

基于 IPP 的 MPEG-4 编解码研究与实现

刘彦, 李仁发, 徐成

(湖南大学计算机与通信学院, 长沙 410082)

摘要:研究了 Intel 提供的用于多媒体处理的集成性能原件 IPP 函数库以及 MPEG-4 视频压缩编解码技术, 提出了一种利用 IPP 函数进行 MPEG-4 视频压缩编解码方法, 并在 Intel PXA255 上成功地实现了无线网络环境下的 MPEG-4 视频图像实时传输, 为在嵌入式系统中进行多媒体实时处理提供了一种新方法。实验结果证明了该方法的有效性。

关键词: 嵌入式系统; MPEG-4 编解码; 多媒体实时处理

Study and Implementation of MPEG-4 CODEC Based on IPP

LIU Yan, LI Ren-fa, XU Cheng

(College of Computer and Communication, Hunan University, Changsha 410082)

【Abstract】 This paper studies the Integrated Performance Primitives(IPP) library for multimedia processing and the MPEG-4 CODEC technology, proposes a method of MPEG-4 CODEC using IPP library, in the background of real-time video processing in embedded system. A MPEG-4 video transmission system based on Intel PXA255 in the wireless network environment is designed and implemented. It is a valuable solution to real-time multimedia processing in embedded system. Experimental results show that the method is effective.

当今, 嵌入式系统需要对多媒体视频进行实时处理。嵌入式技术以其计算能力与多媒体处理能力使人们在移动设备上实现实时图像传输, 实现移动的视频会议、远程监控、远程教学和远程医疗等。多媒体信息中绝大部分是音视频数据, 数字化的数据量相当庞大, 只有采用先进的压缩编码技术才能提高通信线路(特别是移动通信)的传输效率, 实现无线移动多媒体通信系统。具有网络通信功能的嵌入式视频编解码器的研究便成为设计多媒体通信终端系统中的核心内容。MPEG-4 是新一代基于内容的多媒体数据压缩编码国际标准, 它利用很窄的带宽, 通过帧重建技术压缩和传输数据, 以最少的数据获得最佳的图像质量。基于 Intel Xscale 处理器的 IPP 函数库为 MPEG-4 编码提供了专门的优化函数, 提高了编解码器在嵌入式平台上的执行效率, 并能获得良好的视频回放质量。本文针对嵌入式系统的特点, 介绍了一种利用 Intel Xscale 处理器 PXA255 以及 Intel IPP 函数库进行 MPEG-4 视频编解码的方法, 可用于嵌入式视频监控、流媒体实时传输等 [1-2]。

1 MPEG-4 视频编解码

MPEG-4 是新一代基于内容的多媒体数据压缩编码国际标准, 并第 1 次提出了基于对象的视频编码新概念。在 MPEG-4 标准中, 一个 MPEG-4 的视觉场景通常由一个或者多个视频对象(VO)组成。每个视频对象(VO)在时域、空间域、形状、运动和纹理上都有着不同的特征。MPEG-4 在编码过程中针对不同 VO 采用不同的编码策略。这种基于对象的视频编码方式不仅克服了第 1 代视频编码中高压率率编码所产生的方块效应, 而且使用户可与场景交互; 既提高了压缩比, 又实现了基于内容的交互, 为视频编码提供广阔的发展空间。

在图 1 中, MPEG-4 位流对视觉场景定义一个层次结构 [3]。描述视觉场景的术语有: (1) 视频对象序列(Visual Object

Sequence, VOS)。 (2) 视频对象(Video Object, VO)。 (3) 视频对象层(Video Object Layer, VOL)。 (4) 视频对象平面(Video Object Plane, VOP)。 (5) 视频对象平面组(group of video object planes)。

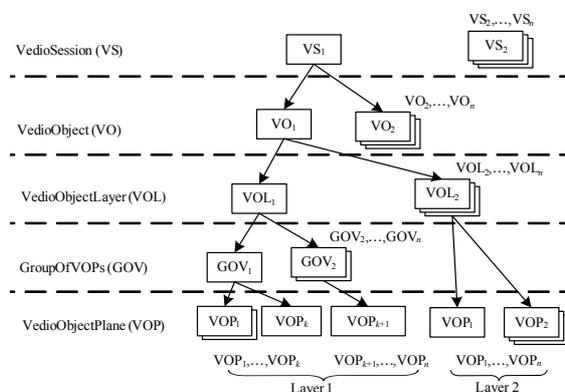


图 1 MPEG-4 编码层次

Sequence, VOS)。 (2) 视频对象(Video Object, VO)。 (3) 视频对象层(Video Object Layer, VOL)。 (4) 视频对象平面(Video Object Plane, VOP)。 (5) 视频对象平面组(group of video object planes)。

传统的矩形图在 MPEG-4 中被看作是 VO 的一种特例。传统矩形画面的一帧, 在 MPEG-4 中即为一个 VOP。这体现了传统编码与基于内容编码在 MPEG-4 中的统一。

在 MPEG-4 标准的简单类(simple profile)中, 定义了两种 VOP 编码模式: 帧内(intra-frame)编码模式(I-VOP)和帧间(inter-frame)预测编码模式(P-VOP)。I-VOP 独立于其他各帧, 只采用本帧内部信息进行编码, 称为帧内编码。而 P-VOP 是

基金项目: 国家自然科学基金资助项目“一类复杂环境下的无线传感器网络定位算法研究”(60673061)

作者简介: 刘彦(1979-), 男, 博士研究生, 主研方向: 系统结构, 嵌入式系统设计, 可重构计算; 李仁发、徐成, 教授、博士

收稿日期: 2007-03-25 **E-mail:** qmttz@163.com

通过对前一帧或几帧进行运动估计(motion estimate)来编码,称为帧间预测编码。

类似于其他视频编码标准, MPEG-4 采用运动预测(motion estimate)和运动补偿(motion compensation)技术来去除图像信息中的时间冗余成分。在 MPEG-4 标准定义的简单类(simple profile)中, VOP 的 2 种编码模式可以是基于 16×16 像素宏块的, 也可以是基于 8×8 像素块的。

纹理编码的对象可以是帧内编码模式的 I-VOP, 也可以是帧间编码模式 P-VOP 运动补偿后的预测误差, 编码方法仍采用基于 8×8 像素块的 DCT 变换。帧内编码模式中还对 DCT 变换的 DC 及 AC 因子进行有效的预测。在帧间编码模式中, 为了对 P-VOP 运动补偿后的预测误差进行编码, 变换之后的 DCT 因子还需过量化(采用单一量化因子或量化矩阵)、扫描及变长编码。

2 PXA255 和 Intel IPP 库

PXA255 是 Intel 公司推出的新一代基于 ARM 核的嵌入式微处理器^[4], 它采用了高性能、低功耗的 XScale 架构并集成了各种丰富的外围接口; 同时采用了 Intel 媒体处理技术, 包含 40 位累加器和 16 位 SIMD 以增强音频/视频编解码性能。另外, Intel 公司专门为 Linux 提供了用于 PXA255 平台的多媒体与无线通信应用的优化函数库, 该处理器具有强大的多媒体处理能力, 适合用来对实时图像进行压缩。

IPP(Integrated Performance Primitives)是 Intel 公司推出的一个跨平台、高性能多媒体函数库。它提供的函数库可广泛应用于多媒体领域, 包括视频编解码(如 H.263, MPEG-4)、音频编解码、图像压缩(如 JPEG, JPGE-2000)、图像处理、信号处理、音频压缩(如 G.723, GSM ARM*)、计算机视觉等。同时, IPP 针对基于 Intel Xscale 技术的 Intel PXA 处理器进行了优化; 使用 IPP 进行多媒体程序开发能够尽可能地发挥处理器的性能, 以达到图像处理的实时性。

3 基于 IPP 视频编解码设计与实现

针对 PXA255 的性能结构特点, 可以利用 Intel 提供的 IPP^[5] 函数, 设计了一个基于 MPEG-4 的编解码系统, 其系统体系结构如图 2 所示。



图 2 编解码系统结构

对于 MPEG-4 编码中的基本操作, IPP 为此提供了一系列高效的函数来实现, MPEG-4 编码 IPP 函数如下:

- (1) 色度空间转换(RGB to YUV)
ippiRGBToYUV420_8u_C3P3
- (2) 运动补偿(Motion Estimate)
ippiSAD16x16_8u32s ippiSAD8x8_8u32s_C1R
- (3) 离散余弦变换(DCT)
ippiDCT8x8Fwd_8u16s_C1R ippiDCT8x8Fwd_16s_C11
- (4) 量化
ippiQuantIntra_MPEG4_16s_C11
- (5) 变长编码(Variable Length Code)
ippiEncode_IntraDCVLC_MPEG4_16s1u
ippiEncode_IntraACVLC_MPEG4_16s1u
ippiEncode_InterVLC_MPEG4_16s1u
- (6) 反量化
ippiQuantInvIntra_MPEG4_16s_C11
- (7) 反离散余弦变换(I-DCT)

ippiDCT8x8Inv_16s_C11

在实际的 MPEG-4 编码过程中, 一般采用 I-VOP 和 P-VOP 交替编码。这里采用相邻的 I-VOP 之间插入 4 个 P-VOP, 目的是使传输的码流低于 32 kb/s, 同时在网络出现丢包的情况下, 利用 I-VOP 能够在 1 s 之内迅速的恢复视频。

3.1 I-VOP 编码

I-VOP 编码过程如图 3 所示。

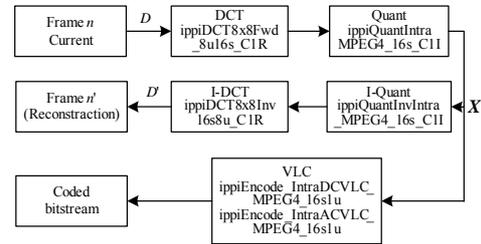


图 3 I-VOP 编码过程

(1) 输入视频的当前帧 F_n , 并以宏块(16×16 像素)为单位对该帧进行处理。

(2) 对每一个宏块 D 进行 DCT 变换(离散余弦变换)。通常每一个宏块被分成 8×8 的子块进行离散余弦变换, 得到一个系数矩阵 X 。

(3) 对该系数矩阵 X 进行量化(quantised), 并进行 AC/DC 因子的预测, 得到系数矩阵 X' 。

(4) 对量化后的系数矩阵 X' 进行变长编码(variable length coding), 去除统计冗余信息。

(5) 在输出的变长编码结果前加上一定的头信息, 输出到 MPEG4 视频流。

(6) 对量化后的系数矩阵 X' 进行反量化和反 DCT 变换, 重建宏块 D' 。

(7) 由重建的宏块 D' 重建当前帧 F_n' , 作为 P-VOP 编码的参考帧。

3.2 P-VOP 编码

P-VOP 编码过程如图 4 所示。

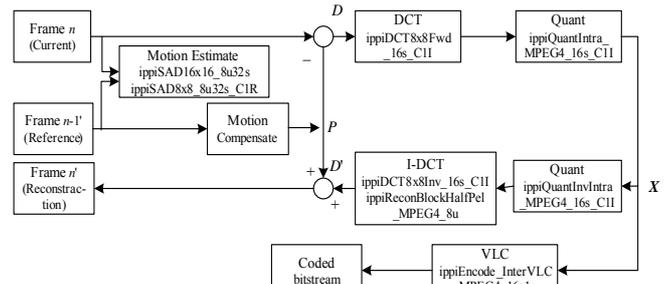


图 4 P-VOP 编码过程

(1) 输入视频的当前帧 F_n , 并以宏块(16×16 像素)为单位对该帧进行处理。

(2) 在参考帧 F_{n-1} 中, 对每一个宏块进行运动估计, 即在参考帧中找到与当前宏块相似度最高的宏块。则当前宏块的位置和搜索到的最匹配宏块之间的位移构成了当前宏块的运动向量。

(3) 根据得到的运动向量, 计算出当前宏块基于参考帧的一个预测值, 记为 P 。

(4) 当前宏块减去 P 即可得到残差宏块 D 。

(5) 对每一个残差宏块 D 进行 DCT 变换(离散余弦变换), 通常每一个宏块被分成 8×8 的子块进行离散余弦变换, 得到一个系数矩阵 X 。

(6)对该系数矩阵 X 进行量化(Quantised),并进行 AC/DC 因子的预测,得到系数矩阵 X' 。

(7)对量化后的系数矩阵 X' 进行变长编码(Variable Length Coding),去除统计冗余信息。

(8)在输出的变长编码结果前加上一定的头信息,就可以输出到 MPEG-4 视频流。

(9)对量化后的系数矩阵 X' 进行反量化和反 DCT 变换,重建宏块 D' 。

(10)由重建的宏块 D' 重建当前帧 F_n' ,作为 P-VOP 编码的参考帧。

在解码端,也可以采用 IPP 为 MPEG-4 解码所提供的函数来实现对应的视频流解码,MPEG-4 解码函数如下:

(1)变长编码解码(Variable Length Decode)

ippiDecodeVLCZigzag_IntraDCVLC_MPEG4_1u16s
ippiDecodeVLCZigzag_IntraACVLC_MPEG4_1u16s

(2)反 AC/DC 预测(I-DC/AC Prediction)

ippiPredictReconCoeIntra_MPEG4_16s

(3)反量化

ippiQuantInvIntra_MPEG4_16s_I
ippiQuantInvInter_MPEG4_16s_I

(4)反离散余弦变换(I-DCT)

ippiDCTInv_8x8_16s8u ippiDCTInv_8x8_16s_I

(5)运动补偿(Motion Compensation)

ippiDecodePadMV_PVOP_MPEG4
ippiLimitMVToRect_MPEG4
ippiMCReconBlock_RoundOn
ippiMCReconBlock_RoundOff
ippiMCBlock_RoundOn_8u
ippiMCBlock_RoundOff_8u

(6)色度空间转换(YUV to RGB)

ippiYUV420ToRGB_8u_P3C3

3.3 I-VOP 解码

I-VOP 解码过程如图 5 所示。

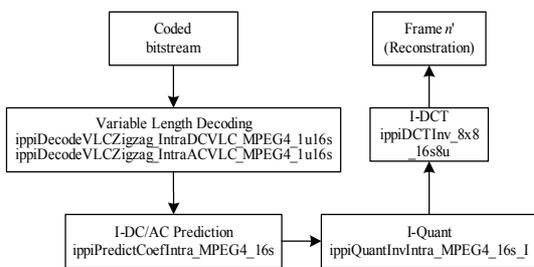


图 5 I-VOP 解码过程

(1)对 MPEG4 视频流解码,得到使用变长编码的宏块。

(2)对 1 中结果进行变长解码,再进行反 AC/DC 预测,得到量化后的系数矩阵 X' 。

(3)对矩阵 X' 进行反量化,以恢复系数矩阵 X 。

(4)对 D 进行反离散余弦变换,得到重建的宏块 D' 。

(5)由重建的宏块 D' 可以重建当前帧 F_n' 。

3.4 P-VOP 解码

P-VOP 解码过程如图 6 所示。

(1)对 MPEG4 视频流进行分解,得到使用变长编码的宏块和运动向量。

(2)对 1 中结果进行变长解码,再进行反 AC/DC 预测,得到量化后的系数矩阵 X' 。

(3)对矩阵 X' 进行反量化,恢复系数矩阵 X 。

(4)对 X 进行反离散余弦变换,得到重建的宏块(残差宏块) D' 。

(5)根据参考帧 F_{n-1}' 和运动向量,计算出当前宏块对应的运动估计值 P 。

(6) D' 加上 P 即可得重建的宏块。

(7)由重建的宏块重建当前帧 F_n' 。

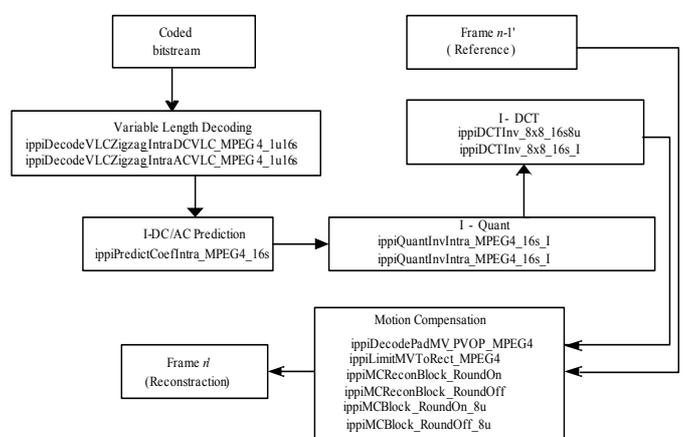


图 6 P-VOP 解码过程

3.5 实验结果与分析

笔者在 PXA255 上基于 IPP 函数实现了 MPEG-4 简单类 (simple profile) 视频编解码,并针对 YUV 测试序列 highway 和 bridge-close 进行了测试。其中,highway 为 QCIF 格式和 CIF 格式;bridge-close 为 QCIF 格式。由表 1 可以看出,测试程序针对 QCIF 格式视频以 10 帧/s 的速度进行编码,针对 CIF 格式的视频以大于 60 帧/s 的速度进行解码,能够满足实时视频传输的要求。

表 1 编解码结果数据

测试序列	帧数	编码		解码	
		时间/s	帧速/(帧/s)	时间/s	帧速/(帧/s)
highway	100	46.3	2.2	1.58	62.9
highway	200	21.0	9.5	1.11	180.1
bridge-close	100	8.8	11.3	0.37	270.0

4 结束语

本文利用 Intel IPP 函数库提供的函数进行 MPEG-4 视频编解码,系统在 Intel Xscale 处理器上实现效果良好。结果分析表明,采用这种方法进行实时图像编码及解码,能有效地提高嵌入式移动终端上多媒体实时处理能力,在众多领域有着较为广泛的应用前景。

参考文献

- [1] 王文庆,毛文林,张利飞.基于 MPEG-4 的视频监控系统模型及实现[J].计算机工程,2006,32(11):237-239.
- [2] 贾贵玺,李洪凤,叶军.基于 MPEG-4 的嵌入式 IP 网络监控系统[J].计算机工程,2004,30(19):171-172.
- [3] Richardson I E G. H.264 and MPEG-4 Video Compression[M]. England: John Wiley & Sons, 2004
- [4] Intel Corporation. Intel PXA255 Processor Developer's Manual [EB/OL]. (2003-03-01). http://www.idt.mdh.se/kurser/cdt214/PXA255_Developers_Manual.pdf.
- [5] Intel Corporation. Intel Integrated Performance Primitives for Intel Architecture Reference Manual[EB/OL]. (2003-03-02). http://www.jaist.ac.jp/iscenter/mpc/altix/opt/intel/iipp-3.0/ipp/doc/ip_piman.pdf.