真模型，可以在接近目标码率的同时保证失真最小的情况下选择量化参数。且实验结果显示这种方法可以有效提高编码效果。TMN8 算法在保证输出码率的同时也尽量的减少了跳帧，这样可以保持主客观质量的稳定和视频观看的流畅。但是该算法还是有需要改进的地方，这种算法固定帧率平均分配码率使得缓冲器内的比特数基本保持恒定有利于低延迟，但是对于图像纹理信息较多，运动剧烈的图像而言重建质量不高，且这种帧层比特分配将使得运动缓慢，纹理信息不多的帧质量很高，这样两者图像之间的图像质量落差较大，容易造成图像的闪烁。TMN8 引入了二次率失真，算法复杂度高不利于实时传输和无线视频通信。由于 TMN8 的这些不足，在实际应用中具有一定的局限性。

2.4 本章小结

本章首先介绍了码率控制基本原理，接着对 H.264 编码技术的码率控制基本概念和 JVT-G012 方案进行了介绍，最后，介绍了几种经典的码率控制算法。

第 3 章 基于人脸感兴趣区域的码率控制算法

随着视频传输服务的快速增长,各种类型的应用不断涌现,但在带宽较窄的情况下,视频整体质量并不能完全的满足人们的需求。在这种情况下人们开始日益关注感兴趣区域(ROI)的视频质量,希望利用各种方法提高感兴趣区域的视频质量,满足主观要求。一种有效的码率控制算法可以显著提高视频质量,因此基于感兴趣区域的码率控制技术已经成为了研究重点。在感兴趣区域的区域中,学者最为关注的就是人脸的区域,对于人脸区域进行码率控制提高视频质量可以广泛应用于基于内容的图像检索、视频监控、视频会议、可视电话以及人机交互等方面。

在这一章节中作者改进了一种针对人脸感兴趣区域的码率控制算法,通过实验结果可以发现这种算法可以有效的适用于低码率情况下的各种视频,有效的提高视频质量,满足了主观要求。

3.1 节介绍了感兴趣区域码率控制技术简介,3.2 节介绍了感兴趣区域的提取算法,3.3 节介绍了针对人脸感兴趣区域的码率控制算法,这种算法分为帧层和基本单元层两个方面。通过两个方面的对控制,可以从 3.4 节看出算法取得了良好的实验结果。

3.1 感兴趣区域码率控制技术简介

感兴趣区域研究源于人类的视觉特性。据研究,人眼的移动会将视觉注意力集中的画面部分映射到视网膜中央窝(fovea)上,这样这个部分将会有更高的视觉分辨率,这部分画面称为感兴趣区域。从图像应用上来说,感兴趣区域由于集中了人眼的较高的注意力因此是图像中的重要区域。在视频电话,可视电话等应用一般关注的头肩区域变化。

随着人类视觉研究的深入和视频分析技术的发展,研究者能够将图像处理算法应用在视频处理和机器视觉两个方面,进行自适应感兴趣区域的提取^[33]。在 MPEG4 标准中已经定义了视频对象的概念,对于这个概念已经有学者提出了集成感兴趣区域划分的算法^[34]。

在码率控制方面,在低码率情况下,由于带宽有限必须合理的给已知的感兴趣区域分配更多的比特,提高主观质量。在现在的一些研究中,有学者根据不同的视频编码标准提出不同的算法:MPEG4 采用了基于视频对象的编码,可以方便的将感兴趣区域整体作为一个视频对象独立进行编码;JPEG2000 可以使用小波变换达到空间分级,将感兴趣区域作为一个层次进行编码,这种基于小波变换的感兴趣区域编码方式在文献[35][36]有所介绍。H.263,H264 都是基于宏块的编码方式,文献[37-38]都是关于这些算法的介绍。

本文是基于 H.264 宏块编码的特性,以宏块为最小单元进行码率控制和比特分配。

3.2 感兴趣区域的提取

3.2.1 经典的人脸区域检测算法介绍

运动人脸的检测是目前一个热点研究问题, 现有的人脸检测算法^[40-49]主要归纳分为以下几类:

1. 基于肤色特征的方法

肤色特征是人脸的重要特征, 利用肤色检测人脸也是一种较为常见的方法。这种方法不依靠面部的细节特征, 由肤色模型描述。具有较好的稳定性, 对于旋转、表情等变化情况都可以适应。

肤色模型主要有高斯模型, 混合高斯模型和直方图模型。在文献[50]中 Cai 等作者根据肤色模型提供的像素似然度, 采用了从局部最大值逐渐扩展的方法得到肤色区域, 然后使用灰度平均脸模板匹配的方法验证区域内是否有人脸。但是肤色特征这种方法也存在一些需要考虑的问题: (1)由于光照的影响, 人脸可能被割裂为若干或不连通的肤色区域, (2)人脸区域可能与其它类肤色区域连接在一起。

2. 基于特征的方法

这种方法是根据人脸的先验知识, 判断图像中抽取的几何形状, 灰度, 纹理等特征来进行判断。

人脸区域的各个器官都有一些特定的特征, 比如五官的空间位置分布规则, 人脸的轮廓规则等。这种方法一般是先用边缘算子提取脸部的眉毛、眼睛、鼻子、嘴巴等脸部特征, 经过这个特征之间的联系利用统计模型检验人脸的存在性。这种方法的缺点是: 计算机量比较大, 受光线, 噪声, 遮挡的影响很大。

3. 基于特征空间的方法

该方法将人脸区域图像变换到某一特征空间, 根据其在特征空间中的分布规律划分为“人脸”与“非人脸”两类模式。常用主分量分析(PCA)这种方法。在进行正交变换(K-L 变换)后消除原有向量各个分量间的相关性。变换得到对应特征值依次递减的特征向量, 即特征脸。Moghaddam 等^[51]发现人脸在特征脸空间的投影聚集比较紧密, 利用人脸训练图像可以构造得到主子元空间 F 和与其正交的补空间, 这个主子元空间具有脸的基本形状, 且这种方法对于光线, 噪声的影响不大, 但是他是非局部化, 非拓扑的。

4. 基于人工神经网络的方法

神经网络方法在适用性上较广, 他可以通过多个检测器对人的正面, 半侧面, 侧面脸进行检测, 得到图像中的规律和规则的隐性表示。

5. 基于外观的方法 (Appearance - Based Method)。

该方法从一组训练图像中学习模型或模板以捕捉人脸外观上的可变性, 这种方法主要是用于人脸检测。基于外观的方法依赖统计分析和匹配学习来寻找相关的人脸和非人脸特征。

6 基于统计模型的方法

基于统计模型的方法是目前比较流行的方法，是解决复杂的人脸检测问题的有效途径。它具有如下优点：(1)采用实例训练的方法获取模型的参数，统计意义上更为可靠；(2)不依赖于人脸的先验知识和参数模型，可以避免不精确或不完整的知识造成的错误；(3)通过增加训练的实例可以扩充检测模式的范围，提高检测系统的鲁棒性。目前基于统计模型的方法主要针对正面人脸的检测，对于旋转人脸、多姿态人脸的检测由于比较复杂和困难，有效的方法还不多。

3.2.2 AdaBoost 算法描述

在现在的科学研究中 AdaBoost 算法是应用最为广泛的人脸检测算法，是一种自适应的 Boosting 算法，该算法利用大量分类能力一般的简单分类器通过一定方法进行叠加，从而构成了一个分类能力很强的强分类器。它源于 PAC 模型，在 PAC 模型中提到了强学习算法和弱学习算法以及二学习算法间的等价性问题。所谓强学习算法是指：存在一个多项式级的学习算法来识别一组概念，并具有很高的正确识别率；弱学习算法：识别的正确率仅比随机猜测略好；如果二者等价，就说明只要找到一个弱学习算法，就可以将其提升为强学习算法。

AdaBoost 算法是一种基于积分图、级联检测器的分类器算法，该算法基本思想：根据人脸面部的灰度分布特征，提取图像中的 Haar-like 特征，然后通过训练选出最优的 Haar-like 特征，再将这些训练得到的 Haar-like 特征转换成弱分类器，并且将得到的弱分类器进行优化组合叠加成为一个强分类器，再将若干强分类器串联成为级联分类器用于图像搜索检测。针对人脸检测问题，AdaBoost 设计了这样一个弱分类器：

$$h_j(x) = \begin{cases} 1 & p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases} \quad (3-1)$$

其中， $h_j(x)$ 表示弱分类器的值， x 表示一个样本， $f_j(x)$ 表示特征值。1 表示为人脸，0 表示为非人脸； θ_j 表示弱分类器的阈值； p_j 表示不等号的方向，取值为 -1 或 +1，如果正样本被分类在阈值以下， $p_j = +1$ ，否则 $p_j = -1$ ；

AdaBoost 的基本框架如下：

(1) 使用积分图像 (Integral Image) 计算矩形特征值 $f(x)$ ，描述人脸特征，以 Haar-like 特征为基础的矩形特征区分人脸和非人脸；

(2) 根据当前特征和权重分布的情况，选择一个最佳的弱分类器 (basic classifier)，确定最优阈值 θ^* ，并且根据加权投票方式叠加 (boost) 成为一个强分类器 (strong classifier)；

(3) 将训练得到的若干强分类器串联起来组合成一个层叠分类器 (cascade classifier) 用于人脸检测，通过设置每层的阈值可以使大部分人脸都能通过，同时这种

结构能有效地提高分类器的检测速度。

在本文中关于人脸区域的提取是在 H.264 参考软件 JM9 中融入 AdaBoost 算法进行人脸检测，其中的 AdaBoost 算法的实现基于 OpenCV 完成。

AdaBoost 算法的效果与前期训练时弱分类器的选取关系重大，而此时样本集的选取可以在一定程度上决定弱分类器。在 AdaBoost 算法中，人脸和非人脸的样本选取比较随机，不能很好的表达人脸和非人脸的信息，所以当人脸倾斜，转向或者遮挡时，很难定位人脸的位置，准确度需要进一步的提高。

3.2.3 本文采取的人脸区域检测算法

由于 AdaBoost 算法存在着不足，本文采取了文献[51]的算法进行人脸区域检测。他的基本思路是根据视频序列具有连续性的特显，利用视频前几帧的人脸区域的运动向量作为参考，预测当前帧的人脸区域位置。同时算法设置了帧间人脸区域帧差的阈值，结合了 AdaBoost 算法验证人脸区域描述的正确性，若发现误检则即使进行修正。

算法步骤如下：

(1) 视频序列的前两帧用 AdaBoost 算法进行人脸区域检测。

(2) 如果编码未结束，则进行：

a.按式(3-2)计算人脸区域的运动趋势，如果预测的人脸区域 V_{face} 运动向量过大转向

(3)，否则转向 b。

$$V_{face}(k) = -\frac{\sum_{(i,j) \in \text{faceregion}(k)} MB_MV_{(i,j)}(k) + \sum_{(i,j) \in \text{faceregion_IntMb}(k)} MB_MV'_{(i,j)}(k)}{L_face} \quad (3-2)$$

其中 $L_face < L$, L_face 是指一帧中人脸区域所占的宏块个数， L 帧中总宏块个数。

b.判断是否已有连续五帧是用预测方式得到，如果是，转向(3)，否则转向 c。

c.根据式(3-3)计算得到的运动向量，预测当前帧人脸区域位置，并用式(3-4)计算人脸区域帧差。将帧差值与阈值 $T1$ 进行比较，如果帧差值小于阈值 $T1$ ，将预测得到的人脸区域作为检测结果，编码当前帧并转向(2)。否则，转向(3)。

$$V_{face}(k+1) = -b * \frac{\sum_{(i,j) \in \text{faceregion}(k)} MB_MV_{(i,j)}(k) + \sum_{(i,j) \in \text{faceregion_IntMb}(k)} MB_MV'_{(i,j)}(k)}{L_face} \quad (k \geq 3) \quad (3-3)$$

b 为帧间相关系数, $(0.9 < \alpha < 1.0)$

$$\text{faceFD}(n) = \left(\sum_{(i,j) \in \text{faceregion}} [f_n^{(i,j)} - \hat{f}_{n-1}^{(i,j)}]^2 \right) / \text{facepix_nr} \quad (3-4)$$

人脸区域帧差 $\text{faceFD}(n)$ ，表示第 n 帧位于人脸区域的像素点 (i,j) 的图像像素值 $f_n^{(i,j)}$ 与第 $n-1$ 帧对应位置像素点的编码器重建值 $\hat{f}_{n-1}^{(i,j)}$ 之差的平方和。

利用这个平方和值除以当前帧中人脸区域所占的像素点数，即得平均每像素点处人脸区域帧差。 facepix_nr 表示人脸区域所占像素点数。

(3) 用 AdaBoost 算法检测人脸区域。

a. 如果检测无人脸区域，则用(2)中预测所得到的人脸位置值作为人脸检测结果，并转向 c. 否则转向 b。

b. 判断检测的人脸区域是否符合帧间人脸运动特征(是否出现人脸连通域的突变，是否出现检测区域的较大偏移)，若不符合，则用(2)中预测得到的人脸位置值作为检测结果，并转向 c. 若符合，则检测结果正确，编码当前帧并转向(1)。

c. 计算人脸区域帧差，如用帧差值小于阈值 T_2 ，则用(2)中预测得到的人脸位置值作为检测结果，编码当前帧，并转向(1)。否则，确定当前帧无人脸区域。

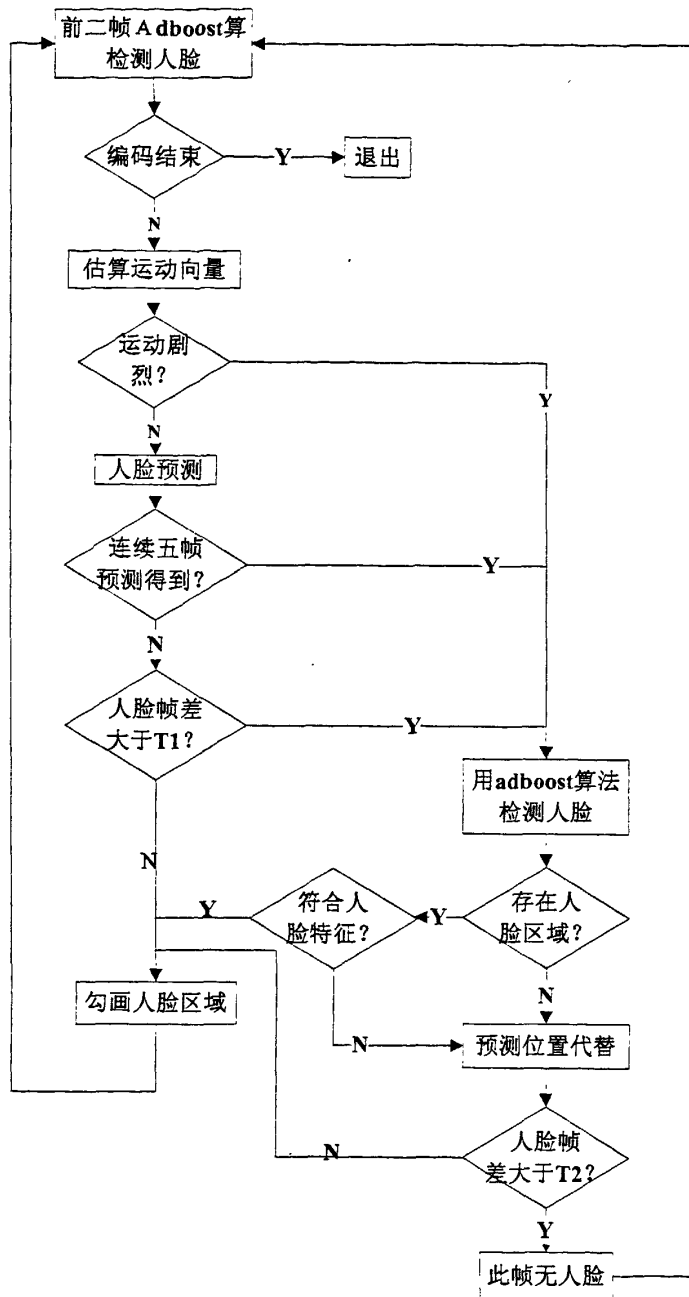


图 3-2 基于视频序列快速准确的人脸检测算法流程图

3.3 基于感兴趣区域的码率控制算法

3.3.1 帧级比特分配策略

在 H.264/AVC 中, 帧级目标比特的分配直接影响了量化参数 QP 的计算。通过第二章的介绍, 可以得知在原有的 H.264 码率控制算法中帧层比特分配没有考虑到图像

每帧的复杂度不一致的问题，采用了均分的方法。在本文中，由于主要应用于低带宽情况下的可视电话、视频会议等应用，应该对运动复杂度高的帧、宏块分配更多的比特，而不是简单的进行均分才能达到更好的码率控制效果。在本文预测了 MAD 值，对帧层比特进行了重新分配。其中每帧的目标比特由以下公式得到：

(3-5)

$$R_{sum} = \alpha \times R_1[i] + (1 - \alpha) \times R_2[i]$$

$$R_1[i] = \frac{W}{f} + \beta \times (B_t - B_c) \quad (3-6)$$

$$R_2[i] = \frac{R_r[i]}{N_r[i]} \times \gamma \quad (3-7)$$

其中 $R_{sum}[i]$ 是目标比特， α β 分别设置为 0.45, 0.7, W 为带宽， f 为帧率； B_t 是实际缓冲区占有率， B_c 是目标缓冲度； $R_r[i]$ ， $N_r[i]$ 分别为 GOP 中剩余的比特数和剩余的未编码帧数。

$$\gamma = \frac{MAD_{pred}[i]}{MAD_{original}[i-1]} \quad (3-8)$$

$MAD_{pred}[i]$ ， $MAD_{original}[i-1]$ 分别为当前帧预测的 MAD 值和前一帧的实际 MAD 值。

γ 进一步限制为

$$\gamma = \begin{cases} 1, & \text{if } 0.67 < \gamma < 1.33 \\ \gamma, & \text{else} \end{cases} \quad \gamma = \max\{0, \min\{\gamma, 2\}\}; \quad (3-9)$$

上述公式中 $R_{sum}[i]$ 由 $R_1[i]$ 和 $R_2[i]$ 两部分构成，其中 $R_2[i]$ 部分主要负责根据剩余的比特数和帧数确定将编码帧应分配的比特数，从公式 3-7 可以发现在算法中修正了 JVT-G012 方案中的均分方法，而是根据编码帧的复杂度进行比特的调控，为了保证视频质量波动的平稳，通过了公式 3-9 控制了 γ 的取值； $R_1[i]$ 负责微调，它的主要目标就是根据目标缓冲级、帧率、可用带宽、实际缓存占用的情况进行该编码帧的比特分配保证目标码率能够比较精确的预估出实际码率。在低带宽情况下，实际缓冲占用和目标缓冲级的差值有时候会比较大，因此为了保证能够比较好的预估出实际比特数， β 取 0.7 这个较大值。这种合理的比特分配策略可以改善视频质量。

3.3.2 改进基本单元层 MAD 预测和 QP 值的调整

本文在基本单元层的码率控制算法中主要改进了基本单元层的 MAD 的预测，并且根据不同宏块的重要性调整了 QP 值。

由于本算法主要应用在低带宽情况下的视频会议，视频电话等应用中，在这些应用中人脸作为最主要和最重要的视频区域，运动信息也是最多的区域。因此在预测 MAD 时，不仅需要时间信息也需要空间信息。在本算法的基本单元层的 MAD 预测采用的是通过计算前一帧的相同位置的实际 MAD 和当前帧的前一个位置的基本单元的实际 MAD 值计算得到。

假设：当前基本单元是该帧的第一个基本单元，则该基本单元的 MAD 值为前一帧相同位置的 MAD 值。

$$MAD[0] = MADp[0] \quad (3-10)$$

其中 $MAD[0]$ 为当前预测基本单元的 MAD 值； $MADp[0]$ 为前一帧第一个基本单元的实际 MAD 值；

如果基本单元不为当前帧第一个基本单元，则此基本单元的 MAD 是通过前一帧的相同位置，前一帧的前一个相邻位置的实际 MAD，当前帧的前一个相邻位置的实际 MAD 计算得出：

$$MAD[i] = (MADp[i] + MADp[i-1] + MAD[i-1]) / 3 \quad (3-11)$$

如果当前基本单元为当前帧最后一个基本单元则

$$MAD[i] = (MADp[i] + MAD[i-1]) / 2 \quad (3-12)$$

在同一个基本单元层中，不同的宏块拥有不同的计算复杂度。为了取得更好的视频质量，算法对相同的基本单元层中的宏块根据不同的复杂度，调整了 QP。

$$MAD_{complex} = MAD_{bu} / MAD_{mb} \quad (3-13)$$

其中 $MAD_{complex}$ 为相同基本单元中每个宏块的复杂度； MAD_{mb} 为预测当前 MB 的 MAD 值， MAD_{bu} 为预测当前基本单元的 MAD 值

根据不同的 $MAD_{complex}$ 在宏块层对 QP 值进行如下调整：

$$\begin{aligned} \Delta QP &= 2(MAD_{complex} \leq 0.75) \\ \Delta QP &= 1(MAD_{complex} \leq 0.85) \\ \Delta QP &= 0(MAD_{complex} \leq 1.15) \\ \Delta QP &= -1(MAD_{complex} \leq 1.35) \\ \Delta QP &= -2(MAD_{complex} > 1.35) \end{aligned} \quad (3-14)$$

$$QP+ = \Delta QP$$

3.3.3 感兴趣区域的比特增强

为了保证感兴趣区域视频质量，将对属于感兴趣区域的 MB 分配更多的比特：

$$B_{MB,i,j}^- = \begin{cases} B_{MB,i,j} \times (1 + \Delta) & \text{if } MB \text{ in } ROI \\ B_{MB,i,j} \times (1 - \Delta) & \text{else} \end{cases} \quad (3-15)$$

根据实验效果， Δ 为 0.1 可以取得较好的效果。

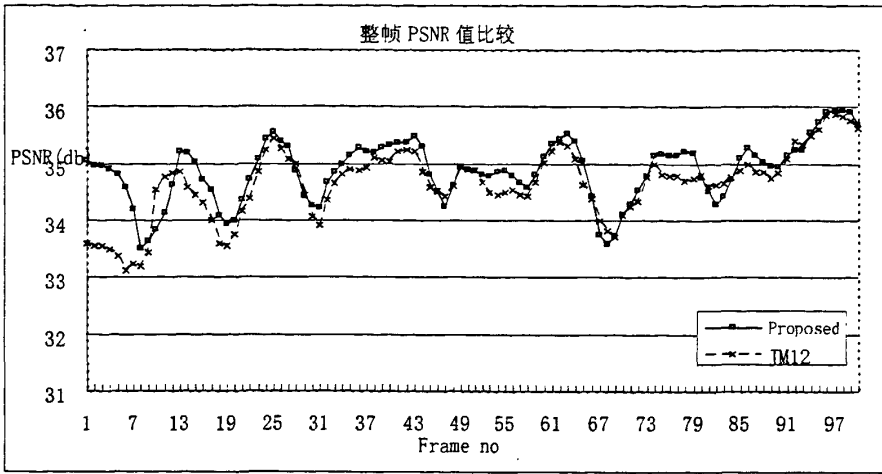
3.4 仿真结果分析

实验通过修改 H.264 的测试模型 JM9 实现了本文提出的针对人脸感兴趣区域流控算法，并与 JVT-G012 算法进行了比较。分别在多个低码率情况针对 Foreman, Claire, Miss 和 Scilent 共 5 个序列进行了测试，每个序列编码 100 帧，帧率为 30fps，比较了整帧平均 PSNR 值，ROI 区域 PSNR 值，非 ROI 区域的 PSNR 值，还有主观质量。

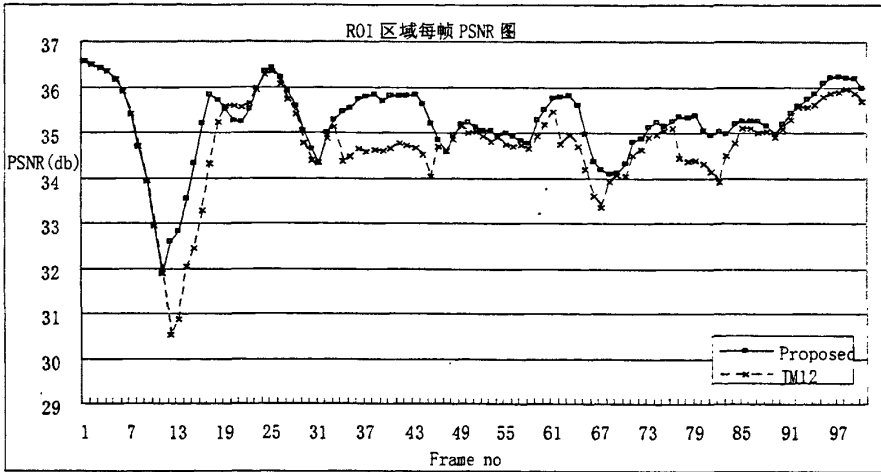
为了比较 ROI 区域的 PSNR 的波动情况, 本文还比较了 ROI 区域 PSNR 值的方差。该方差的计算公式为: $\sum (\text{样本} - \text{平均值})^2$

实验结果如图 3-1 至图 3-11 所示。

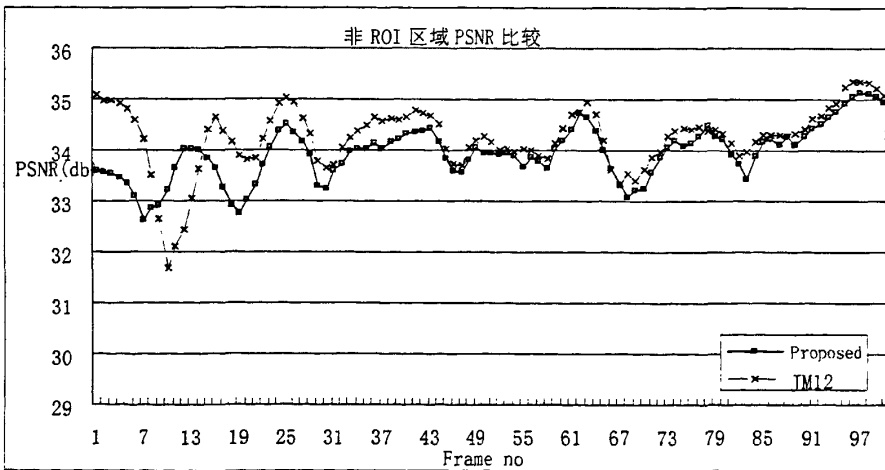
下图分布列出了 Scient, Claire, Foreman, Miss-AM 在低码率情况下, 每帧 PSNR 值比较, ROI 区域的 PSNR 比较和非 ROI 区域每帧的 PSNR 比较, 最后还列出了每个序列的主观图像, 可以从主、客观两个方面显示本算法的实验效果。



(a) 逐帧 PSNR 值

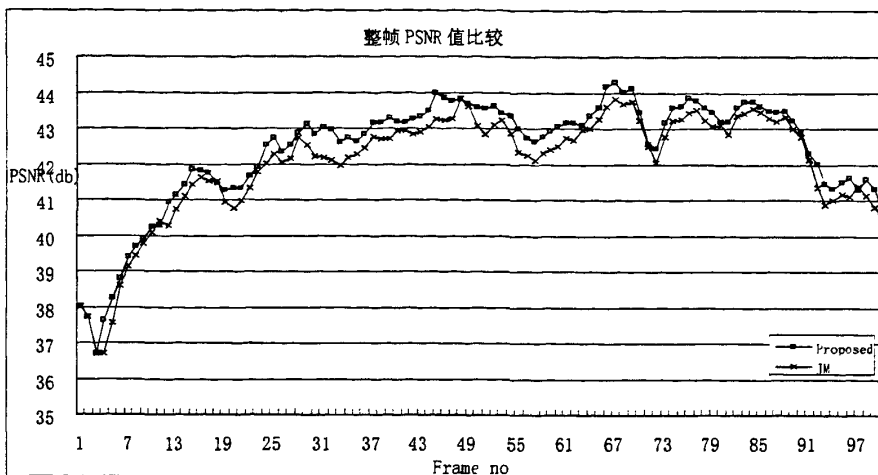


(b) 逐帧 ROI 区域 PSNR 值

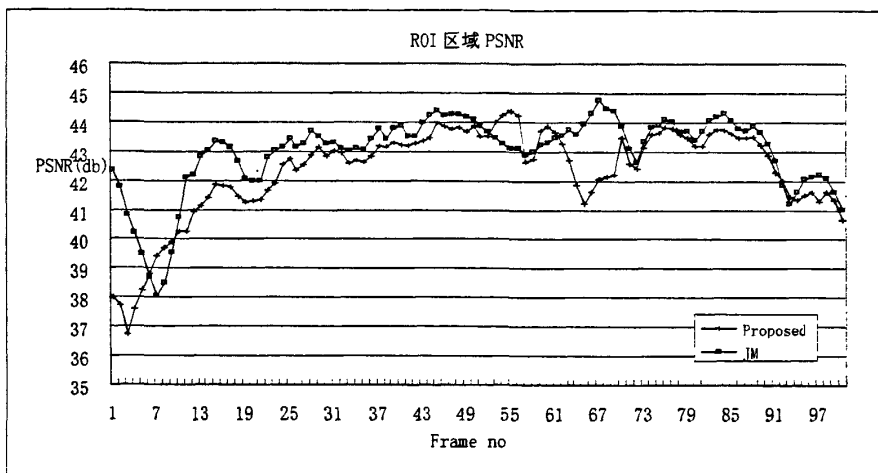


(c) 逐帧非 ROI 区域 PSNR 值

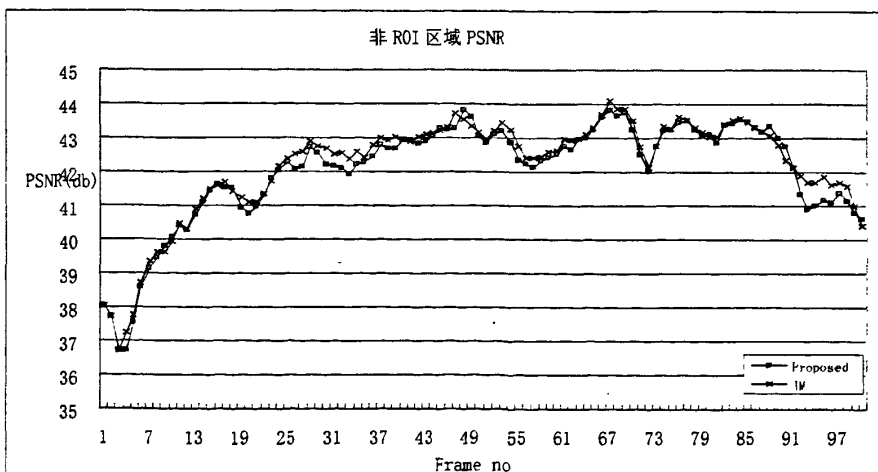
图 3-1 Sclent 序列 PSNR 比较@80k



(a) 逐帧 PSNR 值

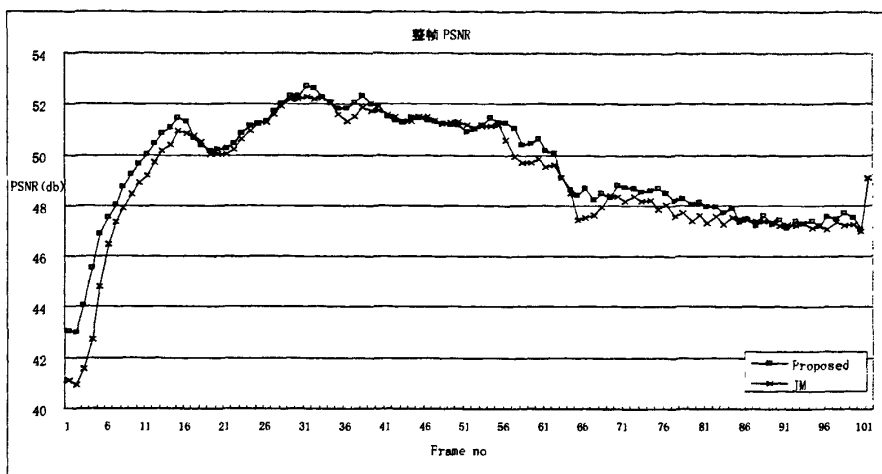


(b) 逐帧 ROI 区域 PSNR 值

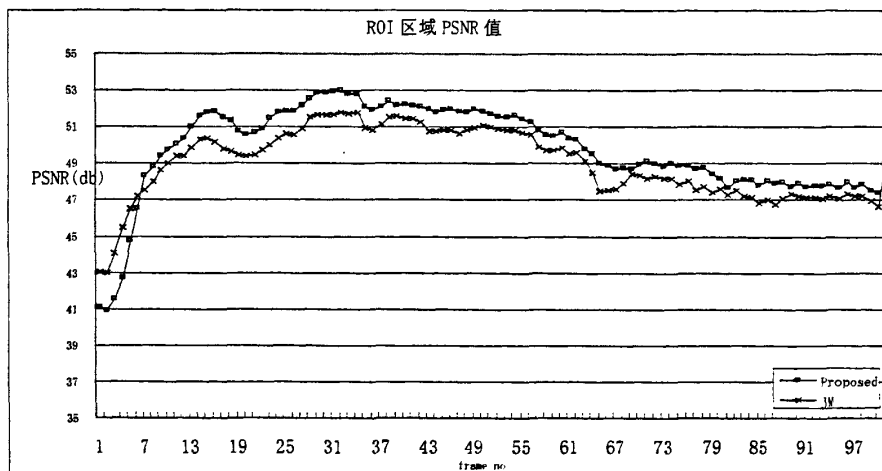


(c) 逐帧非 ROI 区域 PSNR 值

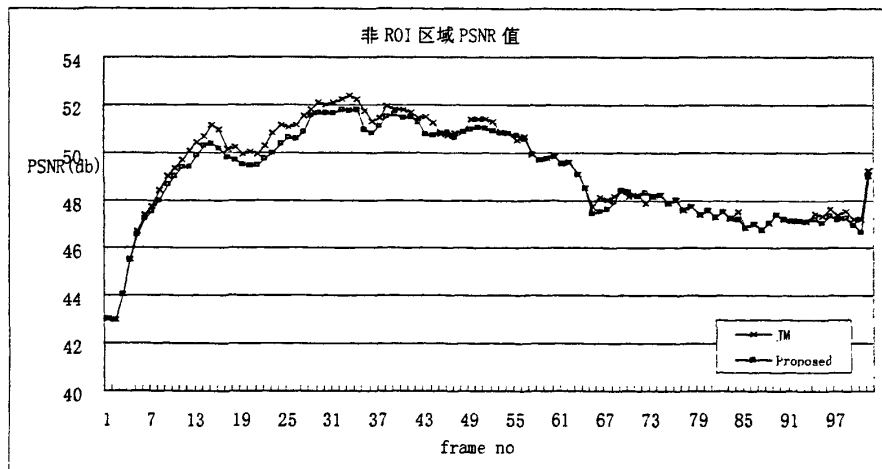
图 3-2 Claire 序列 PSNR 比较@80k



(a) 逐帧 PSNR 值

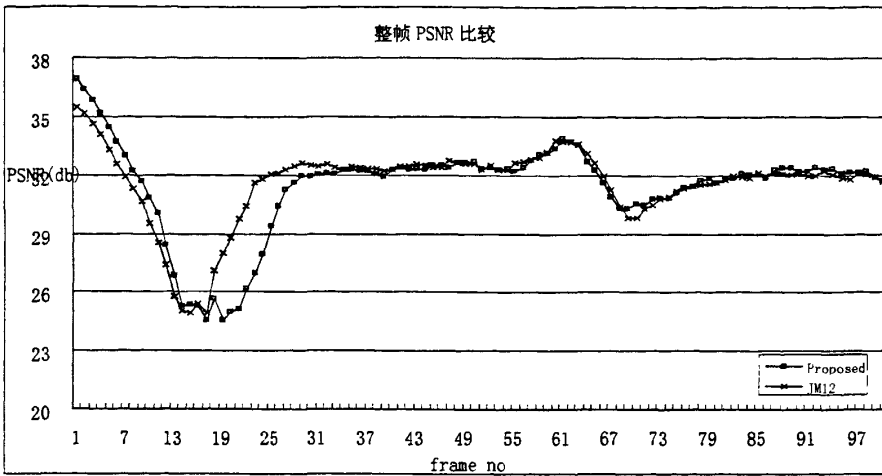


(b) 逐帧 ROI 区域 PSNR 值

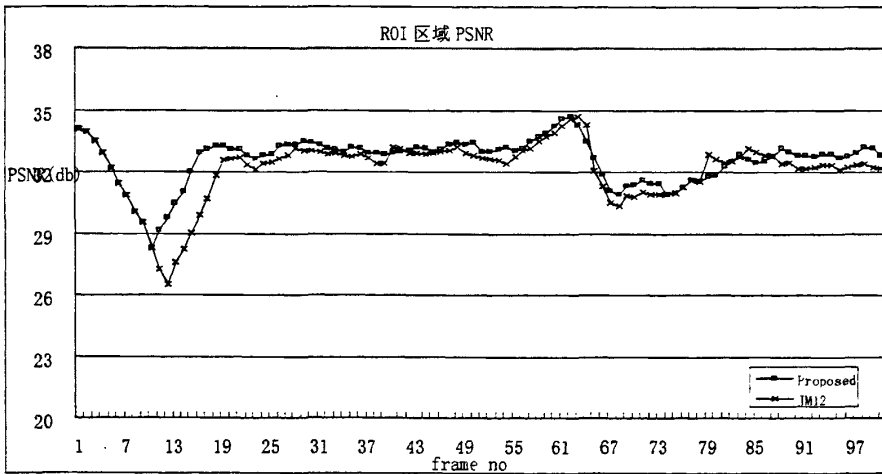


(c) 逐帧非 ROI 区域 PSNR 值

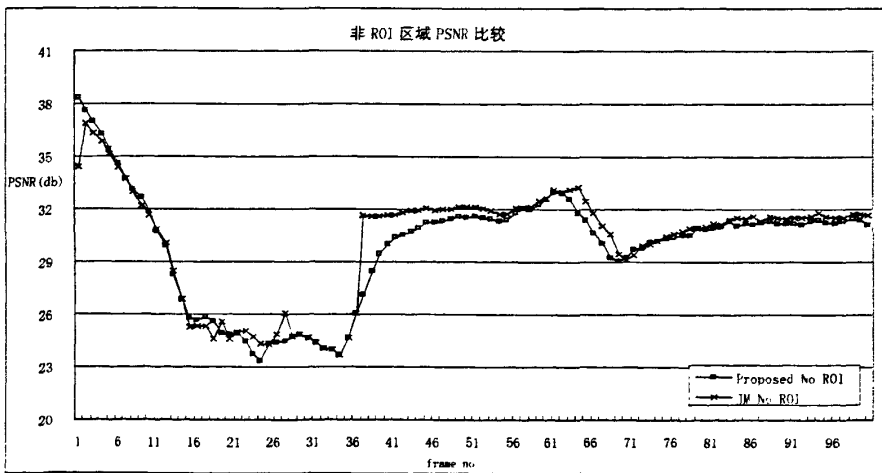
图 3-3 Miss-AM 序列



(a) 逐帧 PSNR 值



(b) 逐帧 ROI 区域 PSNR 值



(c) 逐帧非 ROI 区域 PSNR 值

图 3-4 Foreman 序列@80k



(1)

(2)

(1) Proposed

(2) G012

图 3-5 Foreman 序列主观质量比较图@80k 75th 帧



(1)Proposed (2)G012

图 3-6 Sclient 主观质量比较图 80k 46th 帧



(1) proposed (2) G012

图 3-7 Claire 序列主观质量比较图 @80k 97th 帧



(1) proposed

(2) G012

图 3-8 Foreman 序列主观质量比较图 @56k 27th 帧



(1) Proposed

(2) G012

图 3-9 Foreman 序列主观质量比较图 @56k 75th 帧

(1) Proposed

(2) G012

图 3-10 Sclinet 主观质量比较图 56k 90th 帧

(1) Proposed

(2) G012

图 3-11 Miss-AM 主观质量比较图 56k 98th 帧

从图 3-1 至图 3-4 的实验结果可以发现, 本算法在感兴趣区域 PSNR 值较之 JVT-G012 有了一定的提高。由于本文算法适用在带宽较窄的应用当中, 因此从图 3-1 至图 3-4 的曲线走势可以看出, PSNR 的提高并不是通过提高码率, 提高整帧的 PSNR 方法得到的, 而是通过牺牲一定的非 ROI 区域的 PSNR 质量得出的, 当 ROI 区域的 PSNR 有所提高的时候, 相对应的非 ROI 区域的 PSNR 会出现下降。另外, 在本文当中主要是通过 MAD 的复杂度调整宏块的 QP, 因此对于纹理比较复杂的宏块, 本文算法较之 G012 质量更好, 例如: 在 Foreman 序列, Sclinet 序列从主观质量上来看就要优于 G012, 而从图 3-8, 图 3-11 可以发现当纹理信息比较少的帧主观质量并没有太大的提高, 这也是值得以后继续研究的地方。

表 3-2 Sclent 序列 PSNR 比较

80kbps 初始 QP 30

	整帧 PSNR (db)	ROI 区域 PSNR (db)	非 ROI 区域 PSNR (db)	ROI 区域 PSNR 的方差
Proposed	34.88	34.78	33.94	62.25
G012	34.64	35.23	34.25	107.7
Proposed vs G012	+0.24	+0.55	-0.31	

表 3-3 Forman 序列 PSNR 比较

56kbps 初始 QP 28

	整帧 PSNR (db)	ROI 区域 PSNR (db)	非 ROI 区域 PSNR (db)	ROI 区域 PSNR 的方差
Proposed	31.53	32.58	29.82	125.83
G012	31.60	32.18	30.25	215.26
Proposed vs G012	-0.07	+0.4	-0.43	

表 3-4 Miss-AM 序列 PSNR 比较

80kbps 初始 QP 30

	整帧 PSNR (db)	ROI 区域 PSNR (db)	非 ROI 区域 PSNR (db)	ROI 区域 PSNR 的方差
Proposed	48.53	48.77	48.21	174.96
G012	48.38	48.47	48.31	213.20
Proposed vs G012	+0.25	+0.3	-0.1	

表 3-5 Claire 序列 PSNR 比较

56kbps 初始 QP 28

	整帧 PSNR (db)	ROI 区域 PSNR (db)	非 ROI 区域 PSNR (db)	ROI 区域 PSNR 的方差
Proposed	42.36	42.46	42.02	170.34
G012	42.03	42.89	42.16	247.34
Proposed vs G012	+0.33	+0.43	-0.16	

由表中可以看出, 本文的码率控制算法在不同的序列, 不同的初始 Qp 和不同的目标码率下都可以相应的提高 PSNR 值, 特别是在 ROI 区域的提高比较明显。如 Foreman 序列目标码率 56kb 的时候, 整帧的 PSNR 值略低于 G012, 但是算法通过牺牲非 ROI 区域的 PSNR 值, 可以保证 ROI 区域的 PSNR 提高 0.4 个 db, 满足主观要求。另外实验计算了 ROI 区域的 PSNR 的方差, 方差值越大说明 PSNR 的波动越大, 在一个良好的算法当中应该在保证算法的 PSNR 有所提高的基础上, 保证各帧视频质量的平滑性。

从实验结果可以看出本文的 ROI 区域的 PSNR 方差远小于 G012, 这说明在本文算法当中各帧的 ROI 区域的 PSNR 应该于平均值相差较小, 因此在平均值有所提高的前提下同时说明各个帧中的 ROI 区域都有所提高且整体视频质量波动较小。

3.5 本章小结

本文提出了一种基于人脸的感兴趣区域的码率控制算法。这种算法可以很好的检测到视频序列中的感兴趣区域并且在视频序列中提出了新的帧层码率分配策略, 改进了基本单元层的 MAD 预测和调整了宏块层的 QP 设定。同时对属于感兴趣区域的宏块进行了必要的码率增强。实验结果显示, 本文算法可以提高视频序列的 PSNR 值, 同时改善视频序列的主观质量。

第 4 章 自适应的 MAD 预测和帧级比特分配

4.1 引言

采用良好的码率控制算法是提高视频质量一个很好的方法。码率控制是视频编码器的重要组成部分。在视频编码过程中，输出的视频质量和码率是相互影响的，输出的码率就会越高则输出的视频质量越好，但由于受带宽或存储容量的限制，需要将视频编码器的输出总比特数控制在一定的范围内。在这种情况下，需要一个良好的码率控制算法合理的分配比特以便在满足带宽或存储容量限制的同时获得最好的视频质量，但是在原来的码率控制算法当中存在这一些不足：首先，算法在帧层的码率控制中，当图像中物体运动剧烈或者场景发生改变时，MAD 预测图像复杂度不够精确，不能很好的反应图像的复杂度，导致编码复杂度差别很大，平均分配目标比特势必会导致一帧内不同区域的解码图像差别较大。其次，在码率控制中，二次 R-Q 模型计算量庞大，不合适在低延迟通信环境传输应用。在 JVT-G012 的基础上对码率控制算法进行改进可以提高码率控制的效率。

针对 JVT-G012 有这些不足，Liu^[16]等人提出了一种新的码率控制算法，这种算法可以良好的进行 MAD 的自适应预测，但是 Liu 的这种算法仅仅应用于基本单元层。通过第二章的介绍，可以得之在 H.264 中的码率控制一般分为三个层次：GOP 层，帧层和基本单元层。在 H.264 中基本单元层的码率控制是个可以选择的功能：在配置文件中如果选择基本单元的大小为一帧时，基本单元层的码率控制就会被跳过，仅采用帧层和 GOP 层的码率控制。因此本文针对这个不足修改该了帧层的 MAD 预测和帧级比特分配。本文提出的算法吸收了 Liu^[16]算法中的优点，计算了根据前后两帧空间信息和时间信息得出的 MAD 并且根据判断条件对最后的 MAD 进行确定。本文的算法中在修正 MAD 预测模型的基础上还对帧级比特的分配进行了修改，新的帧级比特分配模型更加注重运动复杂度高的帧，对这样的帧可以自适应的分配较多的比特。实验结果显示本文算法比 Liu^[16]的算法在低码率情况下 PSNR 有所提高，目标码率有所上升，但是上升的幅度不大。

4.2 算法介绍

4.2.1 帧级自适应的 MAD 预测

在 JVT-G012 提案中，为了解决蛋鸡悖论利用前一个宏块的 MAD 值预测当前宏块的 MAD 值。

$$MAD_{cb} = a_1 \times MAD_{pb} + a_2 \quad (4-1)$$

其中 a_1 和 a_2 为预测模型的两个系数。 a_1 和 a_2 的初始值相应的设为 1 和 0。当编码每

一基本单元时它相应的进行更新。

在高速运动或者场景切换的时候，这种根据原有的时间信息进行预测的方法不仅在出现场景切换或者高速运动的时候预测不够精确，而且这种预测模型是通过已产生的实际时域 MAD 值进行修改，这种修改模型的方法使得这种模型只能预测时域部分的 MAD 改变，对空域部分的 MAD 缺乏足够的预测。要预测出更加精确的 MAD 值应该采用空间和时间信息相结合的方式进行预测。

在一些文献中已经指出，利用原始帧和前一帧的重建帧可以得到的 MAD_{rough} 与 MAD_{actual} 曲线的走势基本一致且峰值中基本一样^[52]。因此为了避免蛋鸡悖论的产生 可以用 MAD_{rough} 作为预测当前帧 MAD 时空间信息。

$$MAD_{pred.spat}[i] = C_1[i] \times MAD_{rough}[i-1] + C_2[i] \quad (4-2)$$

这里的 $MAD_{pred.spat}[i]$ 是第 i 帧的 MAD 空间信息的预测。

在时域方面仍然采用 JVT-G012 提案中 MAD 线性模型方法进行预测，在计算出时域和空域的 MAD 之后由于空域的 MAD 预测在突然发生场景切换或者是快速运动时能够体现出良好的效果，但是在纹理细节比较多的情况中时域 MAD 预测比空域 MAD 能够预测的更加准确。因此在计算出两个 MAD 以后，需要根据两者的一贯以来的精确程度进行比较确定。

$$E_{spat}[i] = \sum_{n=1}^i |MAD_{pred.spat}[n] - MAD_{actual}[n]| \quad (4-3)$$

$$E_{temp}[i] = \sum_{n=1}^i |MAD_{pred.temp}[n] - MAD_{actual}[n]|$$

当 $E_{spat}[i] > E_{temp}[i]$ 则

$$MAD_{pred.adapt}[i+1] = MAD_{pred.temp}[i+1] \quad (4-4)$$

当 $E_{spat}[i] < E_{temp}[i]$ 则

$$MAD_{pred.adapt}[i+1] = MAD_{pred.spat}[i+1] \quad (4-5)$$

4.2.2 帧级比特控制

在帧级每个帧的目标码率的分配

$$f(n_{i,j}) = \beta * \hat{f}(n_{i,j}) + (1 - \beta) * \tilde{f}(n_{i,j}) \quad (4-6)$$

β 是常数，没有 B 帧时取 0.5，其它情况下取 0.9

其中每帧的目标码率由两部分构成，一部分是 $\tilde{f}(n_{i,j})$ ，在 JVT-G012 中

$$\tilde{f}(n_{i,j}) = \frac{u(n_{i,j})}{F_r} + \gamma(Tbl(n_{i,j}) - B_c(n_{i,j})) \quad (4-7)$$

其中 γ 是常数，它的典型值是在没有 B 帧的情况下为 0.75，其它情况为 0.25。

$Tbl(n_{i,j})$, $B_c(n_{i,j})$ 分别表示的是目标缓冲级函数和实际缓冲区所占值, 目标缓冲级函数是用来计算每一个 P 帧的目标比特。

另一部分是 $\hat{f}(n_{i,j})$, 在本文中为了满足 $\hat{f}(n_{i,j})$ 能够随着帧的复杂度进行合理的比特分配。

$$\hat{f}(n_{i,j}) = (\alpha \times \frac{B_r[i]}{R_r[i]} + (1 - \alpha) \times \overline{B_{act}}[i-1]) \times \sigma \quad (4-8)$$

$$f(n_{i,j}) = \max(L_{HRD}, \min(\hat{f}(n_{i,j}), U_{HRD}))$$

其中 $B_r[i]$ 和 $R_r[i]$ 分别表示剩余的比特数和剩余的帧数, $\overline{B_{act}}[i-1]$ 表示编码前 $i-1$ 帧所需要的平均比特数, α 是一个平衡两项的因子。其中在本实验中 α 取 0.755. σ 是根据帧的复杂度大小进行比特分配的一个权重值。

$$\sigma = \frac{MAD_{pred.adapt}[i]}{MAD_{actual}[i-1]} \quad (4-9)$$

当将要编码的帧比较复杂, 则 σ 就比较大。这样就可以给该帧分配较多的比特。当然为了保证视频质量各帧之间质量波动不太大, 将 σ 进行限制。

$$\sigma = \begin{cases} 1 & 0.5 < \sigma < 1 \\ \sigma & else \end{cases} \quad (4-10)$$

最后, 所计算得出的 $f(n_{i,j})$ 需要满足 HRD 模型。所以

$$f(n_{i,j}) = \max(L_{HRD}, \min(f(n_{i,j}), U_{HRD})) \quad (4-11)$$

4.2.3 帧层编码步骤

第一步: 预测帧级 MAD, 分以下几步完成:

- 1) 按照公式 4-2 计算基于空间信息的 MAD;
- 2) 按照公式 4-3 计算基于空间信息的 MAD 与平均值的误差值;
- 3) 按照公式 4-1 计算基于时间信息的 MAD;
- 4) 按照公式 4-3 计算基于时间信息的 MAD 与平均值的误差值;
- 5) 按照公式 4-4 和 4-5 确定 MAD 值

第二步: 计算 P 帧的目标比特数:

- 1) 根据公式 2-17 计算目标缓冲区水平;
- 2) 根据目标缓冲区水平、实际缓冲区大小, 信道带宽和帧率通过公式 4-7 计算目标比特 $\tilde{f}(n_{i,j})$;

3) 通过 MAD 值计算得到权重因子 σ ;

4) 根据权重因子 σ , 剩余比特数和剩余帧数按照公式 4-8 得到目标比特的 $\hat{f}(n_{i,j})$;

5) 最终的目标比特有 $\tilde{f}(n_{i,j})$ 和 $\hat{f}(n_{i,j})$ 由公式 4-11 完成。

第三步: 计算 P 帧的 QP

第四步：根据配置文件的设置，进入基本单元层的码率控制。若在配置文件中，基本单元层的大小设置为整帧，则跳过此步。

第五步：更新二次 R-D 模型的参数，更新当前帧的剩余比特数；

第六步：完成当前帧的编码，如果 GOP 还有未编码帧转入第一步。否则完成编码。

下图是帧层码率控制算法流程图：

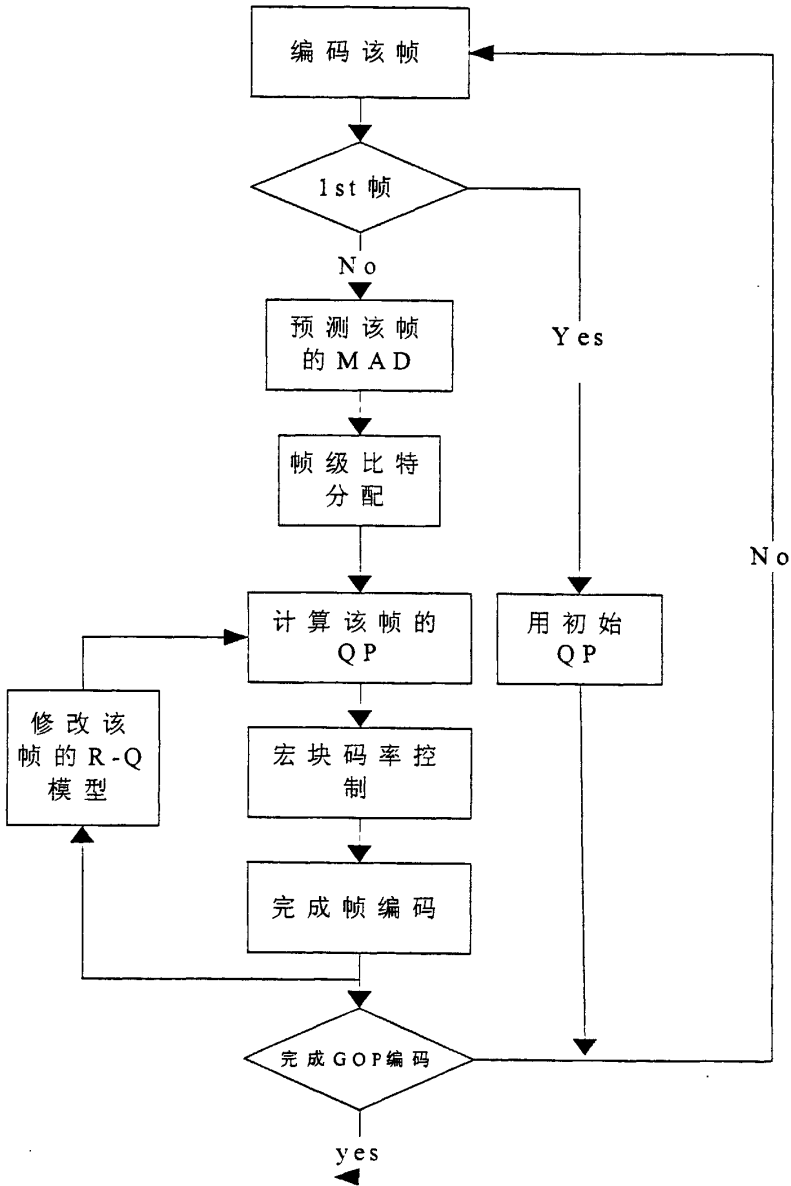
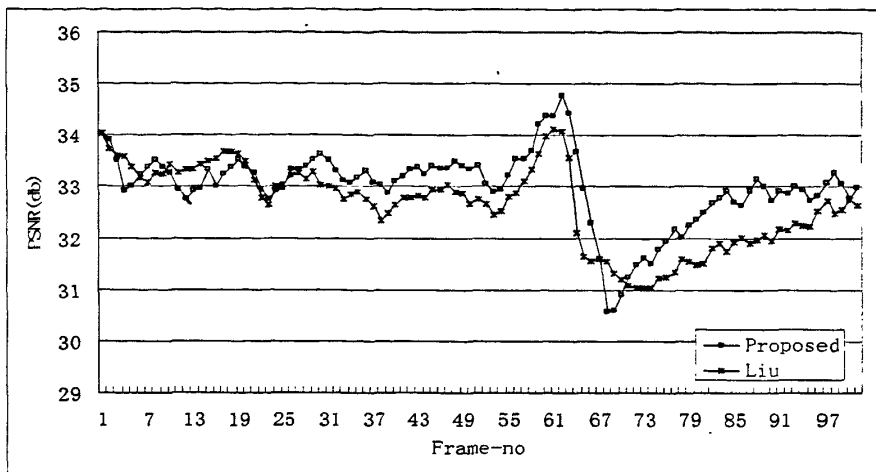


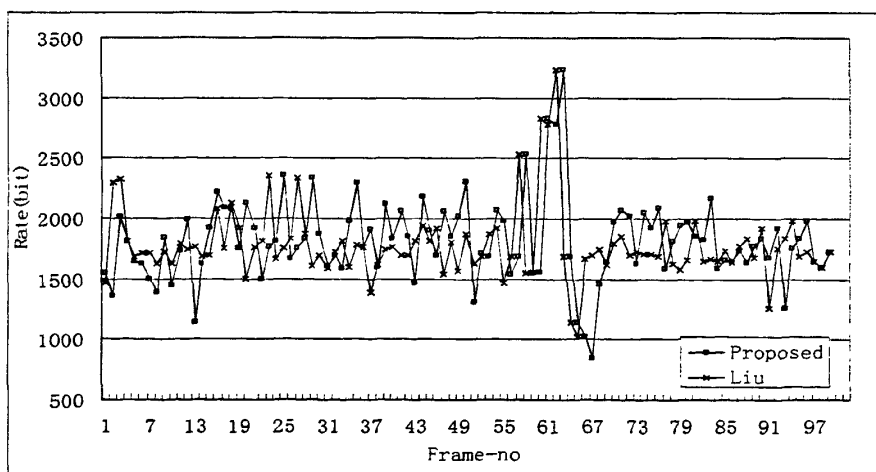
图 4-1 帧层码率控制算法流程图

4.3 实验结果

实验通过修改 H.264 的测试模型 JM9 实现了本文提出的流控算法，并与文献[16]中的算法进行了比较。在低码率(目标码率: 80Kbps/56Kbps, 初始 QP=30/32)的情况下对 Foreman, Claire, News, Miss-AM, Mother and daughter 和 Scilent 共 6 个序列进行了测试, 每个序列编码 100 帧, 帧率为 30fps, 比较平均 PSNR 值, 整帧 PSNR 波动和整帧比特波动。实验结果如图 4-2 至图 4-7 所示。

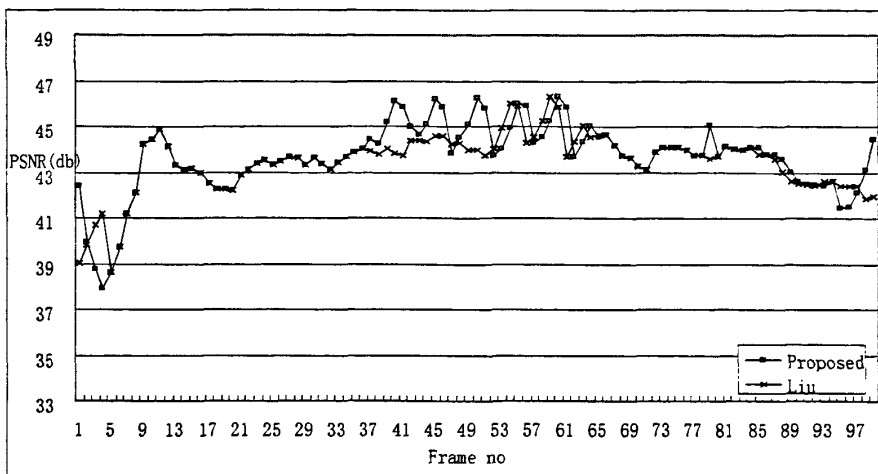


(A) 逐帧 PSNR

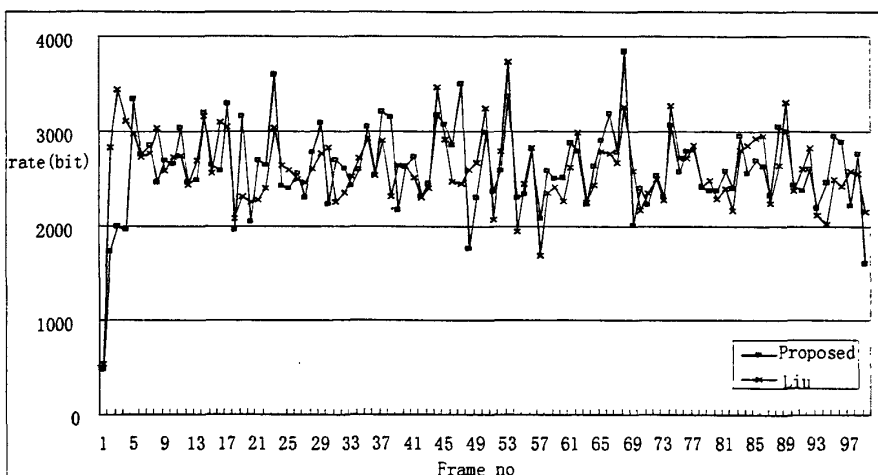


(B) 逐帧码率

图 4-2 Foreman 序列

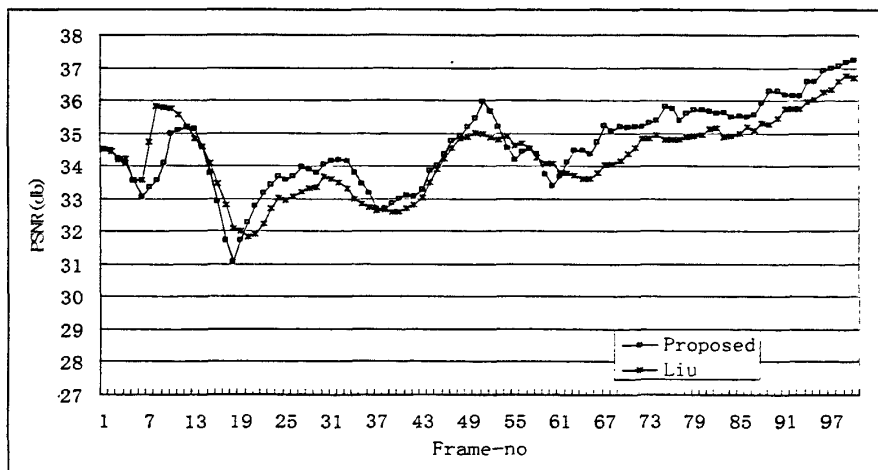


(A) 逐帧 PSNR

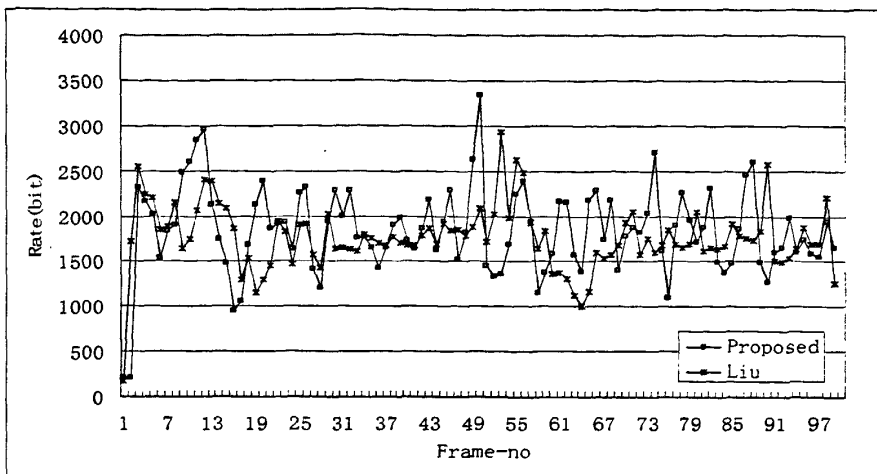


(B) 逐帧码率

图 4-3 Claire 序列

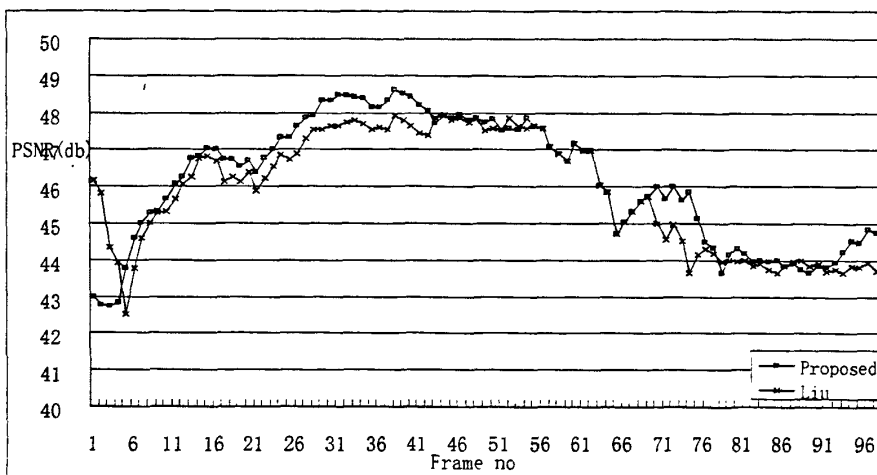


(A) 逐帧 PSNR

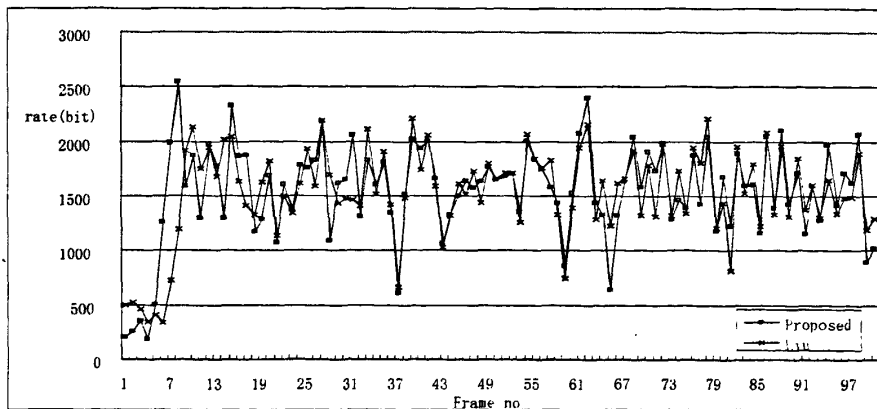


(B) 逐帧码率

图 4-4 M&d 序列

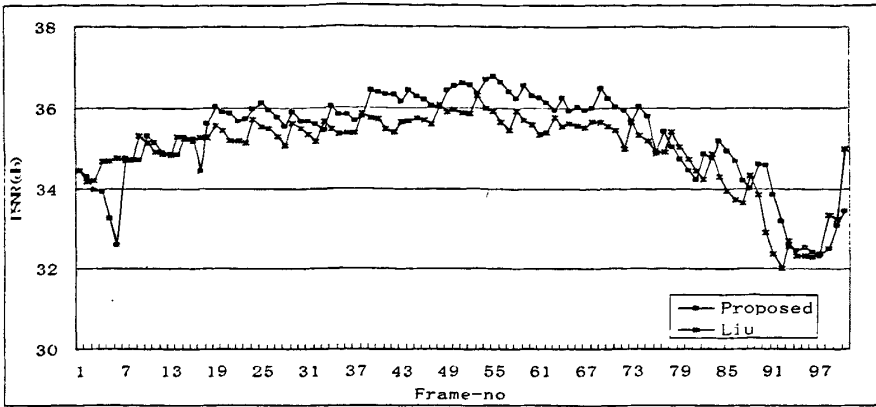


(A) 逐帧 PSNR

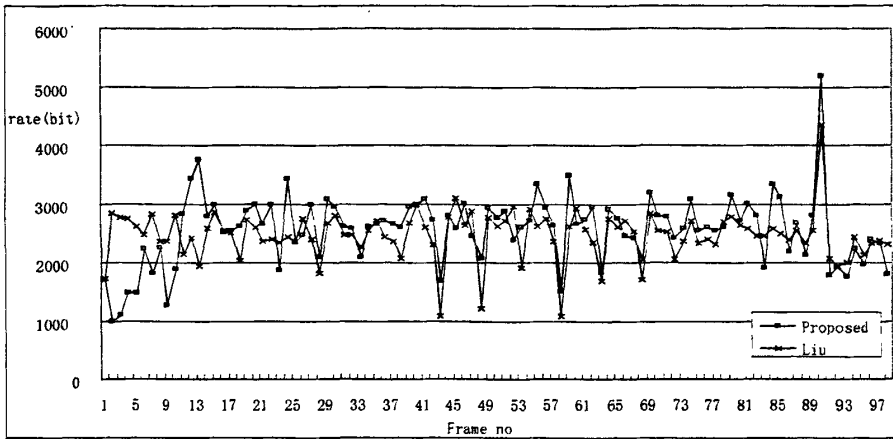


(B) 逐帧码率

图 4-5 Miss 序列

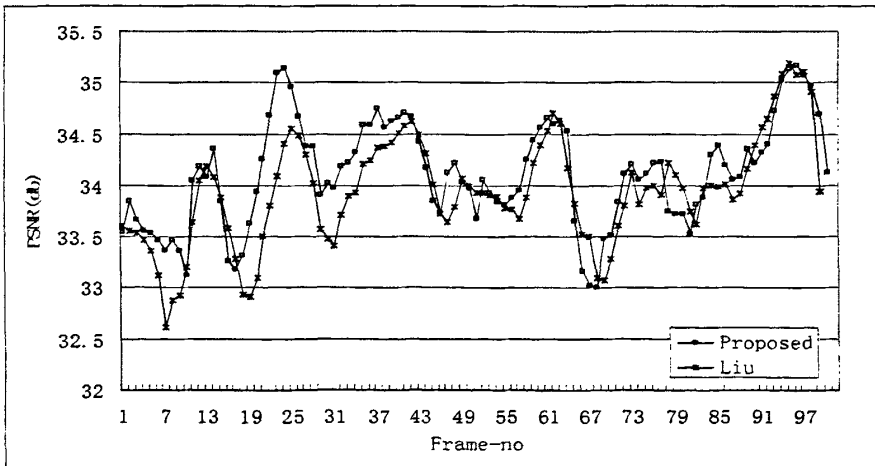


(A) 逐帧 PSNR

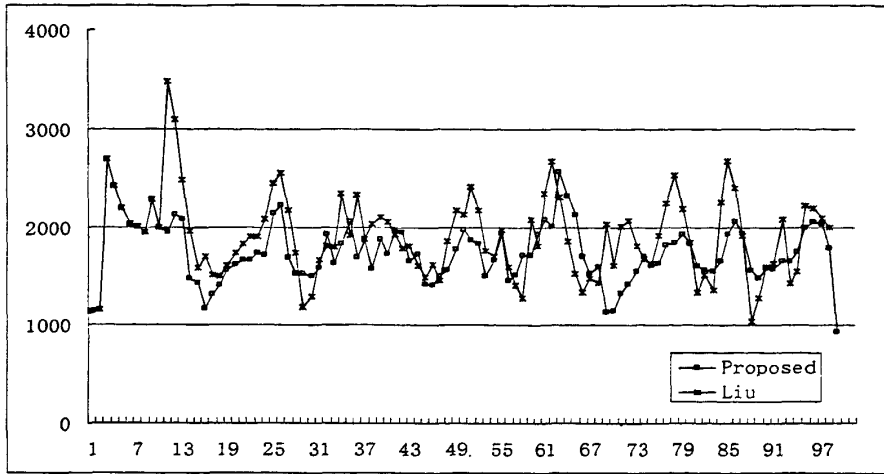


(B) 逐帧码率

图 4-6 News 序列



(A) 逐帧 PSNR



(B) 逐帧码率

图 4-7 Silent 序列

图 4-2 至图 4-7 给出低码率下本文算法与文献在部分序列, 每帧 PSNR 值比较和每帧比特波动情况。通过图示我们可以发现本文的 PSNR 值提高幅度明显, 但是波动情况较之 Liu 的算法在某些序列下波动也比较剧烈, 这是因为本文的算法在帧级比特分配时, 更加注重的是每帧的编码复杂度, 对于复杂度高的帧分配比特较多, 这样就会由于视频内容出现比特的波动, 但是这种波动并不会引起视频质量的平滑性, 从统计的结果可以看出, 这种波动一般控制在 0.25kbs 左右, 是一种允许的波动情况。

表 4-1 低码率下三种算法的 PSNR 比较(80Kbps@30fps, 初始 QP=30)

测试序列	平均 PSNR 值(db)				
	G-012	文献[16]	proposed	Proposed	Proposed
				Vs	vs
				G-012	文献[16]
Foreman	34.12	34.35	34.45	0.33	0.1
M&D	36.09	36.26	36.63	0.54	0.37
Claire	43.31	43.47	43.65	0.34	0.28
News	37.35	37.56	37.80	0.45	0.34
Miss	45.50	45.60	45.82	0.32	0.22
Scilent	36.16	36.30	36.67	0.51	0.37

表 4-2 低码率下三种算法的 Rate 比较(80Kbps@30fps, 初始 QP=30)

测试序列	平均 Rate 值(kbit/s)				
	G-012	文献[16]	proposed	Proposed	Proposed
				Vs	vs
				G-012	文献[16]
Foreman	80.42	80.53	80.64	0.22	0.11
M&D	80.48	80.74	80.84	0.36	0.1
Claire	80.41	80.55	80.77	0.36	0.22
News	80.45	80.60	80.74	0.29	0.14

Miss	80.64	80.75	80.84	0.2	0.09
Scilent	80.72	80.77	80.97	0.25	0.2

表 4-3 低码率下三种算法的 PSNR 比较(56Kbps@30fps,初始 QP=32)

测试序列	平均 PSNR 值(db)				
	G-012	文献[16]	proposed	Proposed	Proposed
				Vs G-012	vs 文献[16]
Foreman	32.64	32.7	32.84	0.2	0.14
M&D	34.38	34.48	34.77	0.39	0.29
Claire	41.8	42.03	42.32	0.35	0.19
News	34.98	35.16	35.43	0.45	0.27
Scilent	33.98	34.23	34.48	0.5	0.25

表 4-4 低码率下三种算法的 Rate 比较(56Kbps@30fps,初始 QP=30)

测试序列	平均 Rate 值(kbit/s)				
	G-012	文献[16]	proposed	Proposed	Proposed
				Vs G-012	vs 文献[16]
Foreman	56.25	56.47	56.60	+0.45	+0.13
M&D	56.45	56.60	56.80	+0.35	+0.15
Claire	56.37	56.42	56.58	+0.21	+0.16
News	56.37	56.57	56.69	+0.32	+0.12
Scilent	56.58	56.82	56.94	+0.36	+0.12

由表中可以看出，本文的码率控制算法在低码率虽然有了一定的波动，但是这种码率的波动不到目标码率的 1%。因此这种波动尚不会影响视频的质量。在实验结果中 6 个序列的平均 PSNR 值均比 JVT-G012 有所提高，与文献[16]中的算法相比也均有提高。其中 News 序列的 PSNR 值在低码率下比 JVT-G012 高 0.45db, Scilent 序列的 PSNR 值在低码率下比参考文献[16]提高 0.51db, 这种 PSNR 的提高是由于在算法中改进了帧级的 MAD 的预测,更加精确的预测可以改善帧层的比特分配从而可以有效提高 PSNR。

4.4 本章小结

分析了 JVT-G012 在帧层 MAD 预测和帧层码率分配存在的不足，并且研究了 Liu 的算法。应用了该算法在帧级 MAD 预测并且提出了一种新的码率分配的方法。这种方法基于自适应的 MAD 预测，对复杂度较高，纹理信息复杂的帧将分配更多的比特，保证在低码率情况下视频质量有一定的提高。从实验结果可以看出，与 Liu 算法相比，本文的改进算法在 PSNR 中有明显的提高。

总结与展望

1. 主要研究内容和成果

由于 H.264/AVC 标准具有较高的编码效率和良好的压缩性能因此受到了诸多科研机构及多媒体工业界的高度重视, 是最为活跃的研究热点。码率控制是数字视频编码中的关键技术, 它决定了对各个帧和宏块的比特分配和 QP 设定, 特别是在低码率情况下, 码率控制对于视频质量有举足轻重的作用。因此对码率控制进行深入研究具有十分重要的理论与实际意义。本文主要研究了在低带宽情况下针对人脸感兴趣区域的码率控制。

(1) 本文介绍了码率控制技术原理并介绍了经典码率控制算法。在此基础上详细介绍了 H.264/AVC 的码率控制算法: JVT-G012 提案, 分析了 JVT-G012 的算法流程。

(2) 本文在一种快速准确的视频人脸检测算法对人脸这个感兴趣区域进行提取的基础上, 改进了一种基于人脸感兴趣区域的码率控制算法, 该算法在帧级提出了新的比特分配策略, 改进基本单元层中 MAD 预测算法, 对不同宏块的 QP 值进行了调整同时对感兴趣区域的比特数进行增强, 实验结果表明, 这种新提出的算法提升了感兴趣区域的 PSNR 和感兴趣区域的主观质量。

(3) 分析了 H.264 中帧级 MAD 线性模型中的不足, 并提出了改进策略。这种改进策略在预测 MAD 时, 改变了原来 MAD 只利用时域信息进行预测的方法, 在利用时域信息进行预测的同时也利用了空域的信息进行预测。本文也改进了原来 JVT-G012 方案中对帧级比特分配策略, 这种新的分配方法结合了缓冲区的充溢度, 能够根据编码的复杂度自适应的分配了比特

2. 存在的不足和展望

在视频研究领域, 码率控制是其中的关键环节. 特别是现在随着 3G 网络的广泛普及, 更需要利用码率控制提高良好的在低带宽情况下的视频质量。文本提出了在低带宽情况下的基于人脸区域的码率控制算法和自适应 MAD 预测帧级比特分配的算法虽然于原始的算法相比较有了改进, 但是由于作者水平有限, 所提出的算法还是有很多地方可以进行改进和研究的:

(1) 本文中, 所提出的自适应的 MAD 预测和帧级比特分配算法中码率波动较之 JVT-G012 相比在某些序列下比较大, 需要进行研究使其能在提高 PSNR 的同时波动能够更加平稳。

(2) 本文中, 对于感兴趣区域的码率控制研究也可以进行深化。例如在本文当中是将人脸宏块根据 MADcomplex 强制的将 QP 值进行调整, 这样的方法并不能很好的应用于所有的视频序列, 在实际应用中更缺乏灵活性。因此这个问题值得进一步研究。

致 谢

首先感谢我的导师彭强教授，感谢导师三年来的悉心指导和精心培养，在老师的严格要求和悉心指导，本论文才得以顺利完成。老师渊博的知识、严谨的治学态度、永远值得我学习。老师谦逊的品格、平易近人的态度也使我深受教诲，是我以后工作和做人的楷模。

同时，感谢虚拟现实与多媒体实验室所有老师、同学。在大家共同努力下，营造了一个友好，团结，协作的求学环境，让我得到很多启发。正是有这样积极、向上的研究环境，让我能够顺利并愉快地完成论文。特别感谢张庆明博士，是他耐心地辅导使我进入视频编码研究领域，并且无私地与我分享各种经验和体会，时刻激励着我克服困难、奋勇向前。在此向张庆明博士表示由衷的感谢。

最后，我要感谢我的父母。感谢他们的养育之恩，以及多年来给我的关爱。感谢他们为我创造的条件和对我求学道路的无条件的支持。感谢他们的理解和支持使我能够选择自己的道路，实现自己的理想。

朱柳

2010年5月于交大

参考文献

- [1] 毕厚杰. 新一代视频压缩编码标准——H. 264/AVC[M]. 人民邮电出版社, 2005. 5
- [2] 陈杰. H. 264 视频编解码传输算法平台及码率控制研究[D]. 硕士学位论文. 上海: 上海交通大学, 2007:8
- [3] 王瑞昕. 感兴趣区域图像编码研究[D]. 硕士学位论文. 武汉: 武汉科技大学, 2008:12
- [4] 蔡克恒. H. 264 码率控制算法的研究与优化[D]. 硕士学位论文. 成都: 电子科技大学, 2008:9
- [5] 闫小涛. 低比特率下帧层和宏块层新型码率控制算法[D]. 硕士学位论文. 成都: 西南交通大学. 2008:10
- [6] Ma Sw G W L Y. etal. Proposed draft description of Rate Control on JVT standard [Z]. Awaji: 2002.
- [7] Celettol M M. A Rate Control Algorithm for H.264 Encoder [Z].
- [8] He Yang Q W. A Novel Maeorblock Layer RateControl for H.264/AVT [C]. 2007.
- [9] 李俊杰, 吕维等一种基于纹理特征的码率控制算法[J]. 计算机工程与应用, 2007, 43(16): 71 — 74.
- [10] 李岚. 人脸检测方法研究. 科技信息[J]. 2009.28(606):742-744
- [11] Shaou yi Chien, Yu-enHuang, etal. Predictive Watershed: A Fast Watershed Algorithm for Video Segmentation[J]. IEEE Trans. Circuits and Systems for Video Technology. 2003. 13(5):
- [12] 黄莉, 付志中. 基于 H.264TM5 码率控制算法的改进. 计算机应用[J]. 2007, 27(3): 749 — 751
- [13] Chun-Hung Lin, Ja-Ling Wu. Content-based rate control scheme for very low bit-rate videocoding[J]. IEEE Transactions on Consumer Electronics, 1997: 123-133
- [14] Supavadee Aramvith, Hatairat Kortrakulkij, Datchakorn Tancharoen, et al. Joint source-channel coding using simplified block-based segmentation and content-based rate-control for wireless video transport[A]. Proceedings of the International Conference on Information Technology: Coding and Computing. Las Vegas: IEEE Computer Science Society, 2002: 71-76
- [15] Chia-Wen Lin, Yao-Jen Chang, Yung-Chang Chen. A low-complexity face-assisted coding scheme for low bit-rate video telephony[J]. IEICE Trans. Information and systems, 2003: 101-108

-
- [16]Yag Liu; Yeng Chai Soh; Region-of- Interest Conversational for Video Communication of H.264/AVC[J], Circuits and Systems for Video Technology, 2008:10(1) 134 - 139
- [17]Yoon K, Jae-Young P, Hye-Soo K, et al. Efficient real-time frame layer rate control technique for low bit rate video over WLAN[J]. Consumer Electronics, IEEE Transactions on. 2003, 49(3): 621-628.
- [18]Jing Y, Xiangzhong F, Hongkai X. A joint rate control scheme for H.264 encoding of multiple video sequences [J]. Consumer Electronics, IEEE Transactions on. 2005, 51(2): 617-623.
- [19]Myoung-Jin K, Min-Cheol H. Adaptive Rate Control Scheme in Frame-layer for Real-time H.264[C]. 2009.
- [20]Yali L, Xiaolan L, Gomila C, et al. A novel fine rate control algorithm with adaptive rounding offset[C]. 2008.
- [21]Jianpeng D, Nam L. A model parameter and MAD prediction scheme for h.264 macroblock layer rate control[C]. 2008.
- [22]王怡平.H.264 的码率控制算法研究[D] 硕士学位论文黑龙江：哈尔滨工业大学，2007
- [23]李慧然.H.264 视频编码标准码率控制算法的优化[D]硕士学位论文成都：西南交通大学 2008
- [24]SiweiMa,WenGao,Yai l Lu,HanqingLu.pronosed draft description of rate control on JVT stanaara[C].JVT-F086.6th meeting, Awaji.Decemlber, 2002.
- [25]Yuanwu,LinShouxun,ZhangYongdong,etal.Optimum Bit Alloecation and Rate Control for H.264/AVC[C].JVT-0016.15thMeeting:Busan,KR.APril, 2005
- [26]ZhengguoLi,FengPan,KengPangLim,etal.Adaptive Basie UnitLayer Rate Control for JVTG012.7thMeeting[C].Pattayall,Thailand.March, 2003
- [27]Pronosed draft of adaptive rate control[C].JVT-H017.8th Meet:Geneva,20-26.May 2003
- [28]ISO/IEC JTC1/SC29/WG11/N0400,MPEG-2 Test Model 5[S].1993
- [29]Vetro,H.Sun and Y.Wang.MPEG-4 rate control for multiple video objects[C] IEEE Trans.Circuit Syst.Video Technology,9:186~199,1999.
- [30]Hung-Ju Lee,Tihao Chiang,Ya-Qin Zhang.Scalable Rate Control for MPEG-Video[C].IEEE. TRANSACTIONS ON CIRCUITS AND SYSTEMS FORVIDEO TECHNOLOGY,VOL.10,NO.6,SEPTEMBER 2000,878~894.
- [31]CCITT SGXV,Description of Ref.Model 8(RM8) [M].,June,1989
-

-
- [32] Ma Siwei, Wen Gao. Rate Control For JVT Video Coding Scheme With HRD Considerations[C]. IEEE ICIP 2005.
- [33] Doulamis, N., et al., Low bit — rate coding of image sequences using adaptive regions of interest[J]. IEEE Transactions on Circuit and Systems for Video Technology, Vol. 8, No. 8, 1998: P228
- [34] Kim, C., dHwang, J-N., Fast and automatic video object segmentation and tracking of content based applications[J]. IEEE Transactions on Circuit and Systems for Video Technology, Vol. 12, No. 2, 2002: P122
- [35] Wang, Bovik, A. C. and Lu, L., Wavelet-based foveated image quality Measurement of region of interest image coding[C]. IEEE Proceedings of 2001 International Conference on Image Processing, 2001, Vol. 2: P. 89 — 92
- [36] Zhang, L-B., and Wang, K., New regions of interest image coding and its applications for sensing image[C]. IEEE Proceedings of ICMMT 4th International Conference on Microwave and Millimeter Wave Technology, 2004.: P952 — 956
- [37] Sanchez, V., Basu, A., and Mandal M.K. Prioritized region of interest coding in JPEG2000[J]. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 14, No. 9, 2004: P1149-1155
- [38] Doulamis, N. et al., Low bit-rate coding of image sequences using adaptive regions of interest[J]. IEEE Transactions on Circuits and Systems of Video Technology, Vol. 8, No. 8, 1998.: P928 — 934
- [39] Lin, T., and Rao, K.R., Region of interest based H.263 compatible codec and its rate control of low bitrate video conferencing[C]. Proceedings of 2005 International Symposium on Intelligent Signal Processing and Communication Systems, 2005.: P249-252
- [40] 刘志新, 雷菁, 雍玲. 卫星信道中 APSK 星座的优化设计[J], 上海航天, 2006.5.
- [41] Sung K., Poggio “ T. Example-based learning for view based human face detection[J]. IEEE Trans Pattern Analysis and Machine Intelligence, 1998, 20(1): 39-51.
- [42] D. Valentin, H. Abdi, A.J. O’oole, G.W. Cottrell, Connectionist models of face processing: a survey[J]. Pattern Recognition 27(1994) 65-77.
- [43] Rama Chellappa, Charles L. Wilson and Sadd Sirohey, Human and machine recognition of faces: a survey [C]. Proceedings of the IEEE, VOL. 83, NO. 5, MAY 1995, pp. 705-740.
- [44] J. Daugman, Face and gesture recognition: overview[C]. IEEE Trans. PAMI, 19(1997) 675-676.
- [45] Maxim A. Drudin, On internal representations face recognition systems[J]. Pattern
-

Reconition, 33(2000), pp.1161 — 1177.

[46]Y. Kara, K. Kobayashi, A basic study on human face recognition[J].Frontiers of Pattern Recognition, Academic Press, New York, 1972, pp.265-289.

[47]Craw, H. Ellis, J.R. Lishman, Automatic extraction of face features [J]. Pattern recogniton Lett. 5(1987) 183 — 187.

[48]T. Poggio, F. Girosi, Networks for approximation and learning [J]. Proc. IEEE 78(1990) 1481-1497.

[49].R. Brunelh, T. Poggio, Face recognition: feature versus templates[J]. IEEE Trans. PAMI15(1993) 1042 — 1052.

[50]Cai J,Goshtasby A.Detection human faces in color images[J].Image and Vision Computing,1999,18(1):63-75

[51]Ruan Ruolin; Hu Ruimin; Xiao Xuqing.A Rate Control Scheme Based on MAD Weighted ModelforH.264/AVC[A].Wireless Communications, Networking and Mobile Computing, 2007. Page(s):2903 - 2906

[52] Yag Liu, Zhengu G. Li,A Novel Rate Control Scheme for Video Communication of H.264AVC[C]. IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 17, NO. 1, JANUARY 2007 Page(s):68 - 78

攻读硕士学位期间发表的论文

- [1] 朱柳.基于人脸感兴趣区域的码率控制算法研究 2009.11 IECT (已发表)
-