

摘 要

随着社会应用需求的发展,实时视频编码技术得到了广泛的应用,尤其在实时监控方面显得更为重要。不仅是大的企业、单位需要进行安防监控,对于私家车等私人物品以及家庭安全的监控也显的尤为重要。中国电信利用 3G 网络提供全球眼来随时监控家庭及通过手机视频电话与家人沟通,这种家庭网络多媒体业务的发展也已经开始。但是视频编码作为视频技术的核心,现在并不能适应所有的环境,这也正是视频电话费用高,且不易推广的原因。为了解决这个问题,本文研究了当前最新视频编码技术 H.264,对其中的关键技术进行分析,最后以其中的运动估计及整数变换与量化为切入点,对算法进行了优化,使编码速度得到提高。最后通过实验测试数据可以看出,在保证视频质量满意的前提下,算法达到了提高视频编码速度的目的。本文所做的主要研究工作如下:

(1) 研究了 H.264 的关键技术运动估计的原理及方法,在学习 H.264 标准中采用的三种搜索算法的基础上选取其中一种效果比较好的算法—非对称十字多层次六边形搜索算法进行了改进。在该算法的搜索过程中加入了自适应的阈值,在搜索的过程中起到了提前截止的作用。并且将其五步搜索,增加一步矩形搜索,大多数情况下避免了耗时的六边形搜索。由于运动估计在整个编码的过程当中,属于较费时过程,所以通过改进,提高了运动估计的速度,从而提高了整体的编码速度。

(2) 整数变换及量化的过程在整个编码的过程当中占有很重要的地位,也是一个非常耗时的步骤。研究了 H.264 所采用的整数变换与量化的过程,通过研究全零块判决的算法,在整数搜索的过程当中加入了全零块判决过程,对于检测到的全零块,省去其编码及量化过程。并且在普通全零块判决算法的基础上,对阈值的选取及搜索过程做了进一步的改进。由于全零块所占比率比较大的原因,经过本文的改进,进一步的缩短了编码的时间。

(3) 最后利用 H.264 标准的软件参考模型 JM14.0 对本文所提的算法进行了实现及实验测试。从实验结果可以看出,本算法所实现的编码图片属于较好的效果,并使实时视频编码的速度得到了很大的提高。

关键词: 实时视频编码, H.264, 运动估计, 变换与量化

Abstract

With the development of social applications, real-time video encoding technology is widely used, especially in real-time monitoring. Not just large enterprises need security monitoring, something personal belongings such as private cars and family thing also important. China Telecom use 3G network realization the world's eyes to monitor the family at any time and adopt the mobile video telephone to communication with families. The development of the home network of multimedia services has begun. However, video encoding as the core of video technology, can not meet all kinds of environment, which is the reasons of the high cost of video telephony and hard to promoting. To solve this problem, this paper study of the H.264 which is the most popularly coding technology current in-depth, and analyzes the key technology of H.264. In the end select the motion estimation and integer transform and quantization as the breakthrough point to improve and optimized the speed of encoding . Finally, under the premise of a satisfactory video quality achieve the purpose of speed up the video encoding. The main researches in this dissertation are illustrated as follows:

(1) Research of the Principle and methods of Motion Estimation which is the key technologies of H.264 deeply. Study the three types of search algorithm used to carry out in the Official version detailed, at the last selected one of the better algorithm-Unsymmetrical-Cross Muti-Hexagon Search to research and improve. Add the adaptive threshold in the process of the search algorithm, which played a role of early cut-off. And increase an rectangular search step to its five-step search, in most cases it can avoid the time-consuming six polygon search step. Because the motion estimation are more time-consuming process in the encoding, through our work improved the speed of motion estimation, thereby increasing the overall speed of encode.

(2) Integer transform and quantization process plays an important role in the whole process of encoding, also a very time-consuming step. This article in depth

study and research the integer transform and quantization process which adopted by the H.264, as well as its basic theory -discrete cosine transform knowledge. Through our study and research and studied the algorithm of all-zero block of the decision process, finally joined the all-zero block of the decision process in the whole search process, save the coding and quantization time for the detected zero-block-wide. And based on the algorithm of all-zero block of the decision process,further improvements the threshold selection and search process. Due to the reason of All-zero block relatively large percentage, after the improvement of this article, further shortened the time of encoded.

(3) Finally, use the version of the official JM14.0 to validate the proposed algorithm. According to the experiments, the realization of the encoding algorithm is a better picture of the results and demonstration by experiments that the encoding speed has been greatly improved.

Keywords: Real-time Video Encoding, H.264, Motion Estimation, Transform and Quantization.

独创性声明

本人声明，所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得武汉理工大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签名：薛志坚 日期：2009.5.22

学位论文使用授权书

本人完全了解武汉理工大学有关保留、使用学位论文的规定，即：学校有权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。本人授权武汉理工大学可以将本学位论文的全部内容编入有关数据库进行检索，可以采用影印、缩印或其他复制手段保存或汇编本学位论文。同时授权经武汉理工大学认可的国家有关机构或论文数据库使用或收录本学位论文，并向社会公众提供信息服务。

(保密的论文在解密后应遵守此规定)

研究生(签名): 李福 导师(签名): 李福 日期 2009.5.22
薛志坚

第 1 章 绪论

实时视频技术在当今社会得到了非常广泛的应用,在远程教学、视频监控、娱乐及军事等领域都用到了实时视频技术。视频监控系统对视频实时传输的要求更为严格一些,所以论文算法应用背景着重考虑视频监控及视频电话的需求。视频监控以其直观、方便、信息内容丰富而广泛应用于许多场合。近年来,随着计机、网络以及图像处理、传输技术的飞速发展,视频监控技术也得到了长足的发展^[1]。视频编码技术作为视频技术的核心,对视频质量好坏及实时性的实现有很大的影响。为了满足普通家庭及个人的需求如:家庭安全监控,私家车等私人物品的监控等等。我们必须拥有更高效的视频编码技术,这样才能使视频技术的应用变的更加广泛及易推广。

1.1 研究背景

1.1.1 课题的题目及来源

课题的题目:实时视频编码优化研究

课题的来源:教育部博士点基金(项目编号:20050497006)

1.1.2 课题研究的目的和意义

视频编码技术是视频技术的核心,而视频技术又是实时监控的核心。通过对视频编码技术的深入研究,使编码速度得到提高,便可使实时视频传输更容易的实现。在实时视频技术的支持之下,可以使视频监控的作用发挥到最大。结合实时捕捉、视频报警、硬盘录像等技术的视频监控技术,使防范工作更安全,更周到,更方便,对罪犯起到了强大的威慑力,能更有效地防止犯罪;严密的计算机监控网络,跨越地域时空,实时跟踪每一个监控现场;迅速的反映能力,能更有效地打击犯罪^[2]。数字视频监控系统的远程访问能力在商业中得到了广泛的应用,使管理者可以远程管理和调配资源,可以很大的节省时间,并且减少因为信息失误而造成的损失。实时视频的功能同样的可以方便现在人与人之间的感情交流,常年无法回家的工作人员可以通过网络视频,或者视频电

话与家人进行一种“面对面”的交流。这将非常有利于我们社会的和谐发展。

由于论文所提算法的使用背景着重考虑实时视频监控及视频电话，所以接下来我们将详细了解一下实时视频监控系统的架构。我们将通过实时监控系统的各个组成部分，来分析实时视频编码技术的重要性。实时视频监控系统是以计算机为中心，数字视频处理技术为基础，利用图像数据压缩的国际标准，综合利用图像传感器、计算机网络、自动控制和人工智能等技术的一种新型监控系统^[3]。实时视频监控系统将摄像机获得的模拟电视信号转变为数字视频信号以便于计算机处理，或者由数字摄像机直接输出数字视频信号，在计算机显示器上显示多路活动图像的同时，可将各路信号分别存储于计算机的硬盘内或在网络上进行传输。在实时情况下，每路信号在监视、记录、回放时都能达到最大的活动图像的效果，其原理如图 1-1：

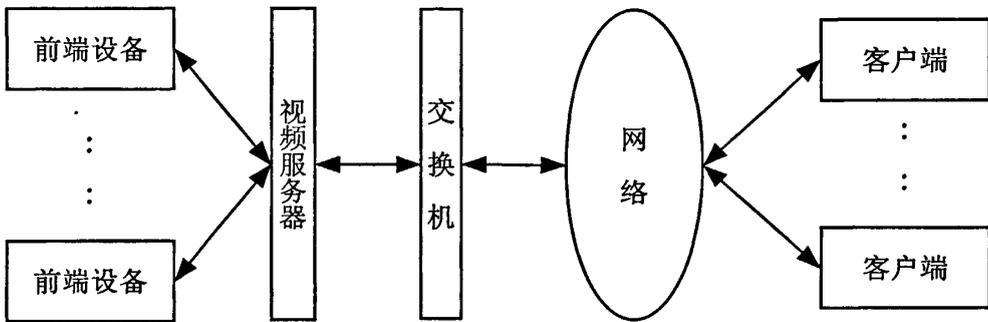


图 1-1 数字视频监控原理结构图

由于视频信息的原始信息量很大，而且实时监控系统是不间断的在采集数据，所以信息量必然非常庞大，这给视频数据的存储以及查看造成了很大的困难。所以我们在对视频数据进行存储的过程中必须先对原始视频信息进行编码压缩^[4]。在对视频的压缩过程中如果一味的追求高压缩率将丢失很多视频信息，使得数据的重现效果很差，但是在保证数据完好无损的情况下，又得不到理想的压缩率，所以在对视频进行压缩的过程中，必须通过实验追求一个可以接受的视频质量情况下的最大压缩率。现在我们常用的主流视频编码标准主要有 MPEG-2、MPEG-4、H.261、H.263、H.264。H.261 是 ITU-T 为了在综合业务数字网 (ISDN) 上开展双向声像业务 (可视电话、视频会议) 而制定的，速率为 64KB/S 的整数倍，H.263 是 ITU-T 为低于 64KB/S 的窄带通信信道制定的视频

编码标准。它是在 H.261 的基础上发展起来的,与 H.261 相比采用了半像素的运动补偿,并增加了 4 种有效的压缩编码模式,MPEG-2 是 MPEG 组织于 1994 年推出的用来实现视/音频服务与应用互操作的可能性^[5]。编码率从每秒 3M 比特~100M 比特,MPEG-2 不是 MPEG-1 的简单升级,MPEG-2 在系统和传送方面作了更加详细的规定和进一步的完善。主要的优点在于:MPEG-2 的编码图像被分为三类,分别为 I 帧,P 帧和 B 帧,从而最大限度的去除了图像中的冗余信息,提高了图像的压缩比。MPEG-4 于 1998 年 11 月公布,MPEG-4 不仅是针对一定比特率下的视频、音频编码,更加注重多媒体系统的交互性和灵活性。传输速率在 4800-64000bits/sec 之间,MPEG-4 利用很窄的带宽,通过帧重建技术,压缩和传输数据,以求以最少的数据获得最佳图像质量。H.264 是由 ISO/IEC 与 ITU-T 组成的联合视频组(JVT)制定的新一代视频压缩标准。它的特别之处在于支持低码率的视频通信,H.264 比 H.263+和 MPEG-4 (SP)减小了百分之五十的码率。

在对视频信息进行远程传输时,我们必然通过网络等条件进行传输,在这个过程中,有很多技术与之有关。网络条件的好坏显然非常重要,但是我们并不能保证网络时时刻刻都处在一个很好的状态,更何况很多情况下,网络环境本来就不是很好,如无线网络的传输,以及窄带的传输等。而在这个过程中,压缩技术的好坏,同样的影响着实时性的实现,由于传输的码率如果可以降到一个网络环境很差的情况下也可以实现的标准,一切问题都将得到解决。理想的状态就是压缩标准及传输标准根据实时网络环境来进行调节,网络环境好的情况下可以质量较好的压缩来传输,也可以以一个相对较快的速度来传输。网络环境质量较差的情况下,将压缩率作为主要目标,画面质量达到最低要求即可。这样同样可以达到我们的目的,并且保持视频的流畅。H.264 标准已经设置了不同的档次来用于不同的用途,但是一个标准的建立,只能以整个社会环境来衡量,这种设置并不能满足所有的情况,我们可以沿着这种思路去进一步的优化算法,使得算法可以接受最恶劣的网络环境来实现实时性。

因此,面对社会各界对视频技术的需求,以及视频传输所需要的必要条件,研究一种可以最大限度适应网络环境的编码方法,将对实时视频技术的发展具有重要的意义,也对人类社会的和谐发展具有非常重要的意义。

1.2 相关技术国内外研究现状

1.2.1 国外研究

从国外运营商发展情况看,相对于视频方面的技术要先进很多,已经深入到了家庭信息化阶段。家庭综合通信、娱乐及综合信息服务、家庭安防、智能家居及远程控制是家庭信息化产品的四个主要方向,家庭视频监控则是家庭安防的核心产品。国外运营商在家庭视频监控方面发展较快,以 at&t、英国电信、法国电信、西班牙电信、NTT、KT、SK 电讯为代表的运营商已经基于手机、PC 和摄像头,通过移动和无线网络,提供了家庭视频监控业务,有的还提供了附加的 IPTV、VoIP 等服务。英国电信(BT)将数字家庭网关称作 HomeHub,通过该网关为用户提供信息、通信、娱乐的综合服务,强调易用性,也就是即插即用和丰富的连接,还强调完整性,通过多样化的家庭终端提供多样化的服务,支持各种智能监控设备的数据接口和物理接口,在视频监控上更侧重于网络摄像机的方案^[6]。法国电信也有相应的动作,它以 Live 为统一品牌,推出了 LivePhone、LiveMusic 等设备,将 LiveBox 通过 ADSL 接入到互联网,然后通过数字家庭内的无线接入方式接入各种监控设备和智能家电。法国电信将视频监控称为 LiveZoom,采用的是 AXIS206W 无线网络摄像机,内嵌 Web 服务器,功能非常丰富,被誉为世界上最小的无线网络摄像机。西班牙电信的家庭网络业务遵循“通信——多媒体——安全、本地和远程管理——自动控制”的路线,在业务提供上具有带宽充足、可支持多个家庭子网的特点。西班牙电信采用 OSGI 的基于 WebService 的方案对家庭内各个设备实现基于 Web 形式的监控,比较侧重于无线网络摄像机监控方案。它的视频监控对任何业务都有充足的带宽提供,支持多个家庭子网,满足家庭中不同业务之间 QoS 的需要;通过 Wi-Fi 网关实现家庭网络与接口网络的互联,尽量使用无线,在屋内的任何地方都可以接入各种服务,对室内设备可以远程管理和维护,远程监控设备也多采用无线的 Wi-Fi 协议接入网关。西班牙电信的视频监控方案被业界称为充满艺术气息的无线智能化数字家庭解决方案。美国 at&t 结合它在固定电话网、移动电话网、有线电视网、互联网等网络基础设施上的优势,与杰尔系统的 TrueONE 解决方案相结合,为用户提供电信级的手机或 PC 上的视频监控应用,以及 IPTV、VoIP 等大流量应用,将数字家庭的智能监控服务送达到远端的 PC 或手机上,支持无线和有线的视频监控方案。日本的 NTTdocomo 利用自身在移动服务上的优势,推

出了 i-mode 运营模式,可以直接通过手机或 PDA 远程控制家中的智能家电,在远程监控上采用了“M2M 远程控制”的理念,实现人和机器、机器和机器之间的远程控制。NTT 推出了可通过因特网或 FOMA 手机控制的 HC-1000 监视摄像头,该设备采用 300 万像素 CMOS 感光器,2 倍数码变焦,以太网接口,支持 802.11a/b/g,可拍摄 QVGA 画质的视频;内建“主动防御”系统,可发出刺耳的警报声吓跑盗贼,甚至还有麦克风/扬声器双向通信系统,方便用户与盗贼谈判。NTT-Neomeit (NTT 西日本的子公司)还推出了通过 PDA 或手机 远程控制家电和接收监控报警信息的服务,使用起来非常方便。除此以外,还实现了生物认证,通过生物认证研发了自动门识别系统,人们站在安装于入口处的摄像机前,只需约 1 秒钟的时间,如果确认来人为公寓居民,大门就会打开,非常方便^[7]。韩国的 KT 和 SK 电讯在数字家庭的远程监控和报警上大量采用了无线传感网络中的普适计算技术 (Ubiquitous Computing),采用 Zigbee 的无线传输协议对家中的智能监控设备进行管理和监控,并支持通过 PC 或手机对数字家庭设备远程监控信息的获取和基于 IP 的视频监控点播,用户点播时直接由数字家庭网关提供视频的处理。KT 将数字家庭监控及联动报警系统称为 Nespote 系统,该系统立足于“控制与防止”,将有线与无线网络结合于一体。采用 Nespote 系统,不论用户在家还是在外,都可通过微型监视摄像头、安装在门上的传感器、煤气泄漏传感器等,将家庭状况实时传到用户的电脑、手机或 PDA 上。用户也可以远程遥控开灯,使家里看上去有人在。紧急情况下,用户还可以呼叫急救中心对家中的病人进行救助。

1.2.2 国内现状

国内现在针对这方面的研究主要集中在了企业、军事、工厂等大规模的监控,对于家庭信息化业务处于刚刚起步阶段。实时监控系统被广泛应用于金融系统、交通系统、公安系统、教育系统和医疗系统等众多领域,纵观监控技术的发展历程,大致经过了三个发展阶段:模拟监控、数字监控、网络多媒体化监控。在黑白摄像机、黑白电视机出现的初期,初期的黑白图像电视监控系统也诞生了。它被应用于保安、生产管理等场合,尽管当时图像质量低,控制功能少,但其直观方便的特性还是被人们所接受,之后投入了大量的人力物力,不断地提高性能、完善功能,很快地由最初的单点、单线制模式变成了点对点、一点对多点监视和多点顺序控制系统。利用当时的电子技术,研制、开发出各

种镜头切换器、云台控制器，以及满足各种场合应用要求的不同种类如红外、防水、防爆、防热等类型的云台和摄像机。当时的监控系统大多采用模拟方式传输，最简单的是将图像基本信息直接送入视频传输电缆进行传送。每一监控点的视频信息都通过一条电缆直接连接到中心控制室的多路切换控制台上，通过手动切换方式选择监控点图像使之显示在电视机上，随着数字化技术的发展，给图像监控技术带来了一次技术革命^[8]。人们从过去模拟处理方式转向数字处理方式，开始使用计算机处理图像信息。利用计算机的数据处理能力与显示器的高解析度，将视频信息通过视频采集卡采集到计算机中，显示在显示器上，大大提高了图像的画质。采用数字监控系统，对目标范围实时监控，并以此为中心建立的一整套软硬件结合的完整体系，减少不必要的环节和操作，提高了整个系统的反应速度和效率。同时引入模块化管理，将监控所涉及的视频信息的采集、处理，视/音频切换，云台、摄像镜头控制，报警采集和处理等内容模块化，相互关联，相互统一。随着数字视频压缩编码技术的日益成熟，为多媒体信息的广泛应用提供了基础，新的监控技术完全打破了传统的结构，依靠功能日益强大的计算机，不仅可以处理文本、数据、图形等信息，还可以处理视频图像、声音等信息，再加上网络技术、通信技术的迅猛发展，对各种媒体信息的应用趋向综合化、交互化。现在的各种通信网如公用电话网、数字数据通信网、综合业务数字网、各种类型的 LAN/WAN、以及发展迅猛的因特网遍布全球的各个角落，将全球连成一体，使您在任何地方都能与地球上其它任何地方交流与传递任何信息。监控系统涉及的区域范围也在不断地扩大，从局域到城域到广域，从一个城市到另一个城市，从一个国家到另一个国家，都能完成在现场所能完成的一切任务，将建成一个网络化的多媒体监控系统^[9]。

中国电信（全球眼）和原中国网通已经在有条件的家庭发展远程视频监控，如宽视界视频监控业务前期已在中国网通的山西、山东、吉林、浙江、河南等省公司推出，受到了客户的普遍欢迎，目前宽视界业务发展迅速，2008 年上半年，仅湖北网通“宽视界—神眼”业务就发展了 500 多户用户。随着家庭信息化的迅猛发展，用户对家庭安防的需求表现得越来越强烈，宽视界将会保持迅速的发展中国移动也在发展以“手眼通”为代表的手机彩信监控，基于 3G 的视频监控也在测试和试商用中。由于数据和信号流庞大，对网络依赖性强，需要家庭部署家庭网关和摄像头等，因此家庭视频监控研发和推广难度还比较大，目前视频监控主要应用在行业领域和家庭紧密联系的小区，家庭监控应用得还比较少。

在平安城市、平安家庭政策的引导下,伴随着家庭网络环境不断改善,用户消费能力提升,安防意识增长,独立式居住家庭增多,国内家庭监控将开始兴起。在我国全业务运营已经到来,运营商必然会给用户包括有线和无线在内的多样化技术监控手段,通过业务融合,给用户带来无所不在的监控服务。监控实现了对家庭安全的发现和预防,也是对家庭关爱的体现。

针对本课题的主要工作优化及改进 H.264 算法的工作,国内外研究现状基本一致。由于应用领域的差别,对视/音频编解码器有不同的优化要求,但它们的共同点是:算法复杂度要尽可能小、算法处理的时间尽可能短、性价比尽可能高。优化工作主要在两个方面:(1) 软件优化,主要包括算法级别的优化和代码级别的优化。算法级别的优化是指采用快速算法实现编解码器中时间复杂度较高的部分,对于 DCT 变换和运动估计类型的 MPEG 编解码算法,采用快速的 DCT,算法和快速的运动估计算法,可以极大的降低 DCT 变换和运动估计的时间复杂度。代码级别的优化是指利用 PC 机 CPU 指令集中相关指令编写程序代码,从代码执行速度上提高编解码算法的执行速度。(2) 硬件优化,主要从三个方面进行考虑:一是减少每个操作的运行周期;二是提高每个周期能够完成的操作数;三是减少总共需要完成的操作数^[10, 11]。

1.3 本文主要工作和组织结构

1.3.1 主要研究工作

本文对实时视频技术以及最新实时视频编码算法 H.264 的关键技术进行研究,主要研究内容包括,研究 H.264 的编解码流程以解决主要改进部分切入点寻找,研究运动估计理论以及分块搜索技术以解决对 H.264 算法中的运动估计算法优化工作,研究整数 DCT 变换、蝶形算法、量化部分以解决对 DCT 变换与量化的优化工作。

主要工作体现在以下几个方面:

(1) 针对运动估计算法优化,首先研究与学习了 H.264 标准的编码与解码流程,并且在此基础上对运动估计部分,尤其是运动搜索部分进行了研究,着重研究了 H.264 参考软件模型 JM14.0 包含的三种搜索模式:全搜索模式、UMHexagonS 模式、以及 EPZS 模式。最后从 UMHexagonS 算法入手对该算法从提前截止搜索,以及搜索范围优化方面进行了改进,最后通过实验证明该优

化算法比原算法在效果上有很大的改进。

(2) 针对变换与量化算法的优化, 通过对 DCT 变换与量化的深入研究, 着重研究了 H.264 所采用的整数 DCT 变换与量化知识。最后对变换与量化算法加入了全零块检测步骤, 对于将要进行变换和编码的系数块, 在做变换域编码之前, 可以先进行全零块检测, 如果事先检测为全零块, 就可以避免相应的变换域编码和量化, 从而可以极大地降低变换和编码时间。经过实验验证与官方代码的对比, 证明本算法在编码速度上有了很大的提高。

1.3.2 本文组织结构

本文的内容分为五章, 具体安排如下:

第 1 章 绪论, 介绍了本课题来源、研究目的和意义、以及本文的主要工作。

第 2 章 对 H.264 运动估计算法的优化, 研究与学习了 H.264 编码标准, 对编码及解码流程进行了深入研究, 最后通过对运动估计理论, 几种常见搜索算法的研究, 给出了本文对运动估计算法的优化方法。

第 3 章 H.264 整数变换与量化算法的优化, 本章深入的研究了整数变换与量化的具体知识, 从研究的过程中发现其中可以进行改进的部分, 最后对整数变换与量化进行了算法上的改进及代码方面的优化。

第 4 章 实验结果分析, 在进行完以上章节工作的改进之后, 对改进后的算法进行仿真实验, 并将实验数据与标准模型以及参考文献中的实验结果进行对比, 并且更形象的通过对编码解压后的视频效果进行对比, 从而得出本算法改进后的先进性。

第 5 章 总结与展望, 对本文的工作进行了总结, 并且指出实时视频技术所需要进一步进行的工作, 以及本算法应该进一步研究的方向。

第 2 章 H.264 运动估计算法的优化

2.1 引言

ITU-T 与 ISO/IEC 是制定视频编码标准的两大主要组织：ITU-T 的标准包括 H.261、H.263、H.264、主要应用于实时视频通信领域，如会议电视、视频电话等。MPEG 系列标准是由 ISO/IEC 制定的，主要应用于视频存储（DVD）、广播电视、因特网或无线网上的流媒体等。两个组织也共同制定了一些标准，H.262 标准等同于 MPEG-2 的视频编码标准，而最新的 H.264 标准则被纳入了 MPEG-4 的第十部分^[12]。各种主流编码标准的发展史如图 2-1 所示：

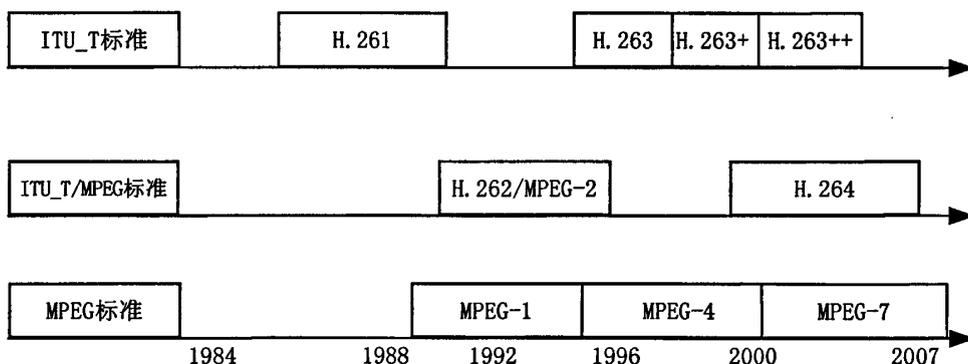


图2-1视频编码标准发展史

H.264 是由 ISO/IEC 与 ITU-T 组成的联合视频组（JVT）制定的新一代视频压缩标准。在众多的编码标准中，H.264 视频编码标准在编码效果方面是最突出的，与别的编码标准相比 H.264 具有以下一些优点^[13]：

(1) 在相同的重建图像质量下，H.264比 H.263+和 MPEG-4（SP）减小百分之五十的码率。

(2) 对信道时延的适应性较强，可工作于低时延模式以满足实时业务，如会议电视、实时安全监控、视频电话等；也可工作于无时延限制场合，如视频存储等。

(3) 网络适应性强, 采用“网络友好”的结构和语法, 加强对误码和丢包的处理, 提高解码器的差错恢复能力。

(4) 在编码器与解码器中采用复杂度可分级的设计, 在图像质量和编码处理之间可分级, 以适应不同复杂度的应用。

(5) 相对于以前的视频压缩标准, H.264引入了一些先进的技术, 主要包括4×4整数变换、空间域内的帧内预测、1/4像素精度的运动估计、多参考帧与多种大小块的帧间预测技术等。

以上所有这些技术都使 H.264的压缩比得到了大大的提高, 同时也加大了算法的复杂度。这使得 H.264的算法对实现视频传输的实时性有了很大的阻碍, 对于一些硬件条件不是太好或者网络环境较差的情况下, 实现实时性更是困难。本章将在深入研究 H.264结构以及核心技术的基础之上, 对其中耗时比较厉害的运动估计部分进行算法的改进及代码的优化。

2.1.1 H.264 的层次结构

H.264 标准压缩系统由视频编码层 (VCL) 和网络提取层 (NAL) 两部分组成。VCL 中包括 VCL 编码器与 VCL 解码器, 主要功能是视频数据压缩编码和解码, 它包括运动补偿、变换编码、熵编码等压缩单元。NAL 则用于为 VCL 提供一个与网络无关的统一接口, 它负责对视频数据进行封装打包后使其在网络中传送, 采用统一的数据格式^[13]。NAL 层可以使 H.264 的 VCL 层和其他一些多媒体传输层实现映射, 并通过这些层实现传输编码得到的数据功能, 其层次结构图如图 2-2:

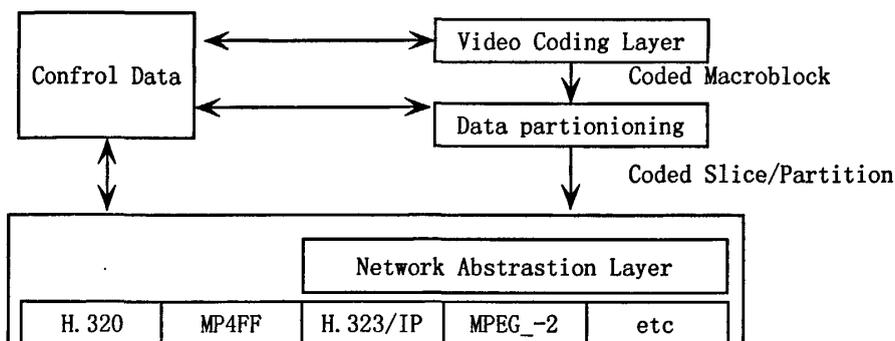


图 2-2 H.264 编码器层次结构图

2.1.2 H.264 的档次分类及应用场合

H.264 规定了三种档次，每个档次支持一组特定的编码功能，并支持一类特定的应用^[14]。

(1) 基本档次：利用 I 片和 P 片支持帧内和帧间编码，支持利用基于上下文的自适应变长编码进行熵编码 (CAVLC)。主要用于可视电话、会议电视、无线通信等实时视频通信。

(2) 主要档次：支持隔行视频，采用 B 片的帧间编码和采用加权预测的帧内编码；支持利用基于上下文的自适应算术编码 (CABAC)。主要用于数字广播电视和数字视频存储。

(3) 扩展档次：支持码流之间有效的切换 (SP 和 SI 片)、改进误码性能 (数据分割)，但不支持隔行视频和 CABAC，主要应用于流媒体中。

如图 2-3 所示为 H.264 的档次图。其中，各个档次具有的不同功能，可见扩展档次包括了基本档次的的所有功能，而不包括主要档次的全部功能。每一档次设置不同的参数 (如取样速率、图像尺寸、编码比特率等)，得到对应的编码器性能的不同级。本文将主要研究基本档次。

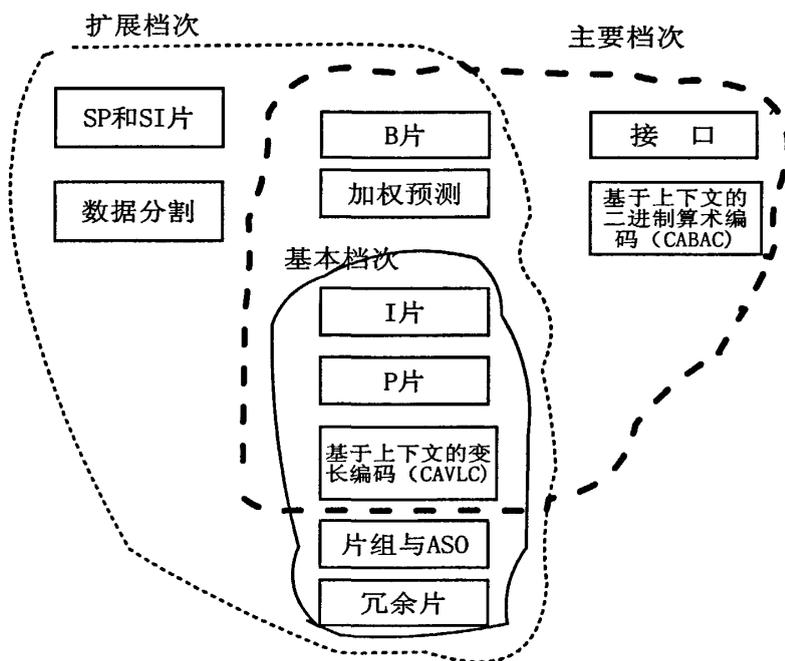


图 2-3 H.264 档次

2.1.3 H.264 的编解码流程

H.264 并不明确地规定一个编码器如何实现,而是规定了一个编了码的视频比特流的句法,和该比特流的解码方法,各个厂商的编码器和解码器在此框架下应能够互通,在实现上具有较大的灵活性,而且有利于相互竞争,H.264 的编码器和解码器的功能组成分别如图 2-4 和 2-5 所示^[15]。

编码器采用的仍是变换和预测的混合编码法,输入的帧或场 F_n 以宏块为单位被编码器处理。首先按帧内或帧间预测编码的方法进行处理。如果采用帧内预测编码,其预测值 PRED 是由当前片中已编码的参考图像运动补偿 (MC) 后得出的,其中参考图像用 F_{n-1} 表示。为了提高预测精度,从而提高压缩比例,实际的参考图像可在过去或未来已编码解码重建和滤波的帧中进行选择。预测值 PRED 和当前块相减后,产生一个残差块 D_n ,经块变换、量化后产生一组量化后的变换系数 X,再经熵编码,与解码所需的一些信息(如预测模式量化参数、运动矢量等)一起组成一个压缩后的码流,经 NAL(网络自适应层)供传输和存储用^[16]。正如上述,为了提供进一步预测用的参考图像,编码器必须有重建图像的功能。因此必须使残差图像经反量化、反变换后得到的 D_n' 与预测值 P 相加,得到 uF_n' (未经滤波的帧)。为了去除编码解码环路中产生的噪声,提高参考帧的图像质量,从而提高了压缩图像的性能,设置了一个环路滤波器,滤波后的输出 F_n' 即为重建图像,可用作参考图像。

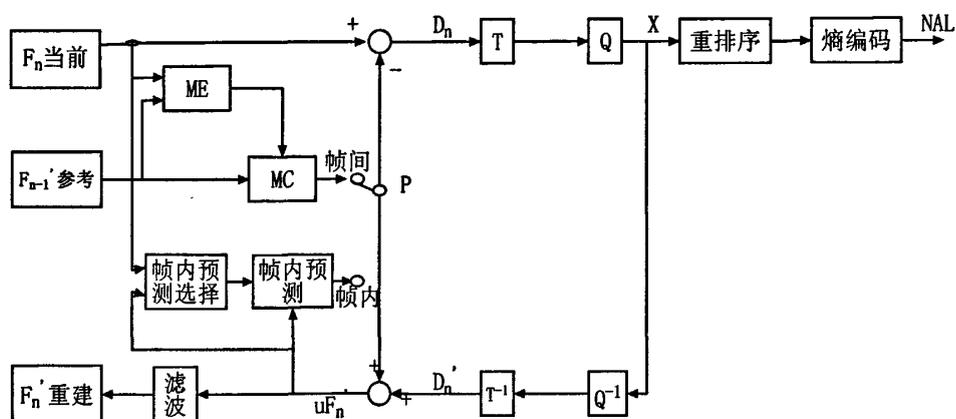


图 2-4 H.264 编码器

由图 2-5 可知,由编码器的 NAL 输出一个压缩后比特流。在图 2-6 中经解码得到量化后的一组变换系数 X,再经反量化、反变换,得到残差 D_n' 。利用

从该比特流中解码出的头信息，解码器产生一个预测块 PRED，它和编码器中的原始 PRED 是相同的。当该解码器产生的 PRED 与残差 D_n' 相加后，就产生 uF_n' ，再经滤波后，最后得到重建的 F_n' ，这个 F_n' 就是最后的解码输出图像。

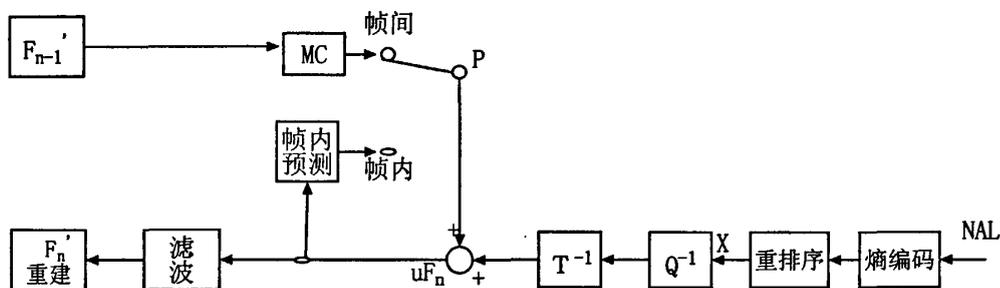


图 2-5 H.264 解码器

2.1.4 H.264 的关键技术研究

(1) 帧内预测

通过实验发现，帧内相邻宏块通常具有很类似的性质，因此，H.264 利用相邻宏块的空间相关性来进行帧内预测编码，以提高帧内编码的效率。对于一个给定的宏块，首先利用相邻的周围宏块对此宏块进行预测（通常利用此宏块左边和上边的宏块来预测，因为它们已经被编码并重建过），然后对实际宏块和预测宏块对应像素做差值，并对差值变换编码，从而可以在一定程度上减少一个 I 帧的数据量。H.264 的帧内预测编码方法一个显著的特征就是帧内预测在空间域进行，而不像其他视频编码标准（如 H.263+、MPEG-4）一样在变换域^[17]。

为进一步提高预测编码效率，对于亮度分量，H.264 对含有较多空域细节信息的宏块采用 4×4 预测（9 种模式），而对于较平坦的区域采用 16×16 的预测模式（4 种模式），另外还提供了不经预测和变换量化的步骤 I-PCM 模式。对于 4×4 的亮度块预测，即把 16×16 的亮度宏块划分成 16 个 4×4 的子块，每个子块的像素点 a-p 用与其相邻的上面和左面已经编码并重建的像素采样点 A-M 进行帧内预测。

对于 16×16 的亮度预测，类似于 4×4 亮度预测，用与每个宏块相邻的已编码重建的像素采样点值对宏块像素值进行预测，对于 8×8 的色度预测，其 4 种预测模式和 16×16 亮度预测模式相似。

(2) 帧间预测

H.264 的帧间预测是基于块的预测，它利用已编码重建的帧/场对当前帧/场进行运动补偿，它与先前视频编码标准不同的地方是:可变块尺寸、多参考帧以及小数像素。下面分别介绍这三个特点^[18]:

1) 多尺寸块模式

H.264 采用如图 2-6 所示的可变尺寸块进行帧间预测，其中，一个 16 x 16 的亮度宏块可以划分为 16x16, 16x8, 8x16 和 8 x 8 四种模式 (图 2-6(a)), 每个划分区域都是一个宏块分区;如果选择 8x8 模式，那么宏块中的 8x8 分区还可以再细分为 8x8, 8x4, 4x8 和 4x4 四种模式 (图 2-6(b)), 称为宏块子分区。这些块尺寸模式可以形象地表示成树形结构，如图 2-7 所示。其中，大的分割尺寸适合于平坦区域，而小尺寸适合于多细节区域。

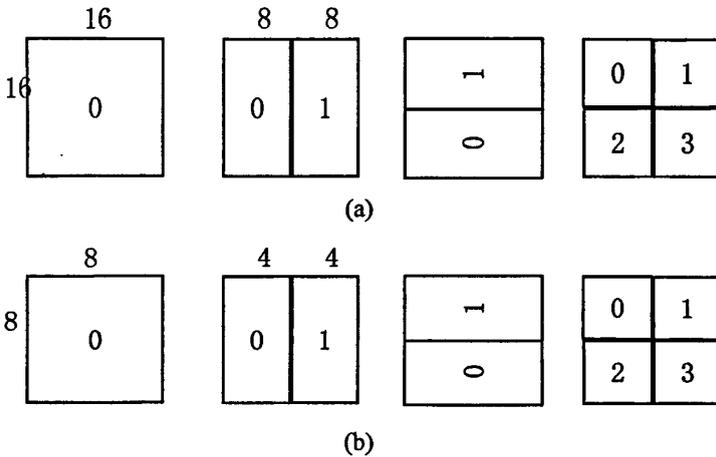


图 2-6 (a)宏块分区 16×16、8×16、16×8、8×8 模式
(b)宏块子分区 8×8、4×8、8×4、4×4 模式

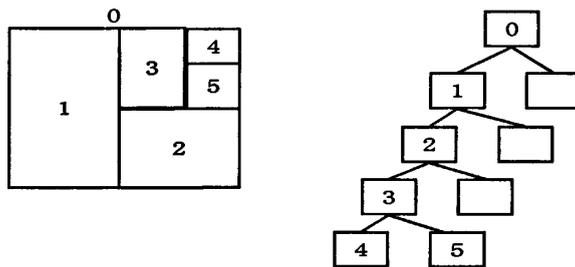


图 2-7 变尺寸块结构的树形模式

2) 双向、多参考帧

在 H. 264 中, 可采用多个参考帧的运动估计, 即在编码器的缓存中存有多多个刚刚编码好的参考帧, 编码器从其中选择一个给出更好的编码效果的作为参考帧并指出是哪个帧被用于预测, 这样就可获得比只用上一个刚编码好的帧作为预测帧更好的编码效果。H.264 帧间预测环节中包含双向的多个参考帧, 每个方向上最多可以有 15 个参考图象。

3) 小数像素

在 H. 263 中采用了半像素估计, 在 H. 264 中则进一步采用 1/4 像素, 以运动矢量位移的精度, 降低码率。H. 264 中采用了 6 阶 FIR 滤波器的内插获得 1/2 像素位置的值。当 1/2 像素值获得后, 1/4 像素值可通过线性内插获得。对于 4:2:0 的视频格式, 亮度信号的 1/4 像素精度对应于色度部分的 1/8 像素的运动矢量, 因此需要对色度信号进行 1/8 像素的内插运算。理论上, 如果将运动补偿的精度增加一倍可有 0.5bit/Sample 的编码增益, 但实际验证发现在运动矢量精度超过 1/8 像素后, 系统基本上就没有明显增益了, 因此, 在 H. 264 中, 只采用了 1/4 像素精度的运动矢量模式, 而不是采用 1/8 像素的精度^[19]。

(3) 整数变换和量化

视频压缩编码中以往的常用单位为 8×8 块。在 H.264 中却采用小尺寸的 4×4 块, 由于变换块的尺寸变小了, 运动物体的划分就更为精确。这种情况下, 图像变换过程中的计算量小了, 而且在运动物体边缘的衔接误差也大为减少。

当图像中有较大面积的平滑区域时, 为了不产生因小尺寸变换带来的块间灰度差异, H. 264 可对帧内宏块亮度数据的 16 个 4×4 块的 DCT 系数进行第二次 4×4 块的变换, 对色度数据的 4 个 4×4 块的 DC 系数 (每个小块一个, 共 4 个 DC 系数进行 2×2 块的变换^[20]。

H.264 不仅使图象变换块尺寸变小, 而且改用整数变换, 并用蝶形运算实现, 这样避免了乘法运算, 且能够在不影响编码性能的前提下, 有效地降低计算复杂度, 也更有利于硬件实现。由于编码器和解码器的变换和反变换的精度相同, 可以有效地消除“反变换误差”。

(4) 熵编码

多媒体数据压缩的方法可分为有损压缩和无损压缩两种, 而无损压缩可称为熵编码或者冗余编码, 这种编码是利用信源的统计特性进行压缩编码的也称

统计编码。视频编码中常用的是熵编码中的变长编码（也称 huffman 编码）和算术编码。

H.264 中所要编码的参数包括:序列层、图像层和片层（Slice）的语法单元，宏块类型，已编码块类型，量化参数，参考帧索引，运动矢量以及残差数据。片层以上的语法单元用定长或者变长的直接二进制编码方式，片层及片层以下的语法单元及编码参数用变长编码（VLC）或者基于上下文的二进制算术编码（CABAC）。而 VLC 又包括统一的变长编码（UVLC）和基于上下文的变长编码（CAVLC）。

（5）去块效应滤波

运动估计和整数变换都以方块为单位，这种人为的分割造成了方块边界的不连贯性，会产生明显的方块效应，这对重建图像的主观质量有很大损伤。H. 264 通过在方块边界使用滤波器来消除方块效应。

有条件的滤波用于图像的所有 4×4 块的边界，除了图像的边界和特别指出不进行滤波的某些边界。滤波过程以宏块地址增加的顺序进行。在对每个宏块滤波时，可以使用当前宏块上方或左方已滤波的宏块值。

去块效应滤波分别用于亮度和色度分量。滤波过程中垂直边界的已改变的值作为同一宏块水平边界滤波的输入。去块效应滤波器用于编码器的反变换之后（重建和存储宏块做进一步预测之前）和解码器中重建和显示宏块之前。

（6）码率控制

对视频通信而言，由于通信信道带宽有限，需对视频编码码率进行控制，来保证编码码流的顺利传输和信道带宽的充分利用。针对不同的应用场合，学者们提出了多种码率控制（Rate Control）策略。

码率控制有两种模式：VBR 和 CBR，即可变比特控制和固定比特控制。VBR 模式是一种开环处理，输入为视频源和一个量化系数 QP 值。CBR 模式是一种闭环处理，输入为视频源和目标比特，它根据对源复杂度估计、解码缓冲的大小及网络带宽估计动态调整 QP，得到符合要求的码率，这也是真正意义上的码率控制。

H. 264 码率控制方法的提案主要有两个：JVT—F086 中 MPEG—2 TM5 改进版本及 JVT—G012 中提出用流量往返模型来分配每个基本单元目标比特数，并在宏块层编码采用二次率失真函数计算量化参数的算法。JVT—G012 还比较了

这两种算法,认为其算法优于 FO86 算法^[21]。

(7) SP/SI 帧

SP/SI 帧是 H.264 的特定编码帧,用于在视频流间高效切换,以便在解码时可以随意跳至某一时刻图像或同一视频内容在多码率传输过程中做流切换。当视频流的内容相同,编码参数不同时,采用 SP 帧;而当视频流的内容相差很大时,则采用 SI 帧将更加有效,它采用 4×4 帧内预测编码。

2.2 问题的提出

自出 H.264 出现以来,已经被应用到了许多产品当中,正如以上章节所述该算法有很多的优点,在实时视频监控、视频电话、网络视频等领域已经占有很重要的地位。现代计算机技术的迅速发展,使行业内对视频技术的研究过程中,往往忽略大的运算量,以及编码时间。在普通的环境中使用官方推出的 H.264 编码标准已经足够了,但是当环境对实时性的要求比较高,而处理器处理能力较差或者网络环境较差的情况下,编码时间的长短显的格外的重要。由于这个原因,很多研究人员都把精力投入到了对 H.264 算法的优化与改进当中。

从整个编码流程中我们可以看出主要的耗时部分集中在了变换与量化以及运动估计中。所以从已发表的文献中,我们可以看到很多的改进工作都集中在了这两个方面当中。本章主要讨论运动估计部分的改进,对于运动估计的改进工作在算法上主要集中在运动搜索部分,其切入点体现在提前结束搜索或者搜索区域的缩小。但是往往会在图像质量和编码速度上达不到平衡。往往是编码时间短的情况下,图像质量达不到要求。所以本文将探索一种在保证图像质量的情况下缩短运动估计时间的方法。

2.3 运动估计研究

在帧间预测编码中,由于活动图像邻近帧中的景物存在着一定的相关性。因此,可将活动图像分成若干块或宏块,并设法搜索出每个块或宏块在邻近帧图像中的位置,并得出两者之间的空间位置的相对偏移量。得到的相对偏移量就是通常所指的运动矢量,得到运动矢量的过程称为运动估计^[22]。

运动矢量和经过运动匹配后得到的预测误差共同发送到解码端,在解码端

按照运动矢量指明的位置，从已经解码的邻近参考帧图像中找到相应的块或宏块，和预测误差相加后就得到了块或宏块在当前帧中的位置^[23]。

通过运动估计可以去除帧间冗余度，使得视频传输的比特数大为减少，因此，运动估计是视频压缩处理系统中的一个重要组成部分。

2.3.1 运动表示法

一般的运动估计方法如下：设 t 时刻的帧图像为当前帧 $f(x,y)$ ， t' 时刻的帧图像为参考帧 $f'(x,y)$ ，参考帧在时间上可以超前或者滞后于当前帧，如图 2-8 所示。当 $t' < t$ 时，称之为后向运动估计，当 $t' > t$ 时，称之为前向运动估计。当在参考帧 t' 中搜索到当前帧 t 中块的最佳匹配时，可以得到相应的运动场 $d(x;t,t \pm \Delta t)$ ，即可得到当前帧的运动矢量^[24]。

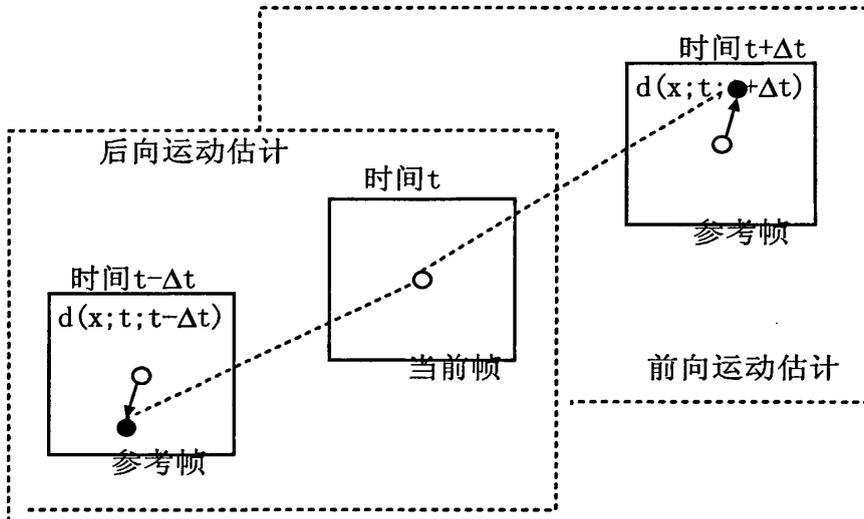


图 2-8 运动估计示意图

一般对于包含多个物体的景物，实际中普遍采用的方法是把一个图像帧分成多个块，使得在每个区域中的运动可以很好地用一个参数化模型表征，这称为块匹配法，即将图像分成若干个 $n \times n$ 块（典型值： 16×16 ）宏块，为每一个块寻找一个运动矢量 MV ，并进行运动补偿预测编码^[25]。

每一个帧间宏块或块都是根据先前已编码的数据预测出的，根据已编码的宏块、块预测的值和当前宏块作差值，该结果被传送给解码器，与解码器所需要的其他信息一起来重复预测过程。

每个分割区域都有其对应的运动矢量，并必须对运动矢量以及块的选择方式进行编码传输。在细节比较多的帧中如果选择较大的块尺寸，意味着用于表明运动矢量和侵害区域类型的比特数会少些，但是运动压缩的冗余度要多一些；如果选择小一点的块尺寸，那么运动压缩后冗余度要少一些，但是所需比特数要比较多。因此必须要权衡块尺寸对压缩效果的影响。一般对于细节比较少，比较平坦的区域选择块尺寸大一些，对于图像中细节比较多的区域选择块尺寸小一些^[26, 27]。

2.3.2 运动搜索算法

匹配误差函数可以用各种优化方法进行最小化，这就需要我们开发出高效的运动搜索算法，主要的几种算法归纳如下：

(1) 全局搜索算法

为当前帧的一个给定块确定最优位移矢量的全局搜索算法方法是：在一个预先定义的搜索区域内，把它与参考帧中所有的候选块进行比较，并且寻找具有最小匹配误差的一个。这两个块之间的位移就是所估计的MV，虽然全局搜索算法的效果是最好的，但是这样做带来的结果必然是极大的计算量。并且与这密切相关的便是搜索区域的选择和搜索步长的决定^[28]。

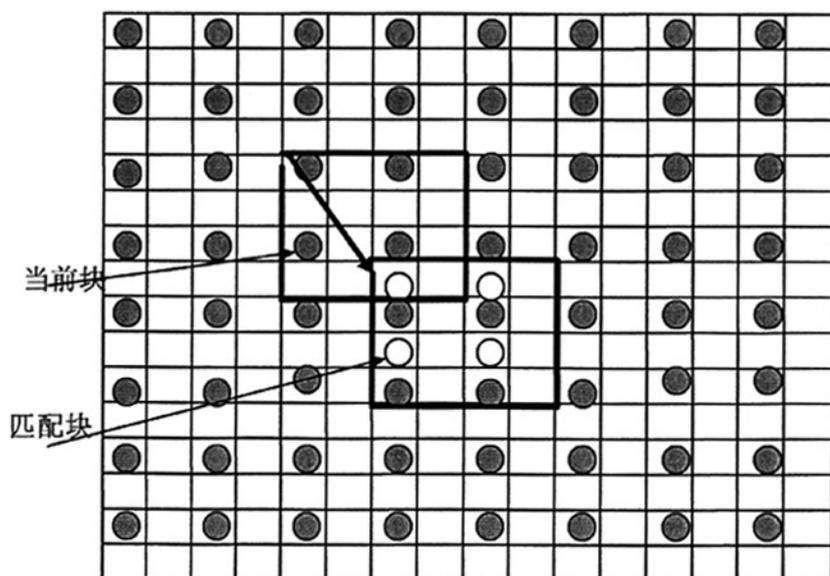


图 2-9 半像素精度块匹配示意图

选择搜索区域一般是相对于当前块对称的，左边和右边各有 R_x 个像素，上边和下边各有 R_y 个像素，如果已知在水平和垂直方向运动的动态范围是相同的，那么 $R_x=R_y=R$ 。估计的精度是由搜索的步长决定的，步长是相邻两个候选块在水平或者垂直方向上的距离。通常，沿着两个方向使用相同的步长。在最简单的情况下，步长是一个像素，称为整数像素精度搜索，该种算法也称为无损搜索算法。由于在全局块匹配算法中搜索相应块的步长不一定是整数，一般来说，为了实现 $1/K$ 个像素步长，对参考帧必须进行 K 倍内插。如图 2-9 所示为 $K=2$ 的例子，它被称为半像素精度搜索。根据实验证明，与整像素精度搜索相比，半像素精度搜索在估计精度上有很大提高，特别是对于低清晰度视频^[29]。

但是，应用分数像素步长，搜索算法的复杂性大大增加，例如，使用半像素搜索，搜索点的总数要比整数像素精度搜索大 4 倍以上。那么，如何确定适合的运动估计的搜索步长，对于视频编码的帧间编码来说，即使得预测误差最小化。经大量实验表明，在低噪声情况下，需要不大于 $1/8$ 像素的精度，而在高噪声情况下， $1/2$ 像素精度就够了。

(2) EPZS 算法研究

EPZS(Enhanced Predictive Zonal Search) 是一种预测搜索算法，该算法从几个非常可能的矢量中预测最佳运动矢量，采用了更多的预测矢量选择和多模搜索路径^[30]。

预测点的选择对算法的性能非常关键，精确选择对运动估计性能有重要影响的预测矢量是 EPZS 的主要特征。这些预测矢量主要根据序列的相关性来选择，包括空间和时间相关性，或在搜索窗内的一些固定位置。EPZS 定义了一个矢量集 S ，按照重要性分成 A、B、C 等 3 个子集，其中：

$A=\{\text{Median Predictor}\}$; 中值矢量，在预测矢量中是最重要的预测值；

$B=\{\text{Collocated Block MV, Left MV, Top MV, Top-left MV, Top-right MV, (0,0)}\}$ ，包括时域相邻的同位置块的运动矢量和相邻块的矢量以及 $(0,0)$ ；

$C=\{\text{Temporal scaled MV, Accelerator MV, Collocated neighbor MV, Reference scaled MV, Inter-mode MV, Sampled pixels}\}$ ；考虑到较大运动的情况下可能的预测矢量，附加预测点的引入会增加运动估计过程的复杂度，因此在算法中引入一种称之为“三阶段处理程序”的过程^[31]：

1) 检测完子集 A 中的预测块，判断当前块的 SAD 值是否满足阈值 $T1$ ，如果满足，终止搜索，否则继续；

2) 检测完子集B中的预测块, 如果当前块的最小SAD值满足自适应计算的阈值T2 则终止, 否则继续;

3) 检测子集C 中的预测块, 如果当前块的最小SAD值满足阈值T3 则终止, 否则用更精细的模式搜索。

EPZS 算法使用3 种搜索模板, 包括小钻石(DS)、正方形EPZS和扩展EPZS, 如图2-10所示:

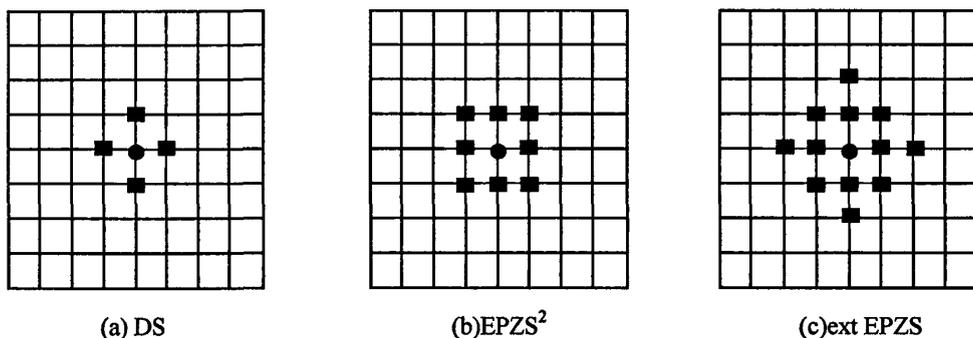


图2-10 EPZS的三种搜索模板

小钻石搜索对于相对复杂的序列很容易陷入局部最优, 而后两种搜索模式则具有较好的搜索性能。EPZS采用以最佳预测矢量为中心按扩展EPZS (extEPZS) 搜索和以中值预测矢量MVP为中心按EPZS²搜索的双模式搜索。

EPZS 算法由局部开始逼近最佳点, 该算法虽然可以较好地避免落入局部最优, 但是仍有其局限性, 即对于搜索范围比较小、运动比较平缓的情况下效果较好, 但在搜索范围较大和运动剧烈的情况下, 很容易在搜索初期就落入局部最优。因此如何避免落入局部最优, 在较大范围内找到最优的匹配点, 仍然是运动估计研究的重点^[32]。

(3) UMHexagonS 算法研究

由于物体的运动千变万化, 很难用一种简单的模型去描述, 也很难用一种单一的算法来搜索最佳运动矢量, 因此实际上大多采用多种搜索算法相组合的办法, UMHexagonS (非对称十字型多层次六边形格点搜索算法) 综合了多种搜索模式, 并利用了分等级运动矢量搜索, 在很大程度提高预测的有效性和健壮性^[33]。UMHexagonS搜索过程如图2-11所示:

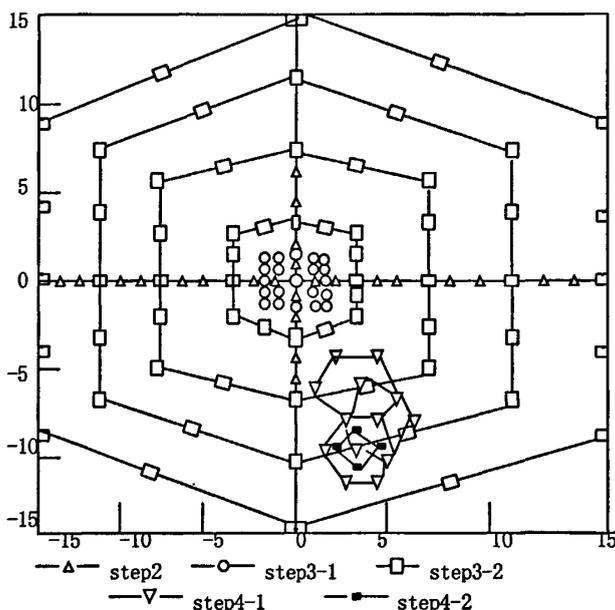


图 2-11 非对称十字型多层次六边形格点运动搜索

具体搜索方法的步骤如下^[34-39];

Step1:搜索过程的起始搜索点搜索

在起始搜索矢量的集合 S 中搜索一个对应费用函数值最小的候选运动矢量作为起始搜索点, 费用函数表达式如下:

$$m_{\min} = \arg[\min_{m_i} J(m_i, \lambda_{MOTION})], s, t, m_i \in S$$

然后判断是否提前截止 Early_Termination。

Step2:不对称的十字交叉搜索

由于自然界物体运动水平方向要比垂直方向剧烈一些, 采用非对称的十字形搜索方法, 搜索点集合为:

$$\Omega = \{m = (m_x, m_y)^T \mid m = (cm_x \pm 2i, cm_y)^T, i = 0, 1, 2, \dots, \frac{W}{2}; m = (cm_x, cm_y \pm 2j)^T, j = 1, 2, \dots, \frac{W}{4}\} cm = m_{\min}$$

其中, W 为搜索区。

搜索步骤如下:

- (1) $m_{\min 2} = \arg[\min_{m_i} J(m_i, \lambda_{MOTION})], s, t, m_i \in \Omega_1$
- (2) $m_{\min} = \arg[\min(J(m_{\min}, \lambda_{MOTION}), J(m_{\min 2}, \lambda_{MOTION}))]$
- (3) Early_Termination

Step3:非均匀多层次六边形格点搜索, 步步分成两个子步骤:

Step3-1:小矩形窗全搜索

定义候选运动矢量集合:

$$\Omega_2 = \{m = (m_x, m_y)^T \mid |m_x - cm_x| \leq 2, |m_y - cm_y| \leq 2\}, cm = m_{\min}$$

然后搜索步骤有:

$$m_{\min 3} = \arg[\min_{m_i} J(m_i, \lambda_{MOTION})], s, t, m_i \in \Omega_2$$

$$m_{\min} = \arg[\min(J(m_{\min}, \lambda_{MOTION}), J(m_{\min 3}, \lambda_{MOTION}))]$$

Early_Termination

Step3-2:扩展的多层次六边形格点搜索

六边形的 16 个点为:

$$\Omega_{16-HP} = \{m = (x, y)^T \mid m = (\pm 4, \pm 2)^T, (\pm 4, \pm 1)^T, (\pm 4, 0)^T, (\pm 2, \pm 3)^T, (0, \pm 4)^T\}$$

采用下列办法扩展搜索区:

For(k=1;k<W/4;k++)

{

$$\Pi_k = \{m = (m_x, m_y) \mid m_x = cm_x + kg \cdot x', m_y = cm_y + kg \cdot y', (x', y') \in \Omega_{16-HP}\}$$

Then:

$$m_{\min k} = \arg[\min_{m_i} J(m_i, \lambda_{MOTION})], s, t, m_i \in \Pi_k$$

$$m_{\min} = \arg[\min(J(m_{\min}, \lambda_{MOTION}), J(m_{\min k}, \lambda_{MOTION}))]$$

Early_Termination;

}

Step4-1:扩展的六边形搜索, 分成两个子步骤:

$$\Omega_3 = \{m = (m_x, m_y)^T \mid m = (cm_x \pm 2, cm_y)^T, (cm_x \pm 1, cm_y \pm 2)^T\}, cm = m_{\min}$$

然后有:

$$m_{\min 4} = \arg[\min_{m_i} J(m_i, \lambda_{MOTION})], s, t, m_i \in \Omega_3$$

$$m_{\min} = \arg[\min(J(m_{\min}, \lambda_{MOTION}), J(m_{\min 4}, \lambda_{MOTION}))]$$

如果 $m_{\min} = cm$, 跳转到 Step4-2;

否则, 跳转至 Step4-1。

Step4-2:基于菱形模块的搜索

这一步骤的基本搜索模块为图 2-11 中所示的菱形模块, 其集合为:

$$\Omega_4 = \{m = (m_x, m_y)^T \mid m = (cm_x \pm 1, cm_y)^T, (cm_x, cm_y \pm 1)^T\}, cm = m_{\min}$$

然后开始搜索步骤如下:

$$m = \arg[\min Cost(m, \lambda_{MOTION})], m \in \Omega_4$$

如果 $m_{\min} = cm$, 搜索截止;

否则跳转至 Step4-2。

2.4 算法改进

上面的章节我们对 H.264 编码标准的整个编码、解码流程进行了深入的研究，并且对其中的关键的技术进行了研究，从这些内容之中我们充分的认识到了运动估计算法的好坏对整个压缩标准的影响，所以对运动估计的内容从理论到现有的比较优秀的已经被 H.264 标准采用了的几种算法进行了深入仔细的研究，接下来我们将对运动估计算法提出本文的改进方案。

2.4.1 改进算法的基础

经过上面对几种常用的算法的介绍，可以从中发现，运动估计的搜索区域、搜索步长、搜索步骤决定了该算法的编码效果的好坏，以及耗时的长短，效果最为优良的全搜索算法，由于其搜索区域及搜索方法的原因，造成了非常大的运算量的问题，这使得在很多情况下都被限制，无法使用。由其对于可视电话、视频会议等图像的运动量通常很小领域，最佳点的位置本来就在起始搜索点附近，采用大范围搜索，在提高画面质量不明显的环境下反而大大加剧了图像搜索的时间^[40-42]。因此，根据图像的运动特性，把待编码的图像块和参考帧中相同位置相同大小的图像块进行差值计算，把计算得到的结果和预先设定的阈值进行比较，根据比较的结果，对图像进行编码。

基于上面的想法，对 H.264 标准已采用的几种算法进行了比较，最后发现在编码效果得到保证的前提下，UMHexagons 算法具有很好的编码效果，所以本文将在该算法的详细研究基础上，对该算法进行一定的优化工作。UMHexagons 算法的编码流程图如图 2-12 所示^[43,44]。

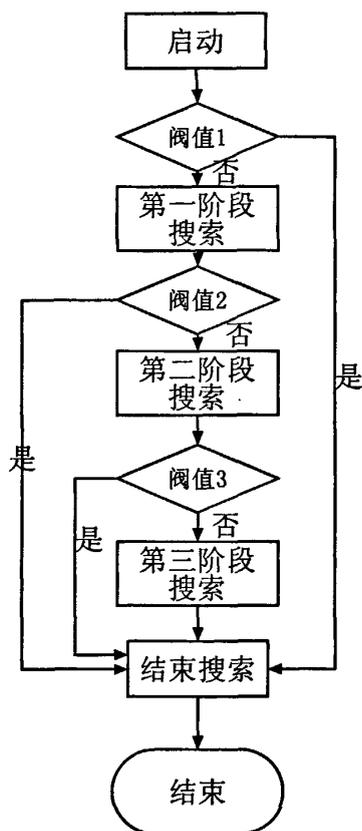


图 2-12 UMHexagons 算法的编码流程

2.4.2 优化方法

优化工作主要可以分成两个大的方面，即算法级的优化和代码级优化，算法级的优化占有主导地位，效果也最显著，本文将先从算法级优化该算法。由于在该算法的搜索过程中，相对于运动幅度小的图像来说，在第一阶段已经满足条件，这将大大的节省了搜索的时间，第一阶段未能结束的进入第二阶段也并不是很费时，基本上图像搜索都能在这一步截止。当这一步无法截止的情况下，进入下面的步骤之后运算量将大幅度的增加。而且在接下去的搜索过程中，必须得一次全部搜索结束之后才开始判断，这样对于在中途已经获得最佳预测的图像来说是非常浪费时间的。

所以接下来我们将主要从两个方面来改进该算法：

第一思路是在获得理想的效果之后，提前截止搜索，这一步骤我们将通过在每一个搜索环节中，自适应的设定阈值，并且在每次搜索的过程当中将得出值与该阈值进行比较，如果符合条件将提前截止搜索，这样将大大的节省搜索时间，尤其是对于进入第二阶段之后的搜索过程，意义是非常重大的。

第二思路是通过搜索范围及方法的改变来减少搜索时间，由于图像的水平活动较垂直方向活动要多一些，并且在前两个阶段的搜索非常快，所以我们将在前两阶段并未得到理想结果的情况下，加入一个横向较大的矩形搜索步骤。

本文所提的改进后算法的搜索过程如图 2-13 所示：

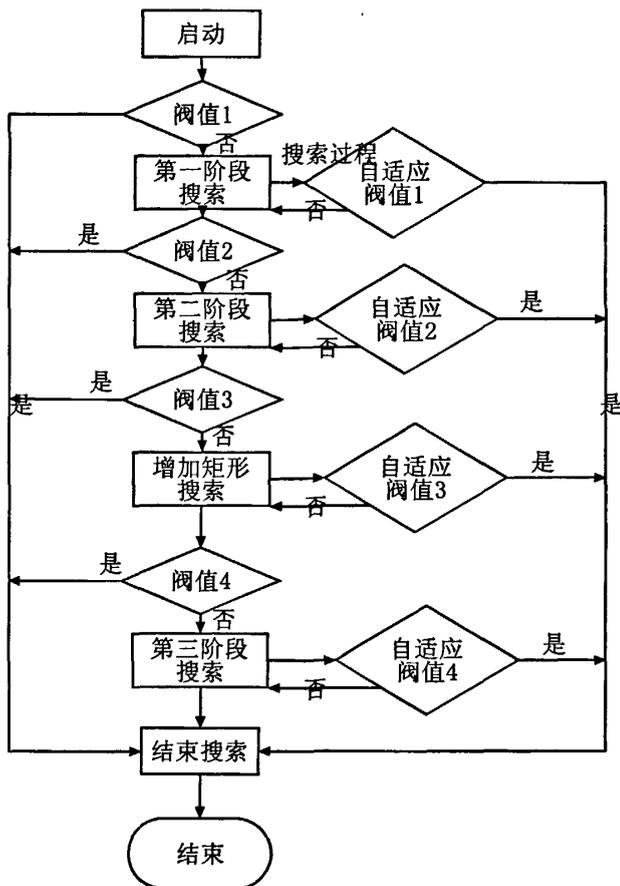


图 2-13 改进算法搜索流程图

通过以上的改进该算法已经在编码速度上有了较大的提高，由于官方版本中所发布的代码比较零乱，在数据结构的组织上并不合理，所以我们通过对代码的优化也能得到一定的提高，本文将从以下两个方面来做代码方面的优化：

(1) 优化数据结构。对于象素的具体位置一定要分开计算，特别是像整象素等情况，对于大量地乘法运算则可以用查找表来代替。

(2) 块效应的优化。针对开辟和释放动态内存，可以考虑一开始一次性开辟，而后再用数组等标识的方法达到静态分配；对于不停地更新参考帧的办法是具体情况具体分析，也就是到了需要时再更新。

2.5 本章小结

本章对 H.264 算法进行了详细的研究,对其中的几项关键的技术进行了细致的学习。最后对运动估计算法进行了深入的研究,对其中几种常用的并且已经被 H.264 采用的算法进行了深入研究,着重对 UMHexagonS 搜索算法进行了论述,并且在该算法的基础上提出了改进想法,在第四章我们将对该改进进行实验论证,经过实验我们可以看出来,经改进的算法在保证图像质量的情况下,比该算法在编码速度方面有一个较大的进步。本文所改进后的算法与以原版本的算法相比较有以下优点:

(1) 在运动估计的过程当中,在每一个搜索阶段进行搜索的过程中,并不一定全部搜索完毕才判断,而是设备自适应阈值,在搜索的过程中当符合该条件时即跳出搜索进入结束搜索阶段,通过这种方法在一个搜索过程的开始便得到理想的情况下,大大的节约了搜索时间。

(2) 在搜索进入比较费时的第三阶段之前,对该算法增加一个矩形搜索过程,由于生活中的事物大部分是水平运动更剧烈一些,所以在该横向较大的矩形搜索过程中,很多情况下可以避免进入接下来的过程,也大大的节约了编码时间。

第 3 章 H.264 整数变换与量化算法的优化

3.1 引言

绝大多数图像都有一个共同的特征：平坦区域和内容缓慢变化的区域占据一幅图像的大部分，而细节区域和内容突变区域则占小部分。也可以说，图像中直流和低频区占大部分，高频区占小部分。这样，如果将空间区域的图像变换到频域或所谓的变换域，这就会产生相关性很小的一些变换系数，并可对其进行压缩编码，即所谓的变换编码。变换中有一类叫做正交变换，可用于图像编码。自 1968 年利用快速傅立叶变换 (FFT) 进行图像编码以来，出现了多种下次变换编码，如 K-L 变换、离散余弦变换 (DCT) 等等。其中，编码性能以 K-L 变换编码最理想，但缺乏快速算法，且变换矩阵随图像而异，不同图像需计算不同的变换，因而只用来作参考比较。DCT 编码性能最接近于 K-L 变换，略次而已，但它具有快速算法，广泛应用于图像编码^[45]。

3.1.1 K-L 变换

设一图像阵列每行由列矢量表示，即 $\mathbf{x}=(x_0, x_1, \dots, x_{N-1})^T$ 。由于图像内容的千变万化该列矢量是随机矢量，用于平均值和协方差表征。

\mathbf{X} 的列矢量平均值 $\bar{\mathbf{X}}$ 为：

$$\bar{\mathbf{X}} = E[x_0, x_1, \dots, x_{N-1}]^T = [\bar{x}_0, \bar{x}_1, \dots, \bar{x}_{N-1}]^T \quad (3.1)$$

\mathbf{X} 的协方差矩阵定义为：

$$\mathbf{C}_x = E\{(\mathbf{X} - \bar{\mathbf{X}})(\mathbf{X} - \bar{\mathbf{X}})^T\} = \begin{pmatrix} c_{00} & \cdots & c_{0,N-1} \\ \vdots & \ddots & \vdots \\ c_{N-1,0} & \cdots & c_{N-1,N-1} \end{pmatrix} \quad (3.2)$$

上式中 $c_{ij} = E\{(x_i - \bar{x}_i)(x_j - \bar{x}_j)^T\}$ ，表示随机变量 x_i 和 x_j 之间的相关程度的协方差（相关系数）。协方差矩阵对角线上的元素是各像素的方差 $e_{ii} = \sigma_i^2$ ，表示该像素的能量大小。

不难求出 \mathbf{X} 的协方差矩阵 \mathbf{C}_x 的特征值及其归一化特征矢量。利用这些特征

矢量构成正交矩阵 A ，则图像 X 作正交变换后得：

$$Y=AX \quad (3.3)$$

该变换便是 K-L 变换，也叫做 Hotelling 变换，可以证明， Y 的协方差矩阵是一个对角矩阵，其对角线上元素便是 C_x 的特征值。

$$C_Y = \begin{pmatrix} \lambda_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \lambda_N \end{pmatrix} \quad (3.4)$$

这说明 K-L 变换后， Y 中的相关性已全部消除，能量只集中在 n 个特征值 λ_i 上，其值按从大到小排列。因此编码只需传送前 n 个特征值，大大降低了码率。可以证明，在均方误差最小准则下，K-L 变换是失真最小的变换。但 K-L 变换不易实现，只是一种理想的变换方法^[46]。

3.1.2 离散余弦变换 DCT

由于 H.264 采用的是整数 DCT 变换，所以本文主要的研究将是该变换，虽然整数 DCT 变换不同于普通的 DCT 变换，但是其基础依然是从这里得来，所以现在我们将深入的研究 DCT 变换，对 DCT 变换的研究我们将从一个由浅入深的过程来研究，

(1) 一维 DCT 变换

DCT 正变换时，设 $f(x) = [f(0), f(1), \dots, f(N-1)]^T$ ，对 f 变换后得：

$$F(u) = \sqrt{\frac{2}{N}} c(u) \sum_{x=0}^{N-1} f(x) \cos\left[\frac{\pi}{2N}(2x+1)u\right] \quad (3.5)$$

$u=0,1,\dots,N-1$

$$\text{其中, } c(u) = \begin{cases} 1/\sqrt{2}, u=0 \\ 1, u=1,2,\dots,N-1 \end{cases}$$

可写成矩阵形式，

$$F = A \bullet f$$

$$A = \sqrt{2/N} \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} & \cdots & 1/\sqrt{2} \\ \cos \frac{\pi}{2N} & \vdots & & \vdots \\ \vdots & & & \\ \cos \frac{(N-1)\pi}{2N} & \cdots & \cos \frac{(2N-1)(N-1)\pi}{2N} \end{pmatrix} \quad (3.6)$$

对应地，一维离散余弦反变换 IDCT 为

$$f(x) = \sqrt{\frac{2}{N}} \sum_{u=0}^{N-1} c(u) F(u) \cos\left[\frac{\pi}{2N}(2x+1)u\right] \quad (3.7)$$

$$x = 0, 1, 2, \dots, N$$

由于 $A^{-1}=A^T$, f 可写成如下矩阵形式:

$$f = A^{-1}F = A^T F \quad (3.8)$$

(2) 二维 DCT 变换

令 $f(x,y)$ 为 $N \times N$ 离散图像序列，二维 DCT 变换表示为:

$$F(u,v) = \frac{2}{N} c(u)c(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cos\left[\frac{\pi}{2N}(2x+1)u\right] \cos\left[\frac{\pi}{2N}(2y+1)v\right]$$

$$u, v = 0, 1, \dots, N-1 \quad (3.9)$$

二维 IDCT 为:

$$f(x,y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} c(u)c(v) F(u,v) \cos\left[\frac{\pi}{2N}(2x+1)u\right] \cos\left[\frac{\pi}{2N}(2y+1)v\right]$$

$$x, y = 0, 1, \dots, N-1 \quad (3.10)$$

也可写成矩阵形式:

$$F = AfA^T \quad (3.11)$$

$$f = A^T F A \quad (3.12)$$

对于图像变换编码，最理想的变换操作应对整个图像进行，以便去除所有

像素间的相关性。但这样的操作计算量太大。实际上，往往把图像分为若干块，以块为单位进行 DCT 变换^[47]。

3.1.3 整数变换及与量化

H.264 对图像或预测残差采用了 4×4 整数离散余弦变换技术，避免了以往标准中使用的通过 8×8 离散余弦变换、逆变换经常出现的失配问题。量化过程根据图像动态范围的大小来确定量化参数，即保留了图像中必要的细节，又可减少码流。在图像编码中，变换编码和量化从原理上讲是两个独立的过程。但在 H.264 中，将两个过程中的乘法合二为一，并进一步采用整数运算，减少了编解码的运算量，提高了图像压缩的实时性。这些措施对峰值信噪比 (PSNR) 的影响很小，一般低于 0.2DB，可忽略不计。H.264 中整数变换及量化的具体过程如图 3-1 所示：

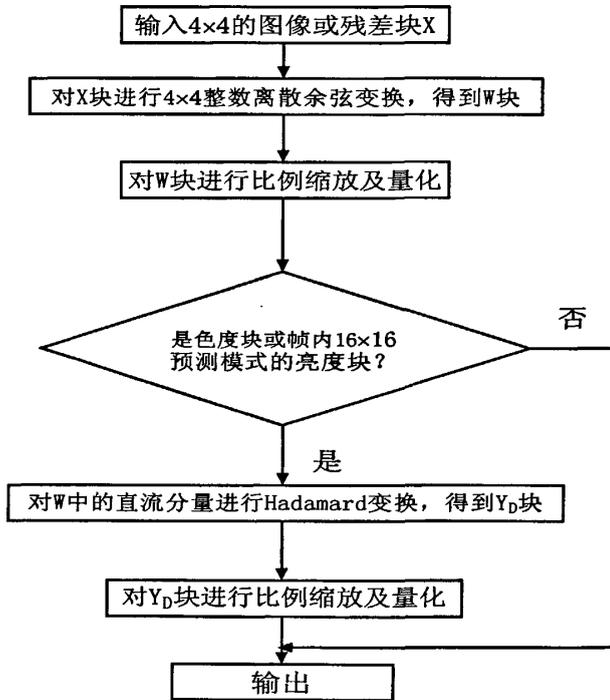


图 3-1 编码器中变换及量化过程

其中，如果输入块是色度块或帧内 16×16 预测模式的亮度块，则将宏块中各 4×4 块的整数余弦变换的直流分量组合起来再进行 Hadamard 变换，进一步压缩其码率。下面将对整数变换及量化分别进行研究^[48]。

(1) 整数变换

一维 N 点离散余弦变换 (DCT) 可以表示为:

$$y_k = C_k \sum_{n=0}^{N-1} x_n \cos \frac{(2n+1)k\pi}{2N} \quad (3.13)$$

其中, x_n 是输入时域序列中的第 n 项, y_k 是输出频域序列中的第 k 项, 系数 C_k 定义如下:

$$C_k = \begin{cases} \sqrt{\frac{1}{N}} & k=0 \\ \sqrt{\frac{2}{N}} & k=1, 2, \dots, N-1 \end{cases} \quad (3.14)$$

每个 DCT 系数 y_k 确定信号 x_n 在相应频率点上的贡献。最低的系数 (即 $k=0$) 为 DCT 系数, 代表信号的平均值, 也称为直流分量。其他系数称 AC 系数, 与递增的较高频率相联系。

一维 N 点离散余弦逆变换 (IDCT) 可以表示为:

$$x_n = \sum_{k=0}^{N-1} C_k y_k \cos \frac{(2n+1)k\pi}{2N} \quad (3.15)$$

二维 $N \times N$ 图像块的 DCT 可以理解为先对图像块的每行进行一维 DCT, 然后对经行变换的块的每列再应用一维 DCT。可以表示为:

$$Y_{mn} = C_m C_n \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X_{ij} \cos \frac{(2j+1)n\pi}{2N} \cos \frac{(2i+1)m\pi}{2N} \quad (3.16)$$

$$X_{ij} = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} C_m C_n Y_{mn} \cos \frac{(2j+1)n\pi}{2N} \cos \frac{(2i+1)m\pi}{2N} \quad (3.17)$$

其中, X_{ij} 是图像块 X 中第 i 行第 j 列图像或其残差值, Y_{mn} 是变换结果矩阵 Y 相应频率点上的 DCT 系数。可以用矩阵表示:

$$Y = AXA^T \quad (3.18)$$

$$X = A^T Y A \quad (3.19)$$

其中, $N \times N$ 变换矩阵 A 中的系数:

$$A_j = C_i \cos \frac{(2j+1)i\pi}{2N} \quad (3.20)$$

H.264 对 4×4 的图像块（亮度块或 Cr、Cb 色度块）进行操作，则相应的 4×4 DCT 变换矩阵 A 为：

$$A = \begin{pmatrix} \frac{1}{2} \cos(0) & \frac{1}{2} \cos(0) & \frac{1}{2} \cos(0) & \frac{1}{2} \cos(0) \\ \sqrt{\frac{1}{2}} \cos(\frac{\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{3\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{5\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{7\pi}{8}) \\ \sqrt{\frac{1}{2}} \cos(\frac{2\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{6\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{10\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{14\pi}{8}) \\ \sqrt{\frac{1}{2}} \cos(\frac{3\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{9\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{15\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{21\pi}{8}) \end{pmatrix} \quad (3.21)$$

设 $a = \frac{1}{2}$, $b = \sqrt{\frac{1}{2}} \cos(\frac{\pi}{8})$ 及 $c = \sqrt{\frac{1}{2}} \cos(\frac{3\pi}{8})$, 则：

$$A = \begin{pmatrix} a & a & a & a \\ b & c & -c & b \\ a & -a & -a & a \\ c & -b & b & -c \end{pmatrix} \quad (3.22)$$

A 中的 a、b 和 c 是实数，而图像块 X 中的元素是整数。对实数 DCT，由于在解码端的浮点运算精度问题，会造成解码后的数据失配，进而引起漂移。H.264 较其它图像编码使用了更多的预测过程，甚至内部编码模式也依赖于空间预测。因此，H.264 对预测漂移是十分敏感的。为此，H.264 对 4×4 DCT 中的 A 进行了改造，采用整数 DCT 技术，可有效地减少计算量，同时不损失图像的准确度。公式(3.18)可以等效为：

$$\begin{aligned}
 Y &= (CXC^T) \otimes E \\
 &= \left(\left(\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & d & -d & -1 \\ 1 & -1 & -1 & 1 \\ d & -1 & 1 & -d \end{pmatrix} X \begin{pmatrix} 1 & 1 & 1 & d \\ 1 & d & -1 & -1 \\ 1 & -d & -1 & 1 \\ 1 & -1 & 1 & -d \end{pmatrix} \right) \otimes \begin{pmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{pmatrix} \right) \\
 &\quad (3.23)
 \end{aligned}$$

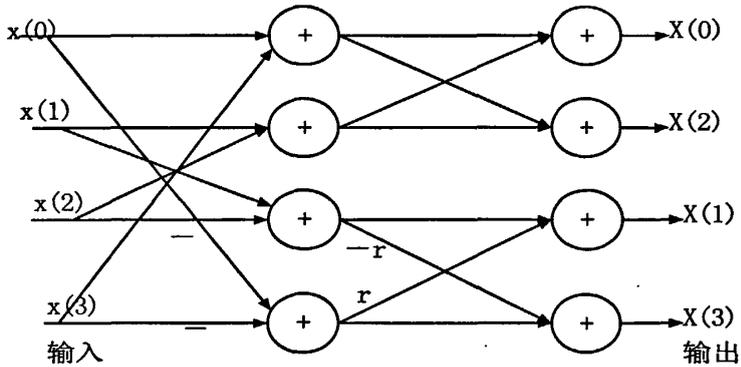
其中, $d = c/b (\approx 0.414)$ 。符号“ \otimes ”表示 (CXC^T) 结果中的每个元素乘以矩阵 E 中对应位置上的系数值的运算。为了简化计算, 取 d 为 0.5。同时为了保持变换的正交性, 对 b 进行修正, 取 $b = \sqrt{\frac{2}{5}}$ 。对矩阵 C 中的第 2 行和第 4 行, 以及矩阵 C^T 中的第 2 列和第 4 列元素乘以 $\sqrt{2}$, 相应的改造 E 为 E_f , 以保持公式(3.23)成立, 得到:

$$\begin{aligned}
 Y &= (C_f X C_f^T) \otimes E_f = \\
 &\left(\left(\begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix} X \begin{pmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{pmatrix} \right) \otimes \begin{pmatrix} a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \\ a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \end{pmatrix} \right) \\
 &\quad (3.24)
 \end{aligned}$$

其中, 运算“ \otimes ”对每个矩阵元素只进行一次乘法, 同时它将被归纳到量化运算中。这样, $(C_f X C_f^T)$ 中只剩下整数的加法, 减法和移位(乘以 2)运算。被称为整数 DCT 的公式(3.24)与通常 DCT 运算结果近似, 但因为 b 和 d 的值有所变化, 所以两者的结果有差别。

H.264 将矩阵乘法运算改造成两次一维整数 DCT 变换, 例如先对图像或其残差块的每行进行一维整数 DCT, 然后对经行变换的块的每列再应用一维整数 DCT。而每次一维整数 DCT 可以采用蝶形快速算法, 以节省运算时间, 如图 3-2

所示：



其中， $r=2$ ：整数 DCT 变换； $r=1$ ：Hadamard 变换

图 3-2 一维快速变换算法

(2) 量化

量化过程在不降低视觉前提下减少图像编码长度，减少视觉恢复中不必要的信息。H.264 采用标量量化技术，它将每个图像样点编码映射成较小的数值。一般标量量化器的原理为：

$$FQ = \text{round}\left(\frac{y}{QP}\right) \quad (3.25)$$

其中， y 为输入样本点编码， QP 为量化步长， FQ 为 y 的量化值， $\text{round}()$ 为取整函数（其输出值为与输入实数最近的整数）。其相反过程，即反量化为：

$$y' = FQ \cdot QP \quad (3.26)$$

在量化和反量化过程中，量化步长 QP 决定了量化器的编码压缩率及图像精度。如果 QP 比较大，则量化值 FQ 的动态范围较小，其相应的编码长度较小，但反量化时损失较多的图像细节信息；如果 QP 比较小，则 FQ 的动态范围较大，相应的编码长度也较大，但图像细节信息损失较少。编码器根据图像值实际的动态范围自动改变 QP 值，在编码长度和图像精度之间折衷，达到整体最佳效果。

在 H.264 中，量化步长 $Qstep$ 共有 52 个值。当 QP 取最小值 0 时，代表最精细的量化，当 QP 取最大值 51 时，代表最粗糙的量化。 QP 每增加 6， $Qstep$ 增加一倍。应用时可以在这个较宽的经步长范围内根据实际需要灵活选择。

在 H.264 在，量化过程是对前面 (3.24) 的 DCT 结果进行操作：

$$Z_{ij} = \text{round}\left(\frac{Y_{ij}}{Qstep}\right) \quad (3.27)$$

其中, Y_{ij} 是矩阵 Y 中的转换系数, Z_{ij} 是输出的量化参数, $Qstep$ 是量化步长。

H.264 量化过程还要同时完成 DCT 变换中 “ $\otimes E_f$ ” 乘法运算, 它可以表述为:

$$Z_{ij} = \text{round}\left(W_{ij} \frac{PF}{Qstep}\right) \quad (3.28)$$

其中, W_{ij} 是矩阵 W 中的转换系数, PF 是矩阵 E_f 中的元素, 根据样本点在图像块中的位置 (i,j) 取值:

$$PF = \begin{cases} a^2 & (0,0), (2,0), (0,2) \text{ 或 } (2,2) \\ b^2/4 & (1,1), (1,3), (3,1) \text{ 或 } (3,3) \\ ab/2 & \text{其他} \end{cases} \quad (3.29)$$

利用量化步长随量化参数每增加 6 而增加一倍的性质, 可以进一步简化计算。设:

$$qbits = 15 + \text{floor}(QP/6) \quad (3.30)$$

$$\text{令 } MF = \frac{PF}{Qstep} 2^{qbits}$$

其中, $\text{floor}()$ 为取整函数。则公式(3.27)可以写成:

$$Z_{ij} = \text{round}\left(W_{ij} \frac{MF}{2^{qbits}}\right) \quad (3.31)$$

3.2 问题的提出

综合以上的论述, 我们可以看出, 为了得到最佳的编码效果, 往往采用全部编码的方式, 但是这样虽然得到了好的编码效果, 却在编码时间上有了很大的影响, 如果对于将要进行变换和编码的系数块, 在做变换域编码之前, 可以先进行全零块检测, 如果能够事先检测为全零块, 就可以避免相应的变换域编码和量化, 从而可以极大地降低变换和编码时间。

在量化的过程中, 由于有多种选择的余地, 51 的参数选择, 对编码的速度有非常大的影响, 官方版本选用了 28 这个参数, 然而对于如监控之类的变化不太大的场合, 该参数并不是最佳选择, 我们依然需要对其进行大量实验来选

取一个更加合适的量化参数。

针对以上两个比较重要的问题，本章将对这两个方面进行一定的改进，通过实验确定了一个全零块检测的阈值，加入全零检测步骤，并且通过对 QP 参数的实验，确定了一个较理想的值。从而可以大大的加快编码的速度。

3.3 算法的改进

对该部分算法的改进我们将从两个步骤来优化，第一个步骤，在参考文献[49]中提出了一种全零块检测的算法^[49]，该算法对编码速度的提高有很大的帮助，本文将先对该算法进行深入研究。在算法中加入非零块的检测，对非零块不进行编码，并且在其基础上进行阈值及搜索模式的改进。第二个步骤，我们将提出 QP 值选取的实验步骤及方法。

3.3.1 全零块检测方法

下面就变换和量化为全零的块，分析在变换之前数据的最大值，把此最大值作为判决门限。

根据前面的变换量化公式我们可以推出，如果所有的系数满足：

$$Y_{ij} = \left| W_{ij} \times \frac{MF}{2^{qbits}} \right| < 1 \quad 0 \leq i, j \leq 3 \quad (3.32)$$

那么就会有下式成立：

$$|LEVEL_y| = 0 \quad 0 \leq i, j \leq 3 \quad (3.33)$$

其中， $LEVEL_y$ 为量化后的系数，如果 $LEVEL_y$ ($0 \leq i, j \leq 3$) 都为零，那么该系数块就为全零。从而有：

$$\begin{aligned} |Y_{ij}| &= \left| W_{ij} \times \frac{MF}{2^{qbits}} \right| \\ &\leq 4 \sum_{u=0}^3 \sum_{v=0}^3 |x_{uv}| \times \left| \frac{MF}{2^{qbits}} \right| < 1 \quad 0 \leq i, j \leq 3 \end{aligned} \quad (3.34)$$

于是得到如下判决式：

$$\sum_{u=0}^3 \sum_{v=0}^3 |x_{uv}| < \frac{1}{4} \left| \frac{MF}{2^{qbits}} \right| = \frac{2^{qbits-2}}{MF} \quad (3.35)$$

只要 4×4 块中的数据满足公式(3.35)，就可判决当前块为全零块，其变换域

编码和量化运算都可以消除了。由于公式(3.35)正好是绝对误差(SAD)的值，由于在运动搜索时已经计算过该值，而且在 H.264 中，进行运动搜索块的大小可以是 16×16 、 16×8 、 8×8 、 8×4 、 4×4 ，所以全零块的判决式如下：

$$SAD \leq M \cdot N \cdot 2^{qbits-6} / MF \quad (3.36)$$

其中 M、N 为 16、8、4

根据以上分析，结合运动搜索的全零块检测方法为，在运动搜索时，加入停止搜索条件，对某个矢量计算出来的 SAD 值进行的全零块检测，即判决门限为公式 (3.36)，如果满足，就停止宏块的运动搜索，并且其后的整数变换和量化也都省略掉。本文中结合以上公式，对该值进行了实验探索，最终选取一个合适的固定值作为该阈值，省去了每次对该式的计算过程。

3.3.2 量化参数优化

由于量化参数可以选取的值共有 52 个，而且该参数的选取，对编码的速度以及图像的质量都有很大的影响，所以选取一个合适的参数对编码将有很大的帮助。凭借对图像质量判决的两咱方式，一种方式是视觉判决，直接通过肉眼看来判决，另一种是根据 PSNR 等主要的参数进行判决。本文将通过以下步骤来完成该参数的选择：

(1) 对单幅图像，选取不同的参数值来进行实验，主要观察实验数据 PSNR 值，以及编码速度，选取几个对速度有利，而又对 PSNR 值影响不是很大的参数值。

(2) 对单幅图像进行编码实验，在不同的参数值所编码出的图像之间进行肉眼的对比，选取一个效果较好，可以接受的图像质量，然后在该质量前提下，对参数值根据编码速度进行选取。

(3) 对多幅具有不同特性的图像进行以上两步实验，观察是否可以达到满意的结果。

3.4 总结

在编码的过程中，变换与量化的过程所占据的时间很大，对该部分算法速度的提高，将在很大程度上提高该整体编码的速度，本章通过算法上的改进，对整体的编码速度有了很大的提高，具体的实验验证，将在第四章选取典型的

例子进行展示。

本章通过对原算法进行改变，比原算法，以及参考文献中用到的算法在效果上都有较大的改进，该算法的主要特点有以下几点：

(1) 通过对变换与量化加入了全零块的判决条件，大大的减少了编码的工作量，缩短了编码时间。

(2) 在理论研究的前提下，根据所推理公式，最后对阈值进行了实验探索，设定了该部分判决的固定阈值，节省了每次运算的时间。

(3) 由于日常生活中所监控的内容以及视频电话等的应用中，画面的变化并不是很大，所以选取大量变化幅度不是很大的视频进行实验，最后得获得了一个较理想的量化参数 QP 值，在保证画面质量可以接受的前提下，加快了编码的速度。

第 4 章 实验结果及分析

针对本文的研究目标，本章所进行的实验数据研究将主要围绕两个大的方面，即编码后视频的质量和编码的时间。对编码时间的评估根据原始编码标准版本各算法的编码数据与本文所改进后算法的编码进行对比，以及改进后编码时间相对原始官方版本的改进率与参考文献中相对于原始官方版本的改进率进行对比。在视频质量方面，分主观质量评定和客观质量评定两个方面进行分析。主观评定主要根据肉眼直接观察，视觉上的感觉来评定。虽然主观评定更能体现人的要求，但在项目及产品开发的过程中往往由于耗时、成本高并不能被广泛应用。客观的质量评定速度快、易于实现，但往往只能大体上反映视频质量的情况。最常用的测试标准是峰值信号与噪声之比(PSNR)，一般讲 PSNR 越高质量应该越好，但是事实上却并不总是这样。

本文的实验环境为：Windows XP 操作系统，Visual C++2008 软件，JM14.0 官方 H.264 编码版本，以及评测所用 YUV 播放器。以下所有的实验数据都是在该环境下完成，下面我们将分别对本文所做的改进工作进行实验验证。

4.1 客观实验数据分析

4.1.1 运动估计算法改进验证

该部分的验证主要从 JM14.0 中的三种算法：快速全搜索、非对称十字多层次六边形搜索算法、EPZS 算法与本文“只改进运动估计的非对称十字多层次六边形搜索算法”对各验证序列的编码时间，运动估计时间，以及 PSNR 为目标来进行对比与分析。如下表 4.1 所示：

表 4.1 各算法编码时间表

| 算法名 测试序列 | 测试项目 | 快速全搜索 | UMHexagonS | EPXS | UMHexagonS(1) |
|-------------|------|-------|------------|-------|---------------|
| Foreman | PSNR | 37.04 | 37.04 | 36.99 | 35.89 |
| | Time | 4.328 | 1.813 | 1.736 | 1.579 |
| | ME | 3.165 | 0.629 | 0.535 | 0.514 |
| akiyo | PSNR | 38.98 | 38.98 | 38.98 | 36.01 |
| | Time | 4.172 | 1.345 | 1.406 | 1.218 |
| | ME | 3.094 | 0.267 | 0.252 | 0.297 |
| bridge | PSNR | 35.26 | 35.25 | 35.25 | 33.59 |
| | Time | 4.281 | 1.562 | 1.531 | 1.329 |
| | ME | 3.062 | 0.297 | 0.311 | 0.233 |
| carphone | PSNR | 37.44 | 37.40 | 37.43 | 35.56 |
| | Time | 4.312 | 1.688 | 1.718 | 1.454 |
| | ME | 3.136 | 0.500 | 0.499 | 0.359 |

表 4.1 中为 4:2:0 的 QCIF 测试序列, Time 为编码总时间, ME 为运动估计的时间, UMHexagonS(1)为只进行过运动估计优化之后的算法。

从上表的对比我们可以看得出来, 经过运动估计的改进, 与官方 JM14.0 版本相比, 大幅度的提高了运动估计的速度, 在编码时间上有了较大的改善。但是我们同时也看出了改善所带来的坏处, 就是 PSNR 有了不小的下降, 但是经过对改进图片的观察, 依然属于可以接受的范围, 下一节中, 我们将从图片看总改进之后的算法的视觉效果。

4.1.2 总改进验证

在经过变换与量化的改进之后, 本文所进行的优化已经全部结束, 在效果上较光改进运动估计的效果更加优越, 接下来我们将对总的结果进行分析, 对最后的效果对各个测试序列进行测试, 所得结果如表 4.2 所示:

表 4.2 总改进效果

| 测试项目 测试序列 | PSNR | Time | ME |
|--------------|-------|-------|-------|
| Foreman | 34.89 | 1.499 | 0.47 |
| akiyo | 36.50 | 1.048 | 0.233 |
| bridge | 32.59 | 1.235 | 0.375 |
| carphone | 34.57 | 1.328 | 0.409 |

上表是在与表 4.1 的实验环境同样的情况下，对完全优化结束之后的算法进行的测试数据，从该数据我们可以看的出来，在时间的改进上，有一个很好的好的效果，在 JM 提供的算法中，UMHexagonS 算法已经有很好的效果，而改进之后的算法比原来的 UMHexagonS 算法有将近百分之二十的改进，当然这种算法的改进对 PSNR 的影响也是较大的。

4.1.3 QP 值选择

H.264 总共设立了五十二个 QP 值，可以通过改变该参数值，满足各种不同的需要，本文能过对 QP 值对编码效果的影响，进行了大量的实验选择，这里利用一组具有典型意义的数据来说明 QP 值的选择，下图为根据各个 QP 值对应的 PSNR 值的关系图，以及对应的编码时间图，如图 4-1 所示：

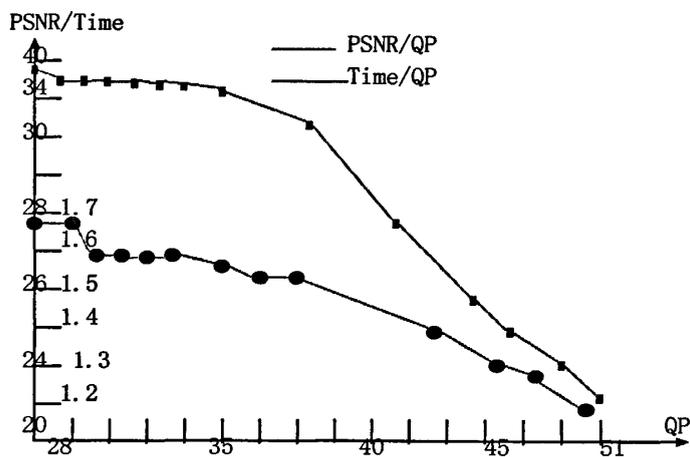


图 4-1 QP 测试图

从上图的测试过程，我们可以看出在 28 到 35 有范围内都是比较平稳的，两条曲线的趋势基本一致，经过实验的探索，以及对图片效果的对比，本文算法最后选取 QP 为 32。

4.1.4 实验结果比较

目前有不少论文对 H.264 进行了改进，本文节选其中几个改进算法与本算法进行编码改进率的对比，改进率的计算是采用改进后速度与官方版本速度相对比得来，用公式表示如下：

$$\text{改进率} = (\text{官方编码时间} - \text{改进算法编码时间}) / \text{官方编码时间} \quad (4.1)$$

具体的改进对比表 4.3 所示：

表 4.3 本文算法与别的文献算法对比

| 算法 | 参考文献 23 | 参考文献 49 | 本文算法 |
|--------|---------|---------|------|
| 改进率(%) | 12 | 10 | 18 |

表 4.3 中所引用参考文献中所进行的算法，是单独的对运动估计进行优化和单独对变换量化进行优化的结果，因未发现与本文类似同时修改的文献，所以只能与上述文献进行对比，在编码的速度上我们可以看出，本文略优于参考文献算法。

由于从其文献中只能看到实验数据，上述数据是经计算后取平均获得，有一定的误差。而且单纯从 PSNR 来看，本文算法略差于参考文献算法。因参考文献中没有对应图片效果，所以无法进行视觉对比。

4.2 改进算法实验结果

在开发与评比的过程中往往更注重对 PSNR 的评比一些，但是事实上并不是 PSNR 所产生的结果就是准确的，有时候人眼对物体的观看侧重点不同，所以最终的效果还需要从实际的观看去证实，本文改进后算法与原始照片，以及由 UMHexagonS 编码的照片演示如图 4-2 所示：

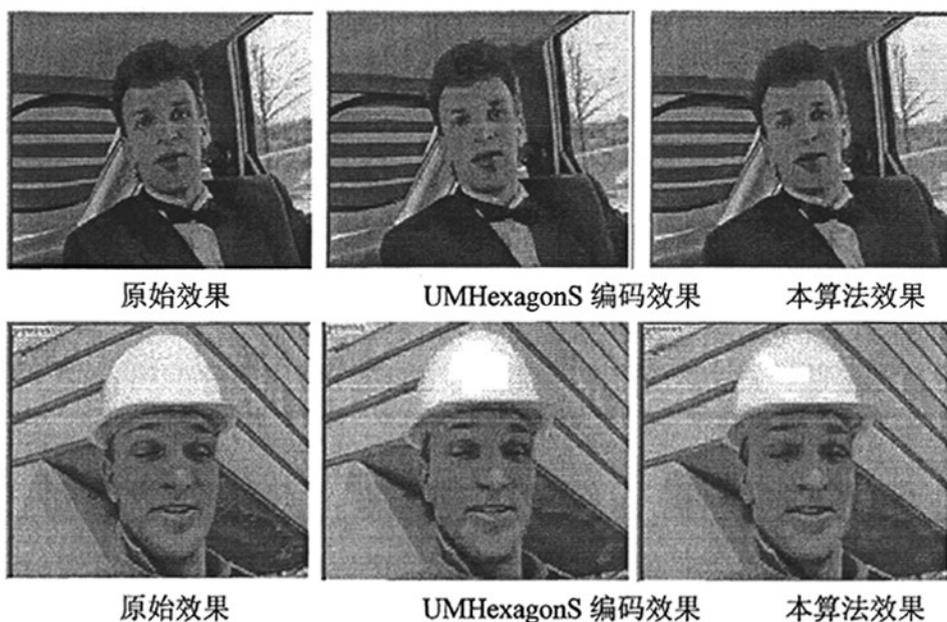


图 4-2 编码对比图片

图 4-2 为将原始图片显示，以及将 UMHexagonS 算法及本算法经编码后的图片解码进行显示的效果对比。从照片效果我们可以看的出来，在大幅度提高编码速度的前提下，该图片的效果虽然有所衰减，但是仍然处于我们可以接受的范围之内，所以本算法还是有一定的优越之处。

4.3 本章小结

本章内容主要是对全文内容中所提到的对算法的改进工作，进行实验的验证以及对比分析，通过本章实验的论证，我们可以看的出来，对于普通的监控或视频电话之类的运动幅度不是很大的情况下，本文所改进算法可以在保证图片质量可以接受的情况下，大大提高编码的速度，所以本算法对于在实时性要求较高的场合有更加实用的价值。

第 5 章 总结与展望

随着社会应用需求的发展,实时视频编码技术得到了广泛的应用,尤其在实时监控方面显得更为重要。不仅是大的企业、单位需要进行安防监控,对于私家车等私人物品以及家庭安全的监控也显的尤为重要。中国电信利用 3G 网络提供全球眼来随时监控家庭及通过手机视频电话与家人沟通,这种家庭网络多媒体业务的发展也已经开始。但是视频编码作为视频技术的核心,现在并不能适应所有的环境,这也正是视频电话费用高,且不易推广的原因。为了解决这个问题,本文研究了当前最新视频编码技术 H.264,对其中的关键技术进行分析,最后以其中的运动估计及整数变换与量化为切入点,对算法进行了优化,使编码速度得到提高。最后通过实验测试数据可以看出,在保证视频质量满意的前提下,算法达到了提高视频编码速度的目的。

5.1 本文工作总结

本文在详细的研究了实时视频技术,并且对算法应用背景—视频监控技术做了简单说明。对最新视频编码技术 H.264 进行了深入的学习。最后以加快编码速度为出发点,争取在保证图片质量的情况下使编码时间大大的缩短,其中主要工作如下:

(1) 研究了 H.264 的关键技术运动估计的原理及方法,在学习 H.264 标准中采用的三种搜索算法的基础上选取其中一种效果比较好的算法—非对称十字多层次六边形搜索算法进行了改进。在该算法的搜索过程中加入了自适应的阈值,在搜索的过程中起到了提前截止的作用。并且由于该算法总共分为五个阶段,而且在前两阶段由于其并不太耗时,属于可以接受的范围,而进入后面的过程则非常耗时,所以本文在第二个阶段之后,基于生活中的物体主要进行的是水平运动,垂直运动非常少这一点考虑,将其五步搜索,增加一步矩形搜索,大多数情况下避免了耗时的六边形搜索。由于运动估计在整个编码的过程当中,属于较费时过程,所以通过改进,提高了运动估计的速度,从而提高了整体的编码速度。

(2) 整数变换及量化的过程在整个编码的过程当中占有很重要的地位,也

是一个非常耗时的步骤。所以本文深入学习与研究了 H.264 所采用的整数变换与量化的过程。由于整数变换与量化是以离散余弦变换为基础，所以本文详细研究了一维和二维离散余弦变换的相关知识，对其整个推算过程进行了细致的学习。并且通过研究全零块判决的算法，在整数搜索的过程当中加入了全零块判决过程，由于对全零块的编码与量化在视频质量方面并不能起到大的效果，相反却对编码速度有很大的影响，所以对于检测到的全零块，省去其编码及量化过程。并且在普通全零块判决算法的基础上，对阈值的选取及搜索过程做了进一步的改进。由于全零块所占比率比较大的原因，经过本文的改进，进一步的缩短了编码的时间。

(3) 最后利用 H.264 标准的软件参考模型 JM14.0 对本文所提的算法进行了实现及实验测试。以及以主观和客观两种对于视频质量评估的判决方法为依据，将本文所提算法进行验证，从实验结果可以看出，本算法所实现的编码图片属于较好的效果，并使实时视频编码的速度得到了很大的提高。

5.2 展望

在本课题的研究过程中，虽然已经深入的研究了视频编码的关键技术，以及详细的研究与学习了最新编码标准 H.264 的关键技术，并在编码技术的核心：运动估计和变换与量化，进行了改进与优化，但是相对于各界的要求以及该学术知识的广泛还显的很微不足道，所以进一步的发展空间还相当的大，在改进方面可以做的工作也还相当的多。对于下一步可以进行的研究总结如下：

(1) 本文虽然在编码的速度上有了很大的改进，但是在图片的质量方面并没有达到最好，更进一步的细致研究，应该可以在编码速度提高的情况下，同样保证视频质量的优良，甚至提高视频质量。

(2) 就本文而言，在两方面的改进工作上并没有得到叠加的效果，相互之间有一定的影响，所起到的效果是 $1+1<2$ 的，所以可以从这个切入点出发，更进一步的提高编码的速度。

(3) 本文所做工作停留在了软件的层次上，并未深入到与硬件结合使用，在与硬件的结合中，可以进一步通过优化代码的实现，充分利用现在处理器的高处理能力，多线程同步处理，更进一步的提高编码的速度。

参考文献

- [1] 高磊. 安防视频监控现状及发展趋势.中国安防,2007.9.
- [2] 蒋卫星.视频监控开发平台的设计与实现:[硕士学位论文].浙江:浙江大学计算机应用技术.2008.
- [3] Giovanni Gualdi,Andrea Prati and Rita Cucchiara. Video Streaming for Mobile Video Surveillance[J]. IEEE,Transaction on multimedia.VOL 10 NO.6.2008-10.
- [4] S.Sutor, F.Matusek, Kraus.K and R.Redu. Large-Scale Video Surveillance Systems:New Performance Parameters and Metrics[C]. IEEE,The Third International Conference on Internet Monitoring and Protection.June 29 2008-July 5 2008 Page(s):23~30..
- [5] 毕厚杰.新一代视频编码标准 H.264\AVC.人民邮电出版社,2005.
- [6] S.Sutor, F.Matusek, Kraus.K and R.Redu.A Nearly Indestructible Video Surveillance System[C]. IEEE,The Third International Conference on Internet Monitoring and Protection.June 29 2008-July 5 2008 Page(s):98~102.
- [7] W.P.Berriss,W.G.Price and M.Z.Bober. REAL-TIME VISUAL ANALYSIS AND SEARCH ALGORITHMS FOR INTELLIGENT VIDEO SURVEILLANCE[C]. VIE 2003,International Conference on Visual Information Engineering.7-9 July 2003 Page(s):226~229.
- [8] 吴俊德.H.264 帧内编码及转换编码研究:[硕士学位论文].南京:国立中央大学.1997.
- [9] 葛呗呗.H.264/AVC 视频编码算法及其 DSP 实现研究:[硕士学位论文].西安:西北工业大学.2006.
- [10]LiZhuo,Qiang Wang,David Dagan,Lansun Shen.Optimization and Implementation of H.264 Encoder on DSP Platform[C].IEEE.International Conference on Multimedia and Expo.2-5 July 2007 page(s):232~235.
- [11]M.Vranjes,S.Rimac-Drlje,D.Zagar.Subjective and Objective Quality Evaluation of the H.264/AVC Coded Video[C].IWSSIP 2008.15th International Conference on Systems,Signals and Image Processing.25-28 June 2008 page(s):287~290.
- [12]杨铭.H.264 视频编码快速算法与算术编码检错研究:[硕士学位论文].北京:清华大学电子工程系.2004.
- [13]李宇峰.H.264 视频编码器的优化及其工程应用研究:[硕士学位论文].南京:南京理工大学系统工程系.2006.
- [14]郑君.H.264 视频编码器在 DM6437 平台上的实现与优化:[硕士学位论文].武汉:华中师范大学电路与系统专业.2008.

- [15] Yun Cheng, Zhiying Wang, Jianjun Guo, Kui Dai. Research on Intra Modes for Inter-Frame Coding in H.264[C]. IEEE. The 9th International Conference on Computer Supported Cooperative Work in Design Proceedings. Volume 2, 24-26 May 2005 Page(s): 740~744 Vol.2.
- [16] Zhenyu Wei, Kai Lam Tang and King N. Ngan. Implementation of H.264 on Mobile Device [J]. IEEE. Transactions on Consumer Electronics. Volume 53, Issue 3. Aug. 2007 page(s): 1109~1116.
- [17] 刘文俊. H.264 帧内预测算法优化及几个重要模块的 FPGA 实现:[硕士学位论文]. 成都: 电子科技大学信号与信息处理. 2004.
- [18] 晏志勇. IP 可视电话的程序优化及软件研发:[硕士学位论文]. 南昌: 南昌大学信息工程学院电子信息工程系. 2008.
- [19] 曹文明. 基于 DM642 的 H.264 标准的可视电话设计与实现:[硕士学位论文]. 长沙: 湖南大学软件学院. 2005.
- [20] 谢仁宏. 基于 TMS320DM643 的 H.264 编码器帧内预测算法优化及实现:[硕士学位论文]. 南京: 南京理工大学通信与信息系统. 2008.
- [21] ZHANG Shi, LIU Xiaozhi, JIAO Tianguang, QI Jinlong. Transplant and Optimization of H.264 Encoder Based on DSP Platform[C]. WICOM 08. 4th International Conference on Wireless Communications, Networking and Mobile Computing. 12-14 Oct. 2008 page(s): 1~5.
- [22] 王会鲜, 陈伟, 谢涛, 郑洪江. 基于块匹配的运动估计算法研究和优化. 武汉理工大学学报. 信息与管理工程版. VOL 30 NO 6 2008. 12.
- [23] 张素文, 杨富森, 汪丽丽. 一种改进的快速多分文分辨率运动估计算法. 计算机应用. VOL 28 NO 5 2008. 5.
- [24] 郑振东, 王沛, 应骏. H.264 JM 模型中运动估计算法及改进方案. 中国图象图形学报. VOL 12 No 10 2007. 10.
- [25] Hoi-Ming Wong, Oscar C. Au, Andy Chang, Shu-Kei Yip, Chi-Wang Ho. FAST MODE DECISION AND MOTION ESTIMATION FOR H.264[C]. IEEE 2006. ISCAS International Symposium on Circuits and Systems. ISCAS 2006. 1692625.
- [26] FREDERIC DUFAUX, FABRICE MOSCHENI. Motion Estimation Techniques for Digital TV: A Review and a New Contribution[J]. Proceedings of the IEEE. Volume 83. Issue 6. June 1995 Page(s): 858~876.
- [27] 沈工建, 马继东, 曲福明, 潘博, 王安娜. 一种基于分类和有效区域的运动估计快速搜索算法. 中国科技论文在线. TN919. 81.
- [28] Bin Li and Kai-Kuang Ma. Unequal-arm Adaptive Rood Pattern Search with Early Terminations for fast block-matching motion estimation on H.264. IEEE. APCCAS 2006. IEEE Asia Pacific Conference on Circuits and Systems. 4-7 Dec. 2006 Page(s): 402~405.

- [29] Jen-Shiun Chinag, Hsin-Guo Chou, and Je-Yu Tzou. An efficient and regular motion estimation algorithm for MPEG4-H.264/AVC coding[C]. IEEE 2005.9th International Workshop on Cellular Neural Networks and their Applications. 28-30 May 2005 Page(s):253~256.
- [30] Xiang Li, Eric O. Li, and Yen-Kuang Chen. Fast multi-frame motion estimation algorithm with adaptive search strategies in H.264[C]. IEEE Proceedings ICASSP 04. International Conference on Acoustics. Volume 3, 17-21 May 2004.
- [31] 丁永林, 沈海斌. H.264/AVC 中的运动估计算法对比及改进. Digital TV & digital Video. 1001-8692(2006)05-0017-02.
- [32] Wei Jian Yang. An Efficient Motion Estimation Method for MPEG-4 Video Encoder[J]. IEEE Transactions on Consumer Electronics. Volume 49, Issue 2. May 2003 Page(s):441~446.
- [33] Jesus Malo, Juan Gutierrez, I. Epifanio, Francesc J. Ferri, and Jose M. Artigas. Perceptual Feedback in Multigrid Motion Estimation Using an Improved DCT Quantization[J]. IEEE Transactions on Image Processing. VOL. 10, No. 10. OCTOBER 2001.
- [34] 石迎波, 吴威柯. 基于 H.264 的多参考帧运动估计快速算法. Computer Engineering 多媒体技术及应用. Vol 34 No 10. May 2008.
- [35] C. J. Duanmu, Yu Zhang, Xing Chen, Shuihong Zhou. Multi-octagon-grid search algorithm for fast motion estimation[C]. IEEE 2008. International Conference on Multimedia and Expo. June 23 2008-April 26 2008 Page(s):749~752.
- [36] Choudhury A. Rahman and Wael Badawy. UMHexagonS Algorithm Based Motion Estimation Architecture for H.264/AVC[C]. IEEE Proceedings. Fifth International Workshop on System-on-Chip for Real-Time Application. 20-24 July 2005 Page(s):207~210.
- [37] Hyosun Yoon, Hyesuk Kim, Miyoung Kim, Anh Nga Lai Thi, Guesang Lee. Motion Estimation for Real Time Transmission on H.264/AVC[C]. IEEE. CIT 2007. seventh IEEE International Conference on Computer and Information Technology. 16-19 Oct. 2007. Page(s):573~576.
- [38] 刘志胜, 彭玉华. 基于 H.264 的自适应快速运动估计算法. 计算机工程与应用. 2008, 44(5).
- [39] Chiung-Chin Lin, Yu-Ming Lee and Yinyi Lin. Fast Motion Search using early Zero-block Detection[C]. IHMSP 2007. Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing. Volume 1, 26-28 Nov. 2007 Page(s):600~603.
- [40] Zhenyu Liu, Satoshi Goto and Takeshi Ikenaga. Fast motion estimation for H.264/AVC using image edge features[C]. IEEE 2008. International Conference on Multimedia and Expo. June 23 2008-April 26 2008.

- [41]Hyosun Yoon,Hyesuk Kim,Miyoung Kim,Lai Thi Anh Nga,Guesang Lee.Hierarchical Integer pixel and Adaptive Fractional pixel Motion Estimation[C].IEEE.8th International Conference on Computer and Information Technology Workshops.
- [42]睦保辉,吴强,徐成,李仁发,刘彦.自适应的快速运动估计算法.计算机工程应用.2008,44(34).
- [43]S.R.Subramanya,Hiral Patel Ilker Ersoy.Performance Evaluation of Block-Based Motion Estimation Algorithms and Distortion Measures[C].IEEE.Coding and Computer 2004.ITCC2004. International Conference on Information Technology. Volume 2,2004 page(s):2~7.
- [44]Jiancong Luo,Ishfaq Ahmad,Yongfang Liang and Viswanathan Swaminathan.Motion Estimation for Content Adaptive Video Compression[J].IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.VOL18.NO.7.JULY 2008.
- [45]乱海幽,砖迭侦,乱抡,欠袍.4x4 整数变换的流算法研究.系统仿真学报.Vol.18 Suppl.1 Aug,2006.
- [46]颜洪奎,朱珍民,沈燕飞,肖建华.一种 H.264/AVC 的自适应去块效应滤波快速算法.计算机工程与应用.2008,44(29).
- [47]Sehyun Song,Changwoo Seo,Kichul Kim.A Unified Transform Unit for H.264.ISOCC 2008.International SoC Design Conference.Volume 02,24-25 Nov.2008 Page(s):130~133.
- [48]Vivek K Goyal.Transform Coding with Integer-to-Integer Transforms. IEEE TRANSACTIONS ON INFORMATION THEORY,VOL.46,NO.2.MARCH 2000.
- [49]胡灏.H.264 标准中变换与量化过程的优化:[硕士论文].天津:天津工业大学计算机应用技术.2007.

致 谢

春去秋来，时光荏苒。转眼间，硕士求学生涯即将划上圆满的句号。三年多来老师的谆谆教诲、同学的热心相助都将成为我生命中最美丽的图画，永不褪色！

首先向尊敬的导师李方敏教授致以最深的谢意和敬意！李老师为人谦和、治学严谨。在他身上，我学到了细心和一丝不苟。在日常项目和学术交流中，总能感受到他对问题的敏锐和执着，他深刻的见解和严密的逻辑思维让我叹服；更重要的是，我从李老师身上不断体会到研究和解决问题的方法，使得研究问题的思路得到了极大拓展，这将对我以后的研究工作大有裨益。在他每一次耐心细致的修改和不厌其烦的指正下，我的论文得以完善和提高。感谢李老师的严格要求，我才能在不断的压力中进取，找到自己的奋斗目标和位置！

非常感谢周祖德教授和刘泉教授在学术上给予的悉心指导和支持。感谢他们在我这三年的硕士生活中对我的关心和指点！

感谢实验室师兄程松林博士、任平博士、刘新华博士、韩平博士、忠毅老师等，和他们的交流和讨论扩展了我的知识面，灵活了我的思路，使得我在研究过程中能够解决每个难题，论文工作得以顺利完成。

感谢所有在学习、工作和生活中帮助过我的老师、同学和同事们！

特别感谢我的父母及家人，他们是我完成学业的精神支柱，也是我心中永远的最爱！

最后，感谢各位专家在百忙之中评阅我的论文！

作者攻读硕士学位期间发表的学术论文

- [1] 薛志坚,李方敏,实时视频最优化编码研究.中国科技论文在线,200809-506.