

摘要：随着多媒体技术和计算机网络的快速发展，数字媒体的制作和传播变得更加方便和快捷，同时盗版和侵权的问题也日益严重。数字水印技术作为一种数字媒体版权保护的有效方法，近年来引起了人们的高度重视。本文详细论述了数字水印技术的背景、原理和典型算法，并在这基础上提出了一种新的基于三维小波变换的自适应视频水印算法。为了实现自适应，该算法利用了人类视觉系统的特性，在空域中对视频图像分块做排序，并根据排序结果，调整水印信息的嵌入策略。实验结果表明，该算法在满足了不可见性的前提下，具有良好的鲁棒性。

关键字：数字水印，版权保护，视频水印，人类视觉系统，~~自适应~~

Abstract *The digital media has become a more fast and convenient way for information communication along with the popularization of Internet and the development of multimedia technique. But at the same time, this also leads to serious problem regarding IPR protection. As an effective method to provide copyright protection for digital media, digital watermarking technology has drawn extensive attention in recent years. For this paper, some important topics about digital watermarking are discussed, which include not only the background and principles of watermarking but also some classic algorithms. Apart from that, we propose an adaptive video watermarking based on three-dimensional wavelet transformation. In order to realize adaptability, the algorithm splits the original frames into 3D-blocks and lines up these blocks according to HVS characteristics. Watermark information with different strength is inserted into these blocks based on the particular order. The experimental results demonstrate that the watermarks are invisible and robust against MPEG compression and some commonly used attacks.*

Keywords: *Digital watermarking, Copyright protection, Video watermarking, HVS, adaptability*

第一章 引言

在多媒体和网络技术迅猛发展的今天,图像、文本、音频和视频等数字形式的产品在最近十年里已经得到了广泛的传播与运用。制造商、销售商和用户都可以方便地利用数字设备制作、处理和存储图像、音频、文本和视频等数字媒体。与此同时,数字媒体又能很容易地借助 Internet 或 CD-ROM 被复制、传播和公开。盗版者正是利用数字产品易复制、易传播及易处理的性能来侵害制造商和用户的合法权利以牟取个人利益,从而就引发出了数字信息安全问题和数字产品版权保护问题。如何在网络环境中实施有效的版权保护和信息安全手段,已引起学术界、企业界以及政府部门的广泛关注。数字水印技术 (*Digital Watermarking Technology*) 就是一个新兴的能够有效地保护数字产品版权的技术。

数字水印技术的基本思想是在数字图像、音频和视频等产品中嵌入秘密的信息,以便保护数字产品的版权或证明产品的真实可靠性。数字水印技术弥补了传统信息安全技术——加密技术的缺陷,因为它可以为解密后的数据提供进一步的保护,而后者在密文数据被解密后,就失去了保护的作用。而且,数字水印技术可以一次性地嵌入大量的秘密信息,能够有效地实现版权标注的功能。因此,数字水印技术已经成为了当前多媒体信息安全研究领域发展最快的热点技术。

可以说,数字水印技术是一门新兴的多学科交叉的应用技术,它涉及不同学科领域的思想和理论,如信号处理、图像处理、信息论、编码理论、密码学、信号检测理论、概率论、随机过程理论、数字通信、对策论、计算机科学及网络技术、算法设计等技术,还包括公共策略和法律等问题。目前,国内外许多的大学和研究机构都在各自的领域中对水印技术进行研究,也诞生了不少有价值的理论和算法。

但是,就目前掌握的情况来看¹,国内外(尤其是国内)的研究大多集中在静态图像的水印技术上,而对音频和视频水印的研究就少了许多。因此,本文的研究对象主要针对数字水印研究的薄弱环节,即视频水印。文中提出了一种新的基于三维小波变换域的自适应视频水印算法,通过实验证明,它可以适用于视频数字媒体的版权保护上。

本篇论文的主要内容如下:

第二章介绍数字水印技术的背景知识。该章涉及到数字媒体的特点以及传统加密技术的不足,信息隐藏学的概念和典型方法,并简单阐述了水印技术的发展历史。

¹ 主要的依据是对 1999.1 到 2002.12 期间发表在国内外主要科技期刊上的论文的收集和统计。

第三章为数字水印技术的综述。本章首先提出了数字水印的系统模型和研究方法。然后对与水印技术有密切关系的几个研究领域，如扩频通信、差错编码理论、数据融合以及分形图像压缩等做了介绍，着重突出了它们对水印技术的影响。之后，本章又给出了数字水印技术的分类和应用领域，以及水印算法的技术要求。由于水印算法必须要面对许多有意或无意的攻击，要设计出可以抵御这些攻击的算法，就应该先了解这些攻击手段，因此本章还详细地介绍了几种典型的攻击方法，如鲁棒性攻击、表达攻击和解释攻击等。在随后的部分，文章还阐释了几种静态图像水印的典型算法，它们对作者设计视频水印的算法有指导和启示的作用。此后，本章又介绍了一种实用的安全数字水印体系模型，它参考了已有的网络认证和保护体系标准，将加密技术和认证体系配合水印算法，形成了安全、完整、可行的数字产品版权保护机制，具有现实意义。最后，本章论述了视频水印技术的研究现状，以及本文算法所属的基本类型。

第四章是论文的核心部分。在本章里，作者给出了一种新的视频水印嵌入和提取算法。此算法的特点是：

一、嵌入的水印为 24 bit 真彩色图像，信息量大，表现能力强；

二、利用了人类视觉系统 (HVS) 的有关特性，根据视频图像内容来自适应地选择水印信息的嵌入位置和相应的嵌入策略；

三、基于局部的三维小波变换，算法运算量较小；

四、提取水印图像时，实现了无需原始视频数据的“盲水印”要求。

由于水印的嵌入和提取过程都在小波变换域中进行，因此本章首先介绍了小波变换的原理以及它在图像处理方面的应用。然后，本章介绍了对水印图像的处理过程，包括水印信息的压缩，二值化、位分离和置乱技术等。本章其余部分详细地阐述了水印嵌入和提取的过程。

第五章为实验结果与分析。由于本文提出的算法主要用于视频的版权保护，所以要求它具备不可见性和鲁棒性。本章主要给出了对水印算法这两项指标的检验。另外，通过与非自适应算法的比较，也有力地说明了本文算法的优越性。

第六章是本文的结论，并且提出了今后研究的方向。

第二章 数字水印的背景知识

《数字化生存 (Being Digital)》的作者 Negroponte 在书中写道：“计算不再只和计算机有关，它决定我们的生存”。人类已经迈入了信息时代，数字化的各类信息，我们可以称之为多媒体信息至少正在改变着我们的生活（如果说“决定”的话）。同时，信息安全也面临着前所未有的挑战，单纯依靠原有的技术已经无法满足新的需要。所以，像数字水印这样的新技术慢慢地发展了起来。本章将对数字水印技术的背景知识作详细地介绍，内容涉及到传统的加密技术、信息隐藏学的概念以及水印技术的历史渊源。

2.1 数字媒体与传统的加密技术

这里所称的数字媒体可以是文本、图像、音频或视频，它们与模拟媒体相比有着鲜明的特点。首先数字媒体的质量要比同类的模拟媒体好得多；其次对数字媒体的编辑更加容易，因为它是离散的，每一个离散的值都可以被修改；数字媒体的复制十分简单而且复制品几乎和“原件”一模一样；最后，数字媒体能更好地在网络上进行传播。

数字化虽然可以使信息成为了大众的资源，但对于数字媒体的所有者来说，就要面对更多的困扰。他们辛辛苦苦的劳动果实也许很快被人当作肯德基的快餐一般转手卖给了他人，而他自己一点回报也得不到。虽然，他可以求助法律来讨回公道，但司法鉴定也无法准确地判断哪个是原件哪个是拷贝。数字媒体的存取方便，篡改也极容易，与存在“笨重”的磁带、电影胶片中的模拟媒体数据相比，它简直可以使那些侵权者不费吹灰之力。所以，如何保护数字媒体的版权，如何防止其拥有者的合法权益受到侵害，成为了信息时代的当务之急。

传统的对数字信息安全的保护方法就是加密技术^[6]。加密技术主要研究如何将机密信息进行特殊的编码，以形成不可识别的密码形式（密文）进行传递。它的关键是密钥技术，其强壮性是依靠不断增加密钥的长度来获取的。这种加密技术作为网络安全技术的主流，也存在着一些不足和局限性。

第一，数字信息的加密仅发生在发送和传输环节，一旦密文被解密也就失去了保护的作用，所以它无法对数字产品进行进一步的跟踪和监督，如果那个合法用户（暂时还可以这么称呼他）将数字产品传播给了非授权者，加密技术对这种情况束手无策；第二，加密后的数字信息是无序的不可识别状态，只有那持有密钥的人才能访问。而人们有时所要禁止的并非是数字产品的内容，而是对数字产品的非法使用；第三，加密算法往往是公开的，对于加密信息的安全性是通过增加密钥长度来实现的。随着计算机处理能力的不断提高，这种方法显

得越来越不安全了；第四，加密信息容易引起某些人的注意，从而增加了其被攻击的可能性；第五，对于数字签名等技术来说，由于应用时需要在原始数据中附加大量的签名数据，所以对于数字图像、音频和视频的应用既不方便也不实用。

鉴于加密技术的种种不足，近年来又一种信息安全技术——信息隐藏学应运而生了，它在信息安全技术领域越来越显示出重要的作用。

2.2 信息隐藏学概述

1996 年，在英国剑桥牛顿研究所召开了第一届国际信息隐藏学术研讨会，这标志着一门新的交叉学科——信息隐藏学的正式诞生（见文献[3]）。信息隐藏也被称为“信息隐匿”或“信息隐形”。到目前为止，信息隐藏还没有一个准确和公认的定义。一般认为，信息隐藏学是与信息安全研究领域与密码技术紧密相关的一个研究领域。信息隐藏学和上节所讲的信息加密技术都是为了保护秘密信息的存储和传输，使之免遭其他人的破坏和攻击，但两者之间有着显著的区别。

信息隐藏学主要研究如何将某一秘密信息隐藏于另一公开的信息中，然后通过公开信息的传输来传递秘密信息。可能的非法截获者难以从公开信息中判断秘密信息是否存在，难以截获秘密信息，从而能保证秘密信息的安全。由此可见，信息加密所保护的是信息的内容。信息隐藏则不同，它的目的是使非法截获者不知道哪里有秘密，它隐藏了信息的存在形式。

为了更清晰地了解信息隐藏的原理，请看图 2.1，它是一个信息隐藏的通用模型。

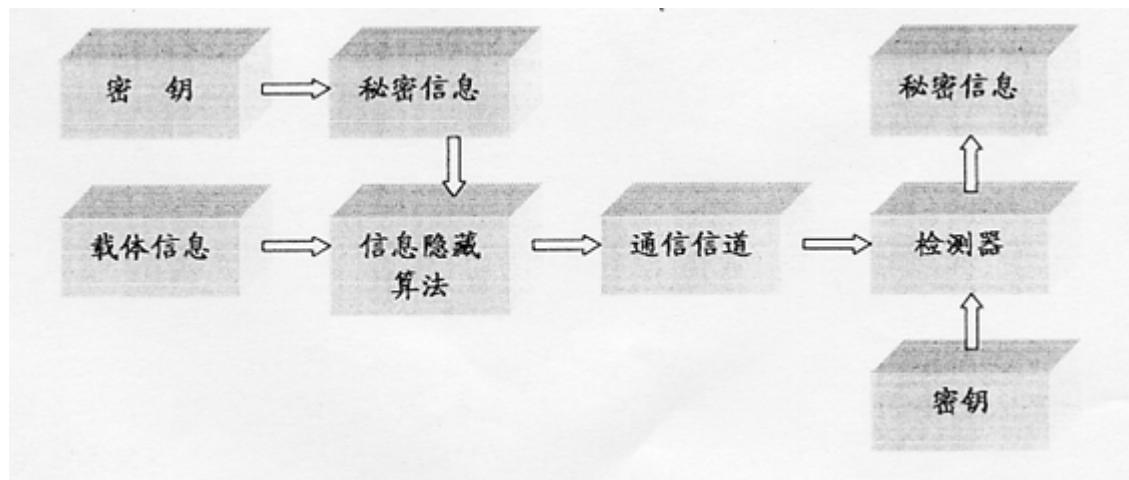


图 2.1 信息隐藏模型

我们称隐藏的信息为秘密信息（*Secret message*），它可以是版权信息或秘密数据，也可以是一个序列号；而公开的信息则称为载体信息（*Cover message*），如视频、音频片段。这种

信息隐藏过程一般由密钥 (Key) 来控制, 通过嵌入算法 (*Embedding algorithm*) 将秘密信息隐藏于公开信息中, 而隐蔽载体 (隐藏有秘密信息的公开信息) 则通过信道 (*Communication channel*) 传递, 然后检测器 (*Detector*) 利用密钥从隐蔽载体中恢复/检测出秘密信息。

信息隐藏技术主要由下述两部分组成: (1) 信息嵌入算法, 它利用密钥来实现秘密信息的隐藏; (2) 隐蔽信息检测/提取算法 (检测器), 它利用密钥从隐蔽载体中检测/恢复出秘密信息。在密钥未知的前提下, 第三者很难从隐蔽载体中得到或删除, 甚至发现秘密信息。

信息隐藏算法层出不穷, 下面介绍几种典型的方法。

(一) 在文本文件中, 字与字间、行与行间均有一定的空白间隔, 把这些空白间隔精心改变后可以隐藏某种编码的标记信息以识别版权所有者, 而文件中的文字内容不需作任何改动。

(二) 商业应用中的数据锁定。出版商从降低成本的角度出发, 可以把多个软件或电子出版物集成到一张光盘上出售, 盘上所有的内容均被分别进行加密锁定, 不同的用户买到的均是相同的光盘, 每个用户只需付款买他所需内容的相应密钥, 即可利用该密钥对所需内容解除锁定, 而其余不需要的内容仍处于锁定状态, 用户是看不到的。这样, 拥有相同光盘的不同用户, 由于购买了不同的密钥, 便可各取所需地得到光盘上相应的内容, 这为用户和商家都提供了极大的便利。

(三) 现在的激光打印机具有很高的解析度, 可以控制字符使之发生微小的位移, 人眼对字间距、行间距的微小差别并不十分敏感, 而现在的扫描仪能够成功地检测到这一微小的位移。我们用扫描仪可以高分辨率地获得印刷品的图像, 并通过适当的解码算法找到其中的隐匿标记。

(四) 利用 ASCII 字符的显示特性, 用那些在 CRT 上不显示出来的字符作为隐匿信息嵌入文件中, 一般的文字处理器读不出这些信息, 而利用特定的软件进行解码运算可以读出隐匿信息。

数字水印技术是信息隐藏学的一个分支, 在版权保护方面有着重要的应用。与其他的信息隐藏技术只关心被藏信息的隐蔽性 (只要未被泄漏就行) 不同的是, 数字水印与它所保护的媒体内容或版权拥有者密切相关, 它不允许嵌入的版权信息丢失或遭到严重的破坏, 所以其鲁棒性的要求更高。下一章将对数字水印的基本原理、技术要求及应用等方面做深入地探讨, 本章的第三节先简单介绍数字水印的历史, 从中可以见到, 这门至今尚未成熟的技术却凝结着许多古老的智慧。

2.3 历史

现代的水印技术是由古老的隐写术 (*Steganography*) 发展而来的, 隐写术一词来源于希腊语, “*Stegano*”对应的是“*covered*”, 它的英文意思是“*covered writing*”。隐写术的应用实例可以追溯到非常久远的年代。被人们誉为历史学之父的古希腊历史学家希罗多德 (*Herodotus*, 486—425), 在其著作中讲述了这样一则故事: 一个名叫 *Histaieus* 的人筹划着与他的朋友合伙发起叛乱, 里应外合, 以便推翻波斯人的统治。他找来一位忠诚的奴隶, 剃光其头发并把消息文刺在头皮上, 等到头发又长起来了, 把这人派出去送“信”, 最后叛乱成功了。

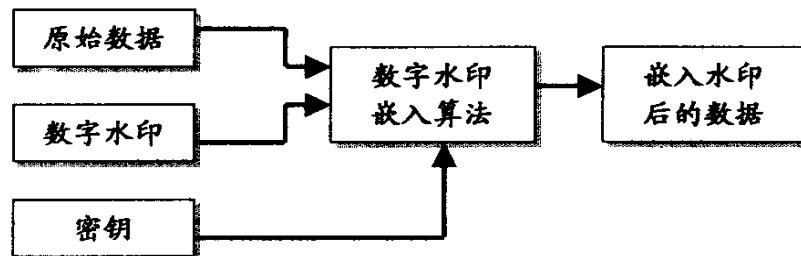
历史上诸如此类的隐写方法还有多种。伟大的音乐家巴赫 (*Bach*) 在自己的乐谱中利用音符的不同排列方式来隐藏自己的名字。17 世纪, 英国的 *Wilkins*(1614—1672)是资料记载中最早使用隐写墨水进行秘密通信的人, 在 20 世纪的两次世界大战中德国间谍都使用过隐写墨水。早期的隐写墨水是由易于获得的有机物(例如牛奶、果汁或尿)制成, 加热后颜色就会变暗从而显现出来。后来随着化学工业的发展, 在第一次世界大战中人们制造出了复杂的化合物做成隐写墨水和显影剂。在中国古代, 人们曾经使用挖有若干小孔的纸模板盖在信件上, 从中取出秘密传递的消息, 而信件的全文则是为打掩护用的。现代又发明了很多方法用于信息隐藏: 高分辨率缩微胶片、扩频通信、流星余迹散射通信、语义编码 (*Semagram*) 等。其中, 扩频通信和流星余迹散射通信多用于军事上, 使敌人难以检测和干扰通信信号; 语义编码是指用非文字的东西来表示文字消息的内容, 例如把手表指针拧到不同的位置可表示不同的含义, 用图画、照片等都可以进行语义编码。

这些例子向我们揭示了信息隐藏学和数字水印技术的一个最基本的目的, 就是将秘密的信息隐藏起来, 不引起别人的怀疑和注意。当然要做到这一点也并非易事, 下面继续介绍数字水印技术的基本原理。

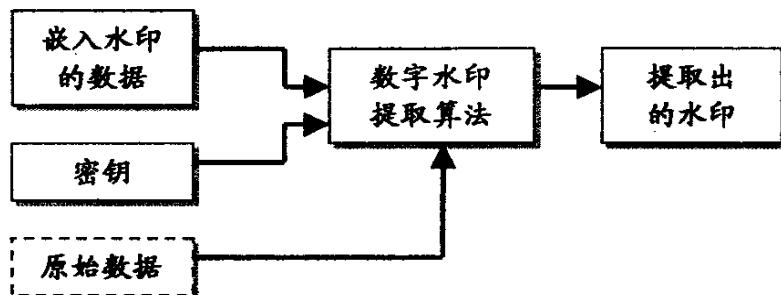
第三章 数字水印技术综述

3.1 数字水印的系统模型

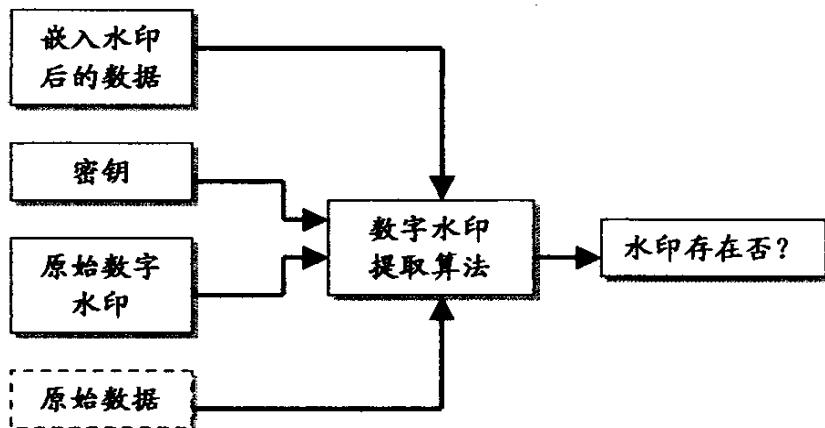
数字水印的系统模型与上一章的信息隐藏模型颇为相似。图 3.1 是数字水印的嵌入模型、提取模型和检测模型，可以同图 2.1 对照来看。



(a) 数字水印的嵌入模型



(b) 数字水印的提取模型



(c) 数字水印的检测模型

图 3.1

图 3.1 中, (a) 为数字水印信号的嵌入模型, 其功能是完成将数字水印信号嵌入原始数据中; (b) 为数字水印信号的提取模型, 其负责从嵌入水印的数据中提取出水印信号; (c) 为数字水印信号的检测模型, 用以判断某一数据中是否有指定的水印信号。

(b) 和 (c) 中的虚框部分表示在提取或检测水印信号时, 原始载体数据不是必要的。

在水印的嵌入过程中加入密钥, 是为了加强水印的安全性, 使得未经授权者 (即无正确的密钥者) 无法检测到水印的存在, 更别说去破坏或消除水印了。水印系统使用密钥, 很好地借鉴了加密技术, 密钥控制了嵌入过程的关键步骤, 如嵌入位置、嵌入强度等, 这样即使攻击者知道了水印嵌入的算法, 如果他没有获得密钥, 同样无法提取、检测或破坏水印。本文将继续对水印中的加密技术作进一步的探讨, 详见第四章。

3.2 数字水印的研究方法

正如第二章所提到的, 信息隐藏学是一门交叉学科, 同样数字水印技术也借鉴了许多其他领域的研究成果。水印嵌入过程采用密钥是一个典型的例子。图 3.2 总结了与水印技术有密切关系的一些领域, 它们虽不能涵盖全部, 但却具有代表性。

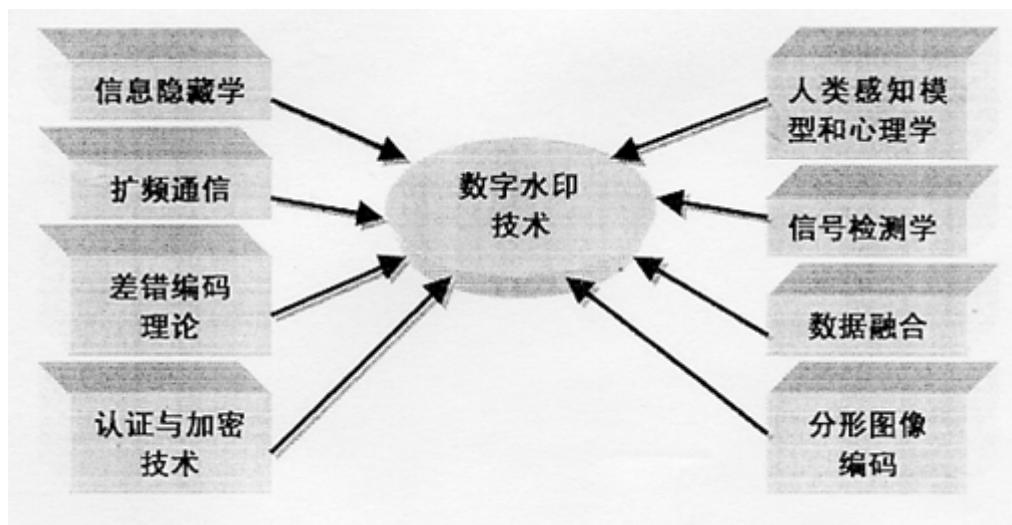


图 3.2 与水印技术密切相关的领域

- ◆ **信息隐藏学** 一般以为, 数字水印技术是信息隐藏学的重要分支 (见第二章)。信息隐藏学的范围更广, 应用的对象不仅是数字媒体, 还包括模拟媒体, 如可以在文字、绘画中隐藏信息。信息隐藏学与数字水印技术的根本目的都是将秘密信息藏起来不为人发现, 但后者对鲁棒性的要求更高。
- ◆ **扩频通信** 水印的关键是鲁棒性, 影响鲁棒性的因素是嵌入对策和水印结构。对于水印

结构, 扩频通信的隐蔽性和强大的抗干扰能力正是数字水印所需要的。利用扩频通信的概念与方法, 将水印信号采用伪随机序列进行调制, 扩展成比原始信息带宽大得多的宽带信号, 提高了水印的鲁棒性。扩频通信的多址原理也对多重水印起一定的作用。本章的 3.7.2.2 节将介绍著名的 NEC 算法, 它就是利用了扩频通信原理的典型算法。

- ◆ **差错编码理论** 鉴于水印系统与通信传输模型的相似性, 有些学者在数字水印算法中引入了差错控制编码技术。比如, 有人在水印嵌入前, 先对它进行卷积编码和交织处理, 以提高其抗误码能力。水印提取时, 采用了软判决维特比译码, 增强了算法的鲁棒性。应该说此种方法在理论上讲是合理的, 但进行差错编码势必会增加信号的带宽, 所以也要考虑到水印不可感知性的要求, 在设计算法时应慎重取舍。
- ◆ **认证与加密技术** 数字水印技术可以借鉴认证与加密技术的原理和方法, 从而使水印算法的安全性更好。对这一点前文已经有所涉及, 这里就不再展开。
- ◆ **人类感知模型和心理学** 人类感知模型和心理学对多媒体数据的压缩编码有指导意义。同样, 在数字水印技术中, 也可以充分地利用它们。水印算法的鲁棒性与不可感知性始终是一对相互矛盾的特性, 鲁棒性要求增大信号的内嵌强度, 而这将使图像或者音乐的质量有所下降。为了解决这对矛盾, 研究者借鉴了人类感知模型和心理学的研究成果, 在视觉或者听觉上敏感的区域减少水印的嵌入强度, 或者不嵌入水印, 而在感觉不敏感的区域加大水印的嵌入强度。这样, 在鲁棒性与不可感知性之间找到了一个平衡点。
- ◆ **信号检测学** 水印的检测过程十分类似信号检测的问题。信号检测学中的最小风险 Bayes 判决准则、Neyman-Pearson 准则以及最小错误概率准则都已被用到了水印检测算法中。
- ◆ **数据融合** 数据融合 (Data fusion) 又称信息融合, 是通过多 (种、类) 传感器数据的总和(集成和融合)以获得比每个单一传感器更多的信息。这里所指的传感器是广义的, 它是指与环境匹配的各种信息获取系统。数字水印技术和数据融合技术也很相似, 例如水印嵌入的过程可以看作是原始数据与水印信息的融合过程, 所以许多数据融合结构如图像的像素级融合、特征性融合、高级决策性融合等都对数字水印技术有启发作用。
- ◆ **分形图像编码** 分形图像编码是目前较有前途的图像编码方法之一。Puate J 等人发明了基于分形图像编码的数字水印算法, 利用了值域子块的最佳匹配定义域子块的位置不变性, 根据最佳匹配定义域子块所处的寻找区域来嵌入和提取有意义的水印。本章的第七节也会对此类算法作进一步的分析。

以上列出的各种研究领域都为数字水印技术提供了宝贵的指导和启示。数字水印技术之所以成为了研究的热点, 一个很大的原因就在于它的“博采众长”。学者们能在各自熟悉的

领域相对比较自由地探讨它,研究它。当然这也会大大增加制定公认的技术标准的难度和时间。

3.3 数字水印技术的分类

按照不同的角度,数字水印技术的分类五花八门,莫衷一是。如按照水印嵌入的载体来分,数字水印可分为文本水印、图像水印、音频水印、视频水印和网络水印等。按照水印的内容来划分可分为有意义水印和无意义水印,有意义水印指水印是某个图标、音频片断、作者名字等有意义的信息,无意义水印是指仅仅为某个随机噪声的水印。这里仅考虑图像与视频的水印,图 3.3 给出了一个典型的水印分类结构,从算法的设计角度,它具有一定的参考价值。

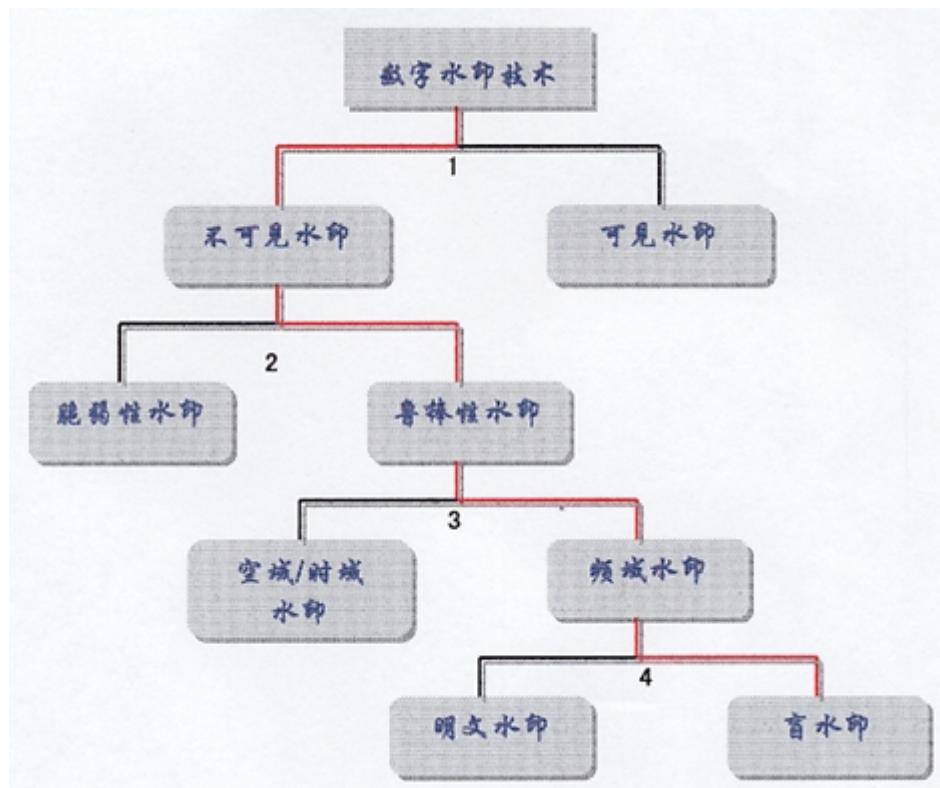


图 3.3 数字水印技术分类树

如上图所示,第 1 种分类是按可感知性来分,不能被人看到的水印称为“不可见水印”,能被人感知的水印称为“可见水印”。本文主要研究不可见水印。第 2 种是按水印的特性来分,脆弱性水印又叫“易损性水印”、“敏感性水印”,它对数据的改动十分敏感,通常被用于完整性保护。鲁棒性水印正好与之相反,要求它能抵得住各种有意或无意的改动。第 3 种是按水印的嵌入方式来划分,空域或称时域水印是指直接改变某些像素值来嵌入水印,频

域水印或称“变换域水印”，它通过某些变换系数被改变以嵌入水印。现在研究的方向主要集中在频域水印上。频域水印又可以分为明文水印和盲水印两种，前者指的是在提取或检测水印时需要原始数据参与的水印方法，而后者在提取或检测水印时无需原始数据。本文第四章提出的算法是基于小波变换域并且是鲁棒的盲水印。

3.4 数字水印的应用领域

数字水印算法的设计要求没有统一的标准，设计者应该根据特定的使用场合来决定算法方案及技术要求。图 3.4 给出了数字水印几种常见的用途。

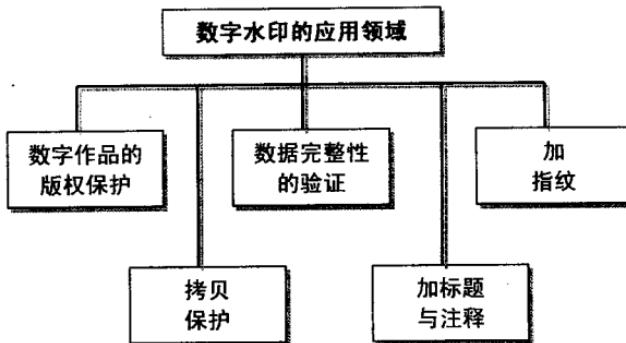


图 3.4 水印的几种常见应用

● 数字作品的版权保护

版权保护是数字水印技术应用的重要方面。随着数字技术和网络的迅速普及，人们获得了越来越多的数字服务，如数字电视、电子商务、远程教育、数字图书馆等等。这些服务的供应者提供的数字作品，如果得不到相应的版权保护，那么供应者的商业利益、社会声誉都极有可能遭受重大的损失。数字水印是解决这个棘手问题的有效手段，嵌入的水印可以是一段文字、标识、序列号和商标等，这种水印通常是不可见或不可察觉的，且与原始数据（如图像、音频、视频数据等）紧密结合，并隐藏其中，已成为源数据不可分离的一部分，它可以经历一些信号处理、非法攻击而存活下来。它为合法的版权所有者提供证明信息，以抵制侵权行为。

● 数据完整性的验证

数据完整性的验证是指要确认数据在网上传输或在存储过程中没有被篡改过。当数字作品被用于法庭、医学、新闻及商业时，常需要确认它们的内容是否被修改、伪造或特殊处理

过。利用脆弱性水印的特性，可以用它来做篡改提示，验证数字作品的可靠性。

● 加指纹

为避免未经授权的拷贝制作和发行，出品人可以将不同用户的 ID 或序列号作为不同的水印（指纹）嵌入作品的合法拷贝中，一旦发现未经授权的拷贝，就可以根据此拷贝所恢复出的指纹来确定它的来源。对这种应用领域来说，水印技术除具有版权保护应用中的特性外，还必须具有防止共谋攻击（多拷贝攻击）等功能。

● 拷贝保护

在数字产品发行体系中，希望存在这样的一个拷贝保护机制，即它不允许未经授权的媒体拷贝。这在开放系统中很难实现，但在封闭或私有系统中，它是可行的。在这样的系统中，可用数字水印来说明数据的拷贝状况。例如，在 DVD 防拷贝系统中，视频数据中含有拷贝信息，它限定了拷贝次数，不允许数据被无限制地制作拷贝。

● 加标题与注释

即将数字作品的标题、注释、作者等内容（如，一幅照片的拍摄时间、地点、摄影师名字等）以水印形式嵌入该作品中，这种隐式注释不需要额外的带宽，且不易丢失。上海的瑞金医院在管理病人的 X 光片时就采用了数字水印的方法。

随着数字水印技术的不断成熟与发展，它的应用领域也会不断扩展。著名的 Adobe 公司已经在其知名的 Acrobat Reader 软件中加入了数字水印来标识其中的图像和文本。相信在不远的将来，会有更多的人致力于挖掘数字水印技术潜在的商业价值。

3.5 数字水印的技术要求

正如上节所提到的，水印技术的应用很广。可以理解的是，不同的应用场合对技术的要求也将各不相同。因此，对所有的水印算法没有统一的技术标准，本节所提及的各种技术要求适合于各种不同的水印算法。

◆ 不可感知性

适用于版权保护、拷贝保护等。要求数字媒体在嵌入水印后不改变其感知效果，即嵌入的水印在通常的感知条件下不被感觉。如利用人类视觉系统或人类听觉系统的属性，经过一系列隐藏处理，使目标图像、视频和音频没有明显的降质现象，而隐藏的数据无法人为地看見或听见。

在图像或视频水印中，不可感知性通常使用“盲检”的方法来验证。方法是随机抽取一定数量的样本图像，其中加入水印的图像和未加入水印的图像各占总数的 50%，让别人通

过主观的判断, 哪些图像是加入了水印而哪些图像没有加入水印, 如果判断的结果两种图像也各占 50% 或十分接近 50%, 那么证明水印算法的不可感知性是具备的。

除了主观判断以外, 还经常使用峰值信噪比 (PSNR) 来衡量不可感知性, PSNR 的定义^[7]如下:

$$PSNR(f, w) = 10 \log_{10} \left[\frac{\max_{(m,n)} f^2(m,n)}{\frac{1}{N_f} \sum_{(m,n)} (f_w(m,n) - f(m,n))^2} \right] \quad (3.1)$$

PSNR 的单位是 DB, f 为载体数据, w 为水印信息, f_w 为嵌入了水印的数据。 (m,n) 为像素坐标, N_f 为 f , f_w 的像素个数。使用 PSNR, 可能不是十分准确, 但它无疑是检验水印不可见性的有效尺度。

◆ 鲁棒性

指因数字媒体的某种改动而导致水印信息的丢失。所谓的改动包括传输过程中的信道噪声、线性或非线性滤波、重采样、有损压缩编码、数/模或模/数转换;

◆ 提取或检测水印时需不需要原始数据

这就是 3.3 节所说的“明文水印”与“盲水印”的区别。一般以为, 盲水印方法在提取或检测水印时无法访问原始数据, 会使水印信息的嵌入量受到限制。而且会大大提高特别是提取水印的难度。然而, 有些应用场合, 必须要求“盲水印”的方法, 原因是无法获得原始数据, 象视频水印, 将原始数据保留下往往是不现实的。本文第四章所提出的算法, 就是“盲水印”。

◆ 自相似性

数字水印在原图像遭到较大的破坏后, 仍能从原数据中恢复出隐藏的数字水印, 而且在恢复中不需原图像, 这需要数字水印算法本身具有自相似性。

◆ 码率的要求

其实这里考虑的是加入水印数据量的问题。有些水印系统使用的是序列号、作者姓名等水印信息, 信息量很小。而有些水印系统使用灰度图像、彩色图像、音频或视频作为水印, 那水印的信息量就很大了。这样势必要提高传输码率。一般要求, 水印算法尽可能少或不要提高原始数据的传输码率。

◆ 安全性

指水印算法有较强的抗攻击能力，即它必须能够承受一定程度的人为攻击，而使水印信息不遭到破坏。数字水印系统可以引入一个或多个密钥来确保安全。哪怕是攻击者知道了水印嵌入算法过程，如果他没有密钥，也同样无法破坏或删除水印。第四章中将详细介绍水印的置乱技术，它是确保安全性的常用手段。

3.6 数字水印的攻击方法^{[3][9][24]}

对水印的攻击方法大致可分为四类：鲁棒性攻击、表达攻击、解释攻击和合法攻击。

◆ 鲁棒性攻击

这类攻击其实是直接攻击，目的在于擦除媒体数据中的水印而不影响媒体数据的使用。大体上可再细分为两种类型：信号处理攻击法和分析（计算）攻击法。

信号处理攻击法包括无恶意的常用的一些信号处理，如压缩编码、滤波、缩放、打印及扫描等。分析（计算）攻击法包括在水印的插入和检测阶段采用特殊方法来擦除或减弱媒体中的水印。

针对静止图像的共谋攻击是一类典型的分析攻击法，它使用同一幅图像嵌入了不同水印后的不同版本组合而产生一个新的“嵌入了水印”的图像，从而减弱水印的强度。

许多水印方案都易于遭到共谋攻击。为了说明共谋攻击的危害，在这里举一个例子，先考虑这样的水印方案：在 f_i 随机地加上由 1 或 -1 构成的水印而得到 f'_i 。对于这种水印方案，一旦发现两个文档中有不同的 f'_i ，便可得到 f_i 的值，从而可以完全消除水印的这个分量。如果采用这种水印方案的 t 个文档，将这 t 个文档求平均就几乎可以擦除所有的水印分量而只剩下 2^{1-t} 部分未擦除。

◆ 表达攻击

这类攻击有别于鲁棒性攻击之处在于它并不需要除去数字媒体内容中嵌入的水印，它是通过操纵内容从而使水印检测器无法检测到水印的存在。例如，表达攻击可简单地通过不补齐一个嵌入了水印的图像来愚弄自动水印检测器（如：基于 Web 的智能代理或 WebCrawler 等），实际上在这种攻击中并未改变任何图像像素值。

剑桥大学计算机实验室的 F. Petitcolas 提出的攻击方法就是一个很好的表达攻击的例子，该攻击方法目的是挫败 WebCrawler。它将一个嵌入了水印的图像切成许多小块，这些小块在 Web 页上按相应的 HTML 标记再组装起来。WebCrawler 只能去查看每个图像小块，但这些小块由于太小而无法容纳任何水印数据，所以 WebCrawler 无法发现水印。该攻击方法实

际上并未导致任何图像质量的下降，因为图像像素值被完全保留了。

还有一种所谓的“跳跃攻击”，也属于表达攻击的范畴。跳跃攻击主要用于对音频信号数字水印系统的攻击。其一般实现方法是，在音频信号上加入一个跳跃信号(*jitter*)，即首先将信号数据分成 500 个采样点为一个单位的数据块，然后在每一数据块中随机复制或删除一个采样点，来得到 499 或 501 个采样点的数据块，接着再将数据块按原来的顺序重新组合起来。实验表明，这种改变即使对古典音乐信号数据也几乎感觉不到，但是却可以非常有效地阻止水印信号的检测定位，以达到难以提取水印信号的目的。类似的方法也可以用来攻击图像数据的数字水印系统，其实现方法也非常简单，即只要随机地删除一定数量的像素列，然后用另外的像素列补齐即可，该方法虽然简单，但是仍然能有效破坏水印信号存在的检验。

◆ 解释攻击

在一些水印方案中，可能存在检测出的水印的多个解释。例如，一个攻击者试图在同一个嵌入了水印的图像中再次嵌入另一个水印，该水印有着与所有者嵌入的水印相同的强度。由于一个图像中出现了两个水印，所以导致了所有权的争议。在解释攻击中，图像的像素值或许被改变或许不被改变。此类攻击往往要求对所攻击的特定的水印算法进行深入彻底的分析。

著名的 IBM 攻击就是解释攻击的典型例子。IBM 攻击利用水印方案的可逆性，其原理为，设原始图像为 I ，加入水印 W_A 的图像为 $I_A = I + W_A$ 。攻击时，攻击者首先生成自己的水印 W_F ；然后创建一个伪造的原图 $I_F = I_A - W_F$ ，也即 $I_A = I_F + W_F$ ；此后，攻击者可声称他拥有 I_A 的版权，因为攻击者可利用其伪造原图 I_F 从原图 I 中检测出其水印 W_F ，但原作者也能利用原图从伪造原图 I_F 中检测出其水印 W_A ，这就产生无法分辨与解释的情况。而防止这一攻击的有效方法就是研究不可逆水印嵌入算法，如 Cox I. J. 提出的 NEC 算法。

◆ 合法攻击

这类攻击同前三类攻击都不同，前三类可归类为技术攻击，而合法攻击则完全不同。攻击者希望在法庭上利用此类攻击，它们的攻击是在水印方案提供的技术优点或科学证据的范围之外进行的。合法攻击可能包括现有的及将来的有关版权和有关数字信息所有权的法案，因为在不同的司法权中，这些法律有可能有不同的解释。合法攻击还可能包括所有者和攻击者的信用，攻击者使法庭怀疑数字水印方案有效性的能力。除了这些之外，可能还和其他一

些因素密切相关，如：所有者和攻击者的金融实力的对比、专家的证词、双方律师的能力等。

理解和研究合法攻击要比理解和研究技术上的攻击困难得多。作为一个起点，我们首先应致力于建立一个综合全面的法律基础设施，以确保正当的使用水印和利用水印技术提供的保护，同时，避免合法攻击导致降低水印应有的保护作用。合法攻击是难以预料的，但是一个真正优秀的水印方案必须具有这样的优点：攻击者使法庭怀疑数字水印方案有效性的能力降至最低。

深入研究上述四类攻击手段是有意义的，因为这样做会使我们清晰地意识到，数字水印技术依然存在着不少缺陷和局限性，同时又为我们提供了研究的方向与目的。当然，要求某一水印系统能抵挡住所有的攻击是不现实的，我们应该从具体应用的要求出发，来寻找重点克服一种或几种攻击的算法。

3.7 静止图像水印的典型算法分析

从数字水印研究的现状来看，绝大部分人的注意力都放在了静止图像水印上，而相比之下，对于音频水印、视频水印的研究就不是很多了。本节将介绍几种典型的静止图像水印算法，从中可以反映出水印算法的基本思路和方法，这对视频水印等其他数字媒体的水印算法有指导和启发的作用。

根据实现的过程，静止图像水印算法主要可分为两类：空域算法和变换域算法。下面分别作介绍。

3.7.1 空域算法

空域算法通过直接改变宿主图像某些像素值来嵌入水印。这类算法包括文本水印算法、Schyndel 算法和 Patchwork 算法等。其中 Schyndel 在 ICIP'94 会议上发表的题为 “A digital watermark” 的文章被认为是一篇具有历史意义的文献，它是第一篇在主要会议上发表的关于数字水印的文章，文章阐明了一些关于水印的重要概念和鲁棒水印检测的通用方法（相关性检测法），该文提出的 Schyndel 算法首先把一个密钥输入到一个 m -序列发生器来产生水印信号，然后此 m -序列被重新排列成 2 维水印信号，并按像素点逐一插入到原始图像像素的最低有效位 (Least significant bits — LSB)。该方法的主要缺点是抗 JPEG 压缩的鲁棒性不好，易受攻击。Patchwork 提出了一种基于改变图像数据统计特性的水印算法，该算法首先随机选取 N 对像素点，然后通过增加像素对中的一个点的亮度值，而相应降低另一个点亮度值的方法来隐藏信息，为增加其水印的鲁棒性，他还把像素对扩展为小块的像素区域（如 8×8 ），再通过增加一个区域中的所有像素点的亮度值，而相应减少对应区域中所有像素点

的亮度值的方法来隐藏信息，但该方法嵌入的水印码率低，且对表达攻击（具体地说是串谋攻击）的抵抗力弱。

除了上述两种方法之外，Bruyndoncky 等人提出了一个基于空域分块的方法，通过改变块均值来嵌入水印。Nikolaidi 等人根据一个二进制伪随机序列，把图像中的所有像素分为两个子集，改变其中一个子集的像素来嵌入水印。空域上的水印算法虽然比较简单，但是鲁棒性较差，易于受到攻击，现在已经研究的不多。

3.7.2 变换域算法

变换域上的水印算法对有损压缩和其他信号处理具有较强的抵抗力，因此比空域算法更具优势，目前已占据了主导地位。下面先介绍一下数字水印常用的变换域，它们是算法的基础。

3.7.2.1 数字水印常用的变换域

➤ 离散傅立叶变换 (DFT) 域

一般将图像看作二维信号，用 $g(i, k)$ 表示，图像大小为 $N \times N$ 。则它的二维离散傅立叶变换 (DFT) 为

$$G(m, n) = \frac{1}{N} \sum_{i=0}^{N-1} \sum_{k=0}^{N-1} g(i, k) e^{-j2\pi(m\frac{i}{N} + n\frac{k}{N})} \quad (3.2)$$

逆 DFT (反变换) 为

$$g(i, k) = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} G(m, n) e^{j2\pi(i\frac{m}{N} + k\frac{n}{N})} \quad (3.3)$$

➤ 离散余弦变换 (DCT) 域

二维离散余弦变换 (DCT) 定义如下

$$G_c(m, n) = a(m)a(n) \sum_{i=0}^{N-1} \sum_{k=0}^{N-1} g(i, k) \cos\left[\frac{\pi(2i+1)m}{2N}\right] \cos\left[\frac{\pi(2k+1)n}{2N}\right] \quad (3.4)$$

逆变换为

$$g(i, k) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a(m)a(n) G_c(m, n) \cos\left[\frac{\pi(2k+1)m}{2N}\right] \cos\left[\frac{\pi(2k+1)n}{2N}\right] \quad (3.5)$$

其中系数为

$$a(0) = \sqrt{\frac{1}{N}}, \text{ 且 } a(m) = \sqrt{\frac{2}{N}}, \quad 1 \leq m \leq N$$

➢ 离散小波变换(DWT)域

二维的离散小波变换将在下一章详细介绍，可参阅 4.1 节。

➢ 分形(the fractal)域

分形域指的是分形图像压缩中定义域块的集合。具体的分形图像压缩算法可参看文献 [18]，这里仅作简要介绍。

Jacquin 的分形图像压缩方案的编码步骤如图所示：

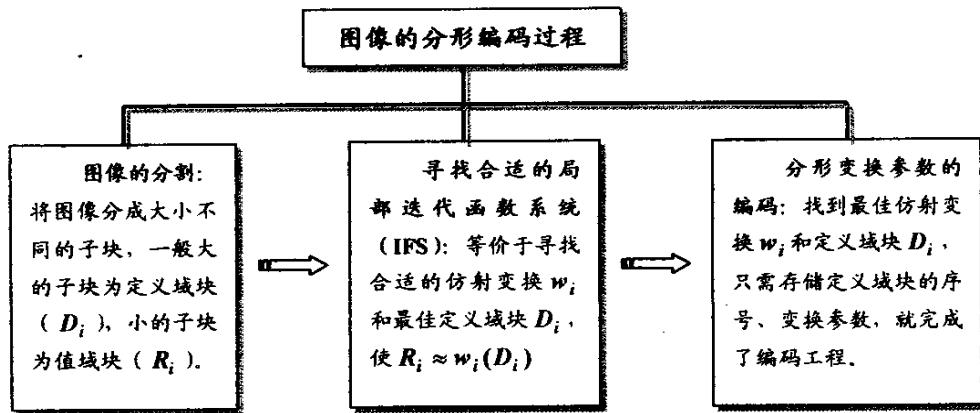


图 3.5 Jacquin 的分形编码过程

解码过程十分简单，只需要将存储的参数形成 IFS 代码，然后对相应的定义域块作有限次迭代就可以了。分形编码能获得很高的压缩比（大致在 20 至 50 倍之间），是一种很有前途的压缩方法。

3.7.2.2 NEC 算法

Langelaar 等人采用与调整了的 LSB 类似的做法，把水印嵌入图像频域的高频系数中，然而，Cox 等人却提出了一个不同的观点：水印应放在视觉上最重要的地方（主要对应于频域的低频系数）。这就是 NEC 算法的基本思想。

水印为什么要嵌入到低频系数中呢？当时 Cox 的想法可谓独树一帜，因为感觉上重要的分量是图像的主要成分，携带较多的信号能量，在图像有一定失真的情况下，仍能保留主要成分，作为水印的载体，有利于提高水印的鲁棒性。那么嵌入低频系数的水印会不会影响图像的质量呢？Cox 采用了扩频通信的思想来解决这个问题。

扩频通信是一种信息传输方式，其信号所占有的频带宽度远大于所传信息必需的最小带

宽；频带的扩展是通过一个独立的码序列来完成，用编码及调制的方法来实现的，与所传信息数据无关；在接收端则用同样的码进行相关同步接收、解扩及恢复所传信息数据。扩频通信的模型如下所示：

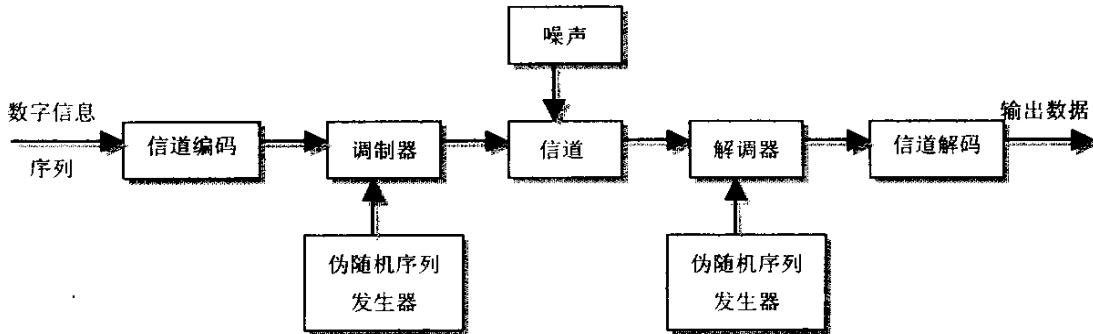


图 3.6 扩频通信模型

扩频通信看上去“浪费”了带宽，但它却获得了其他方面的好处。信息论中的仙农 (Shannon) 公式为：

$$C = W \log_2 (1 + P/N) \quad (3.6)$$

式中：

C --- 信道容量(用传输速率度量)

W --- 信号频带宽度

P --- 信号功率

N --- 白噪声功率

式 3.6 说明，在给定的传输速率 C 不变的条件下，频带宽度 W 和信噪比 P/N 是可以互换的。即可通过增加频带宽度的方法，在较低的信噪比 P/N (S/N) 情况下传输信息。扩展频谱以换取信噪比要求的降低，正是扩频通信的重要特点。

Cox 将数字水印系统看作是扩频通信的实例。将图像的 DCT 系数作为“信道”，水印信息就是其中传输的信号，水印信号是实数序列，并服从 $N(0,1)$ 分布，它相当于扩频通信中被展宽频谱的信号。而有损压缩、滤波等信号处理以及一些对水印的恶意攻击被当作是“噪声”。水印的嵌入可采用下面的式子：

$$f_i' = f_i (1 + \alpha w_i) \quad (3.7)$$

其中 f_i 为图像的 DCT 系数，Cox 选用了除直流分量外的 1000 个低频系数来嵌入水印， w_i 为水印信息，长度为 1000， α 为水印强度，要求 α 与图像的失真程度成反比， f_i' 为嵌

入了水印的图像 DCT 系数。嵌入过程如下图所示。

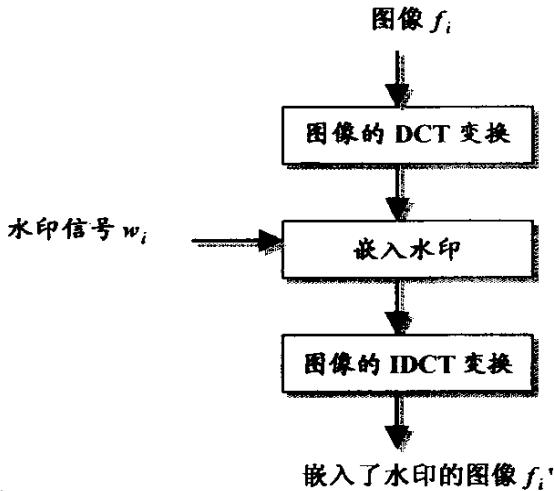


图 3.7 NEC 算法：水印的嵌入过程

水印的检测过程较为简单，将 f_i' 作 DCT 变换，取出前 1000 个低频系数，与原图像对应位置上的 DCT 系数作差，得到 w_i^* ，计算 w_i^* 与 w_i 的相似度：

$$sim(w_i, w_i^*) = \frac{w_i, w_i^*}{\sqrt{w_i, w_i^*}} \quad (3.8)$$

如果 $sim(w_i, w_i^*) > T$ (T 为门限值)，则水印 w_i 存在，否则， w_i 不存在。

NEC 算法在变换域水印中占据了重要的地位，它的思想到现在还不断被人采用。

3.7.2.3 基于二维离散 DWT 和数据融合的水印算法

本节简单介绍另一种数字水印算法，它是基于小波变换和数据融合的。水印图像是二值图像，大小为 $2N_x \times 2N_y$ ，载体图像的大小为 $2^{M-1} \times 2N_x \times 2^{M-1} \times 2N_y$ 。

基于鲁棒性的考虑，水印图像的统计特性应类似于白噪声，其均值为 0。所以首先将二值图像中的 0 和 1 值变换为 -1 和 +1。然后对水印图像作一次小波分解，得到 4 个大小为 $N_x \times N_y$ 的子块，它们中 LL_1 、 LH_1 、 HL_1 子带集中了水印的绝大部分能量。定义每一子块为 $w_{1,k}^i(x, y)$ (其中 $k = 1, 2, 3$)。载体图像作 M 次小波分解后的最小子块的大小和 $w_{1,k}^i(x, y)$ 的大小相同。

载体图像作 M 次小波变换后，除了 LL_M 外，将任一层 m ($m = 1, 2, \dots, M$) 的每个子带图

像 $k (k=1,2,3)$ 分为不相重叠的 $N_x \times N_y$ 大小的子块 $f_{m,k}^i(x,y)$ 。水印的嵌入方法类似于数据融合的过程，其方法如下面的公式：

$$g_{m,k}^i(x,y) = f_{m,k}^i(x,y) + \alpha_{m,k} w_{1,k}^i(x,y) \quad (3.9)$$

该式中， $\alpha_{m,k}$ 为加权因子，它必须根据 HVS 的特性来计算，力求水印的加入不会影响图像原来的视觉效果。计算 $\alpha_{m,k}$ 的具体方法，可参阅文献。

水印提取时，可将嵌入了水印的图像按上述嵌入算法的反过程和原始图像相比来提取水印。提取出的水印可通过直接观察来比较提取出的水印图像与实际水印的相似度，也可用类似 3.8 的公式来计算相似度，从而做出较为客观的评价。

3.7.2.4 基于分形图像编码的数字水印算法^{[19][20][21]}

3.7.2.1 中对分形图像编码的过程作了简要地介绍，利用分形编码中值域子块的最佳匹配定义域块的位置不变性，根据最佳匹配定义域块所处的寻找区域也可以来嵌入和提取水印信息。

设 $W = \{w_0, w_1, \dots, w_{N-1}\}$ 是 N 比特的水印，如 3.7.2.1 所述，嵌入水印的过程分为编码和解码两个过程。设 A 、 B 为两个局部寻找区域 (LSR)，把它们的总和定义为 $LSRC$ ，如图 3.8：

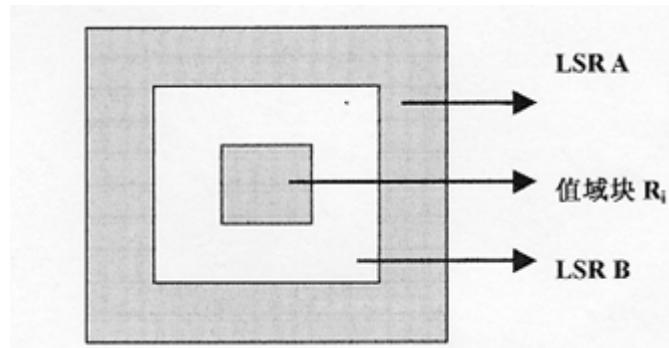


图 3.8 值域子块 R_i 与它的 $LSRA$ 和 $LSRB$

则编码过程为：

- 1) 对于每个比特 w_i ，值域子块 R_i 是随机选取的；
- 2) 如果 $w_i = 1$ ，值域子块 R_i 在 $LSRA$ 中寻找它的匹配定义域块 D_i ；
- 3) 如果 $w_i = 0$ ，值域子块 R_i 在 $LSRB$ 中寻找它的匹配定义域块 D_i ；

4) 其他的值域子块 R_i 在 $LSR\ C$ 中寻找它的匹配定义域块 D_i 。

提取水印时, 通过给定值域子块的最佳定义域块的位置来确定水印:

- 1) 如果最佳定义域子块属于 $LSR\ A$, 则嵌入的水印为 1;
- 2) 如果最佳定义域子块属于 $LSR\ B$, 则嵌入的水印为 0.

3.8 安全数字水印体系模型

在目前开放的 *Internet* 的环境下, 数字水印技术存在于一个复杂的应用环境之中, 不同的数字产品所有者对嵌入的水印有不同的要求。例如, 有的所有者目的是保护版权, 需要嵌入鲁棒性强的水印; 而有的所有者要保护数字产品的真实性, 需要嵌入脆弱的不可见的水印。

在具体的使用水印过程中, 也存在着以下的几种情况:

(1) 水印的嵌入和提取(或检测)可由任何用户进行, 无需水印密钥。这类水印的安全性全部依赖于水印算法的秘密性, 一旦算法公开了, 水印极易被擦除。

(2) 水印的嵌入和提取(或检测)都由所有者进行, 所有者选择并且保留密钥。这会给人造成黑匣子的感觉, 很难让局外人信服。

(3) 水印的嵌入由所有者进行, 所有者选择并保存水印密钥。提取或检测则由公众进行。当发生版权纠纷时, 由所有者出示水印密钥来检验水印的存在。但是水印密钥一旦公开, 就会被人利用来除去水印, 所以每次验证后, 还需要更换密钥, 重新嵌入水印, 诸多不便。

(4) 水印的嵌入和验证由可信任第三方进行。水印密钥的选择由可信任的第三方和所有者共同决定并保存。为了更有效地实现数字产品版权和真实性、完整性的保护, 必须采用一套严密完整的体系标准, 规定网络上利益联系的实体、可信任第三方, 嵌入和验证水印的实体、各个实体的责任、应遵守的协议等, 即安全数字水印体系。

文献[55]借鉴了 *IMPRIMATUR*¹ 体系结构和 *DHWM* 协议² 的思路和方法, 提出了改进的安全数字水印体系模型, 如图所示:

¹ *IMPRIMATUR* 是由欧洲委员会 DGIII 制定的网络数字产品的知识产权 IPR (Intellectual Property Rights) 认证和保护体系标准。

² *DHWM* 协议由欧洲文化数据库 AQUARELLE 所提出并已被 AQUARELLE 采用, 它结合了数字水印技术和密码学领域的认证体制, 为用户提供了一个较完善的 IPR 保护机制、协议模型。

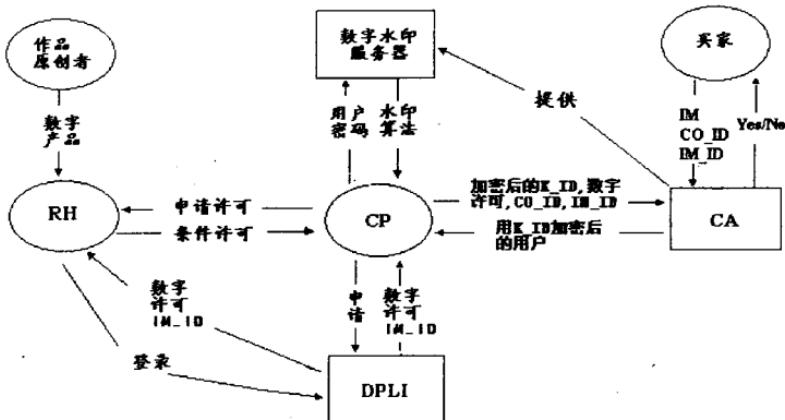


图 3.9 安全数字水印体系模型

图中各个角色的意义是：

- *RH*: (*Rights Holder*): 版权所有者。
- *CP* (*Creation Provider*): 网络数字产品的提供商。*CP* 应该是得到授权的合法数字产品提供商。
- *CA* (*Certification Authority*): 认证中心, 是数字水印检验机构, 同时负责提供水印算法和分发水印密钥。
- *IM*: 原始图像。未嵌入水印的数字产品。
- *IM_ID*: 唯一标识 *IM* 的序列。在 *IMPRIMATUR* 中 *IM_ID* 是由产品序列号分发机构 (*UNI*) 统一发布。
- *CO_ID*: 由 *CO* (数字产品版权所有者 *Copyright Owner*) 自定义的标识, 用来唯一标明身份。
- *K_ID*: 用来嵌入和提取 (或检测) 水印的密钥, 不同的数字产品有不同的 *K_ID*。
- *DPLI*: 将 *IMPRIMATUR* 中的 *UNI* 和 *IPR* 信息数据库¹的功能结合在一起, 由一个机构统一管理, 称为 *DPLI* (*Digital Product License Issuer*)。*DPLI* 负责对申请 *IM_ID* 的 *CO* 进行身份验证, 并登记版权所有者的版权信息、授权日期和权限等, 还保存该 *CO* 的数字产品信息, 如产品名称、类别、生产日期等。*DPLI* 接着为每个 *CO* 生成一个序列号即 *IM_ID*。为了防止攻击者通过直接与 *CA* 交互嵌入水印来寻找漏

¹ *UNI* (*Unique Number Issuer*): 产品序列号分发机构, 主要负责为数字产品产生唯一的标识序列号。

洞, *DPLI* 将对所有合法的 *CO* 发布许可证书。具体实现上, 可以采用数字签名技术。*CO* 必须向 *CA* 出示 *DPLI* 签名的许可才能得到 *CA* 提供的水印服务。

- 数字水印服务器: 由 *CA* 提供一个水印算法的服务器, 该服务器上存放着水印算法都是经过 *CA* 认定的安全算法。只有合法的用户才能从该服务器上下载水印算法。这些合法用户就是经 *CA* 核查无误的 *CO*。*CA* 向每个合法的 *CO* 提供一个账号和密码, 以便该 *CO* 登录水印服务器。*CA* 在向 *CO* 传递账号和密码前可以先用该 *CO* 的 *K_ID* 加密。水印服务器可以根据 *CO* 的要求提供多样的水印算法。如用于验证版权的鲁棒性水印、用于验证真实性的脆弱性水印、用于标识数字产品的可见水印、用于提供侵权证据的不可见水印等。

在进行水印密钥的传递时, 这种模型采用 *DHWM* 水印协议的方法, 并加以改进。具体是采用公钥加密体制(如: *RSA*)。*CO* 选定 *K_ID* 后, 将 *K_ID*, *CO_ID* 用 *CA* 的公钥加密再传给 *CA*。*CA* 收到后, 用自己的私钥解开, 核查无误后保存在数据库中, 然后生成一个确认信息, 用 *K_ID* 加密后回送给 *CO*。这样完成了用户的认证过程。

从更全面的角度看, 仅仅研究数字水印算法是不够的, 还必须配合安全体制的制定与完善, 这样做才能真正发挥作用。本小节介绍的数字水印体系模型有效地引入了加密技术和用户认证体制, 为 *IPR* 的保护提供了有效的解决方案。

3.9 视频水印的研究现状

目前绝大多数的数字水印算法都是针对静止图像的, 而对视频图像水印技术的研究还较少。与静止图像相比, 视频水印除了要满足静止图像水印的要求之外, 还有更高的要求。首先, 要满足“盲水印”的要求, 即水印的提取或检测不需要使用原始的视频图像。在视频水印中, 水印信息是周期性地嵌入到图像序列中去的, 如果不满足盲水印的要求, 每次进行水印的提取或检测都要求存储原始视频图像, 这是不现实的。其次, 视频水印算法应该考虑到视频图像序列运动的特点, 设计算法时充分利用人眼对物体运动区域和静止区域的不同敏感度, 以达到水印不可感知的要求; 另外, 视频水印算法的运算复杂度应尽可能地低, 能满足实时性的要求。

视频水印算法大致可以分为两类, 图 3.10 显示了它们的基本框图:

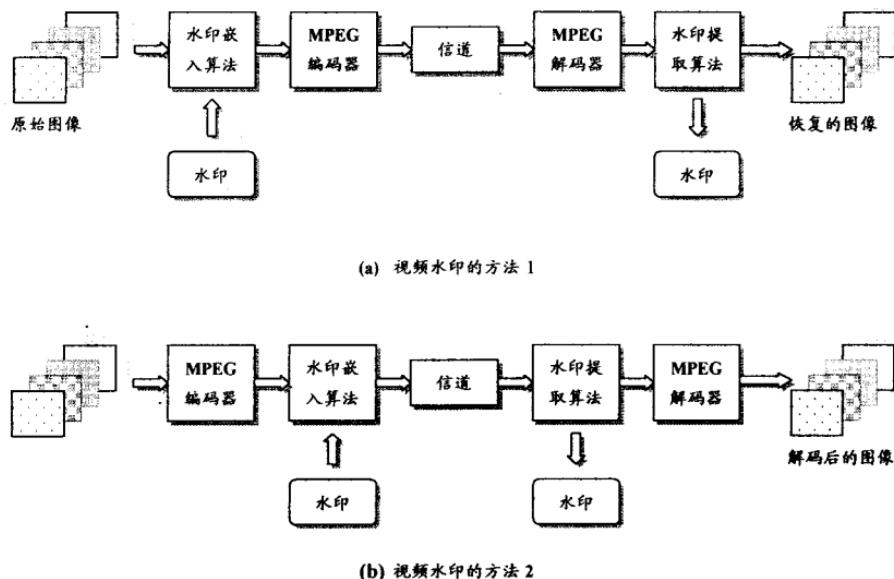


图 3.10 视频水印的两种主要方法

比较上图的 (a) 和 (b) 发现, 在第 1 类方法中, 水印嵌入在未经 MPEG 压缩的原始帧图像中, 而第 2 类方法中, 水印嵌入在经过压缩的 MPEG 码流中。

不因简单地把第 1 类方法当作是静止图像水印算法在视频中的应用, 因为视频与静止图像相比, 最大的特点就是视频包含运动的部分, 每一帧图像都发生着变化。统一的水印嵌入策略对一段视频的所有图像未必都适用, 比如随着视频中场景的变换, 加入水印的图像的质量可能会变坏, 原来不可见的水印就可能被人发现。再则, 人眼对运动区域的空间敏感度不如对静止区域的空间敏感度强, 这样的特点也可以被利用。由此可见, 好的视频水印算法应该充分考虑视频图像序列在时间维度上变化的特点, 参考人类视觉系统, 使水印信息更加有效地嵌入其中。

第 2 类方法也可称为基于压缩域的水印算法, 这是针对前文所讲的空域算法、频域算法来说的。视频图像经常以压缩的形式存储和传输, 如 MPEG1 数据流、MPEG2 数据流, 基于 MPEG 标准的压缩域数字水印算法在数字电视广播及 VOD 中有一定的实用价值。

拿 MPEG2 数据流为例, 它可分为数据头信息、运动矢量和 DCT 编码信号块三部分。任何一个部分都可以被嵌入水印。但在经过压缩的数据流中嵌入水印, 有许多不利的地方。

首先考虑到水印的加入不能改变原来数据传输码率，因此可以嵌入的水印信息量就十分有限。其次在嵌入水印中，假如改变的是 DCT 系数的 *Huffman* 编码值，这相当于引入了一个误差信号，而在嵌入时很难判断这种误差信号会对最终解码出来的视频产生什么样的影响，事实上要根据数据流来调整嵌入策略是十分困难的。假如改变的是运动矢量，由于它在压缩编码中是用于运动补偿，对它的改变（这种改变相当于迭加噪声）将会引起误差的扩散和累积，从而引起视觉变形。再则基于压缩域的水印必须依赖于之前对视频采用的压缩算法，一旦重新对视频作压缩编码，水印信息也很有可能会随之丢失，要重新嵌入一次。

综合上述考虑，作者决定采用第一种方法来设计视频水印算法。下一章将系统地介绍一种基于离散小波变换的自适应视频水印算法。

第四章 彩色视频水印算法的设计

本章将给出一种新颖的视频水印嵌入与提取算法。此种算法的特点是，第一，它嵌入与提取水印都是在小波变换域中进行。先将彩色视频图像序列的 RGB 色彩表示转换成 YCbCr (亮度色度色度) 表示，再将表示帧序的 l 坐标代换成 z 坐标，于是可将其看成是三维空间中的“体”，将其分成三维子块后，对需要嵌入水印的块进行可分离的三维小波变换，这样做的目的是避免对整个视频图像序列作小波变换，从而降低了算法的运算复杂度；第二，嵌入的水印为 24bit 真彩色图像，它蕴含的信息量十分可观。本算法借鉴了 JPEG 压缩方案的思想，有效地压缩了水印的信息量，而提取出的水印依然是原水印图像很好的近似，实践证明它表现力强、给人的印象直观；第三，算法实现了“盲水印”的要求，即在水印的提取过程中无需原始视频图像的参与。一般以为盲水印实现的难度要大于明文水印。本章所论述的水印算法有较好的鲁棒性，能抵抗诸如 MPEG 压缩、各类噪声和线性滤波等常见的图像处理。

一般的水印算法包含水印结构、嵌入方式和提取方式等三个部分。本章将对这种视频水印的嵌入及提取算法作详细地介绍。第一节阐述该算法重要的理论基础，即多分辨率分析和小波变换。第二节将介绍对水印图像的处理方法，涉及信息的压缩，编码和置乱（加密）等操作。第三节给出了嵌入水印的具体方法。第四节讨论如何实现无需原始数据的水印提取。

4.1 小波变换和多分辨率分析的原理及其应用

4.1.1 小波变换的定义

文献[44]中定义的小波函数族如下式：

$$\psi_{a\tau} = \frac{1}{\sqrt{a}} \psi\left(\frac{t-\tau}{a}\right) \quad (4.1)$$

式 4.1 中， $\psi(t)$ 被称为基本小波或母小波 (*mother wavelet*) 函数。 $\psi(t) \in L^1 \cap L^2$ ，且满足容许条件：

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\psi(t)|^2}{\omega} d\omega < \infty \quad (4.2)$$

$\psi(t)$ 之所以是“小”的，是因为规定 $\psi(t) \in L^1$ ，知 $\int_R |\psi(t)| dt < \infty$ ，即 $\psi(t)$ 具有衰减属

性, 特别地, $\psi(t)$ 是局部非零的紧支函数; $\psi(t)$ 之所以是“波”, 是由式 4.2 的积分有界推知, 当 $\omega=0$ 时, $\psi(t)$ 的傅里叶变换必须为零, 即 $\int_R \psi(t) dt = 0$, 这说明 $\psi(t)$ 具有波动性。

式 4.1 中的 $a > 0$ 为尺度因子, 它的作用是将基本小波 $\psi(t)$ 作伸缩, a 越大 $\psi(\frac{t}{a})$ 越宽, $\psi_{a\tau}(t)$ 的幅度与 \sqrt{a} 成反比, 保证了能量的不变性。 τ 反映位移, 其值可正可负。

设 $x(t) \in L^2(R)$, 一维连续的小波变换可看作是 $\psi(t)$ 与 $x(t)$ 的内积, 如下式所示:

$$WT_x(a, \tau) = \frac{1}{\sqrt{a}} \int x(t) \psi^*(\frac{t-\tau}{a}) dt = \langle x(t), \psi_{a\tau}(t) \rangle \quad (4.3)$$

其逆变换为:

$$x(t) = C^{-\frac{1}{2}} \iint_{R^2} \frac{1}{a^2} WT_x(a, b) \psi(\frac{t-\tau}{a}) da d\tau \quad (4.4)$$

把尺度参数 a 与平移参数 τ 作离散化, 即 $a = a_0^j, \tau = k\tau_0, j, k \in \mathbb{Z}$, 一般 $a_0 > 1, \tau_0 > 0$ 。

式 4.1 的 $\psi_{a\tau}(t)$ 改写成 $\psi_{a_0^j, k\tau_0}(t)$, 有:

$$\psi_{a_0^j, k\tau_0}(t) = a_0^{-\frac{j}{2}} \psi(a_0^{-j} t - k\tau_0) \quad (4.5)$$

对应的离散小波变换 (DWT) 记作:

$$WT_x(a_0^j, k\tau_0) = \int x(t) \psi^*_{a_0^j, k\tau_0}(t) dt \quad j = 0, 1, 2, \dots, k \in \mathbb{Z} \quad (4.6)$$

其重构公式为

$$x(t) = \sum_{j \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} C_{j,k} \psi_{a_0^j, k\tau_0}(t) \quad (4.7)$$

其中, 离散化小波变换系数可表示为:

$$C_{j,k} = \int x(t) \psi^*_{a_0^j, k\tau_0}(t) dt = \langle x, \psi_{a_0^j, k\tau_0} \rangle \quad (4.8)$$

4.1.2 多分辨率分析

Mallat 在构造正交小波基时提出了多分辨率分析 (Multi-Resolution Analysis) 的概念, 从函数空间的概念上形象地说明了小波的多分辨率特性, 并给出了正交小波变换的快速算法, 即 Mallat 算法。

对于平方可积的函数 $x(t) \in L^2$ 看成是某一逐级逼近的极限。每级逼近都是用某一低通

滤波函数 $\phi(t)$ 对 $x(t)$ 作平滑的结果, 这里的 $\phi(t)$ 被称为尺度函数 (Scaling Function), 它也作逐级伸缩, 也即说用不同的分辨率或不同尺度来逐级逼近 $x(t)$, 这就是多分辨率分析的基本思想。它的主体分析路径是: 先从 L^2 的某个子空间出发, 并在此子空间中建立一个基底; 然后利用极为简单的变换, 将此基底扩充到 L^2 的其他子空间中去。其具体过程如下:

把空间作逐级二分解产生一组逐级包含的子空间:

$$\cdots, V_0 = V_1 \oplus W_1, V_1 = V_2 \oplus W_2, \dots, V_j = V_{j+1} \oplus W_{j+1}, \dots, \quad (4.9)$$

j 是从 $-\infty$ 到 $+\infty$ 的整数, j 值愈小空间愈大。

这种剖分是完整的, 意即:

当 $j \rightarrow -\infty$ 时 $V_j \rightarrow L^2(R)$, 包含整个平方可积的实变函数空间。在逐级包含的条件下, 式

4.9 等效于:

$$\bigcup_{j \in \mathbb{Z}} V_j = L^2(\mathbb{R}) \quad (4.10)$$

当 $j \rightarrow +\infty$ 时 $V_j \rightarrow \{0\}$, 即空间最终剖分到空集为止。在逐级包含的条件下, 式 4.9 等效于:

$$\bigcap_{j \in \mathbb{Z}} V_j = \{0\} \quad (4.11)$$

上述剖分方式显然保证了空间 V_j 与空间 W_j 正交, 且各 W_j 之间也正交。

进一步要求剖分还具有以下两项特性:

(I) 位移不变性: 函数的时移不改变其所属空间。即: 如 $x(t) \in V_j$, 则

$$x(t-k) \in V_j \quad (4.12)$$

(II) 二尺度伸缩性: 如 $x(t) \in V_j$, 则

$$x\left(\frac{t}{2}\right) \in V_{j+1}, \quad x(2t) \in V_{j-1} \quad (4.13)$$

在上述基础上对各子空间内的结构作进一步分析:

(I) 子空间 V_0 : 设 V_0 中有低通的平滑函数 $\phi(t)$; 它的整数移位集合 $\langle \phi(t-k), k \in \mathbb{Z} \rangle$ 是 V_0 中的正交归一基。根据正交归一性, 可见: $\int \phi(t) dt = 1$ 。因此 V_0 中的任意函数必可表为 $\langle \phi_{0k}(t), k \in \mathbb{Z} \rangle$ 的线性组合。也就是设 $P_0 x(t)$ 代表 $x(t)$ 在 V_0 上的投影, 则必有:

$$P_0 x(t) = \sum_k x_k^{(0)} \phi_{0k}(t) \quad (4.14)$$

其中 $x_k^{(0)}$ 是线性组合各权重，其值可如下求：把上式两边对 $\phi_{0k}(t)$ 作内积，由正交归一性使得：

$$x_k^{(0)} = \langle P_0 x(t), \phi_{0k}(t) \rangle = \langle x(t), \phi_{0k}(t) \rangle \quad (4.15)$$

$P_0 x(t)$ 称为 $x(t)$ 在 V_0 中的平滑逼近，也就是 $x(t)$ 在分辨率 $j=0$ 下的概貌。 $x_k^{(0)}$ 称为 $x(t)$ 在分辨率 $j=0$ 下的离散逼近。

(2) 子空间 V_1 ：如果 $\phi(t) \in V_0$ ，则根据二尺度伸缩性， $\phi(\frac{t}{2}) \in V_1$ 。而且如果 $\langle \phi_{0k}(t), k \in z \rangle$ 是 V_0 中的正交归一基，则 $\langle \phi_{1k}(t), k \in z \rangle$ 必为 V_1 中的正交归一基。因此 V_1 中的任意函数，例如 $P_1 x(t)$ ，必可表为 $\langle \phi_{1k}(t), k \in z \rangle$ 的线性组合：

$$P_1 x(t) = \sum_k x_k^{(1)} \phi_{1k}(t) \quad (4.16)$$

且权重

$$x_k^{(1)} = \langle P_1 x(t), \phi_{1k}(t) \rangle = \langle x(t), \phi_{1k}(t) \rangle \quad (4.17)$$

$P_1 x(t)$ 是 $x(t)$ 在 V_1 中的平滑逼近，也就是 $x(t)$ 在分辨率 $j=1$ 下的概貌。 $x_k^{(1)}$ 是 $x(t)$ 在分辨率 $j=1$ 下的离散逼近。

(3) 子空间 W_1 ：如果在子空间 W_0 中能找到一个带通函数 $\psi(t)$ ，其整数位移的集合 $\langle \psi(t-k), k \in z \rangle$ 构成 W_0 中的正交归一基，则同样根据二尺度伸缩性， $\psi(\frac{t}{2}) \in W_1$ ，且 $\langle \psi_{1k}(t) = \frac{1}{\sqrt{2}} \psi(\frac{t}{2} - k), k \in z \rangle$ 必构成 W_1 中的一组正交归一基。又由于 $\psi(t)$ 是带通函数，所以 $\int \psi(t) dt = 0$ 。

因此 W_1 中的任意函数必可表为 $\langle \psi_{1k}(t), k \in z \rangle$ 的线性组合。也就是说，设 $D_1 x(t)$ 代表 $x(t)$ 在 W_1 中的投影，则必有：

$$D_1 x(t) = \sum_k d_k^{(1)} \psi_{1k}(t) \quad (4.18)$$

且权重

$$d_k^{(1)} = \langle D_1 x(t), \psi_{1k}(t) \rangle = \langle x(t), \psi_{1k}(t) \rangle \quad (4.19)$$

因为 $V_0 = V_1 \oplus W_1$ ，所以有：

$$P_o x(t) = P_1 x(t) + D_1 x(t) \quad (4.20)$$

或

$$D_1 x(t) = P_0 x(t) - P_1 x(t)$$

也就是说 $D_1 x(t)$ 是 V_0 、 V_1 两级相邻平滑逼近之差，反映这两级逼近间的细节差异。因此称 $D_1 x(t)$ 为分辨率 $j=1$ 下的细节函数。 $d_k^{(1)}$ 是 $j=1$ 下的离散细节。值得注意的是由式 (4.19) 可见 $d_k^{(1)}$ 实际上就是离散的 $j=1$ 时的小波变换 $WT_x(j=1, k)$ 。 $\psi(t)$ 就是具有带通特性的离散小波函数。这样便把多分辨率分析与小波变换联系起来了。

上述讨论可以很自然地推广到 V_{j-1} 与 V_j 、 W_j 之间，即：

$$\langle \phi_{jk}(t) = \frac{1}{2^{j/2}} \phi(2^{-j}t - k); k \in \mathbb{Z} \rangle \text{ 必是 } V_j \text{ 中的正交归一基。}$$

$$\langle \psi_{jk}(t) = \frac{1}{2^{j/2}} \psi(2^{-j}t - k); k \in \mathbb{Z} \rangle \text{ 必是 } W_j \text{ 中的正交归一基。}$$

也就是：

$$\langle \phi_{jk}(t), \phi_{j'k'}(t) \rangle = \langle \psi_{jk}(t), \psi_{j'k'}(t) \rangle = \delta(k - k') \quad (4.21)$$

且

$$P_{j-1} x(t) = \sum_k x_k^{(j-1)} \phi_{j-1,k}(t), \quad x_k^{(j-1)} = \langle x(t), \phi_{j-1,k}(t) \rangle \quad (4.22)$$

$$P_j x(t) = \sum_k x_k^{(j)} \phi_{jk}(t), \quad x_k^{(j)} = \langle x(t), \phi_{jk}(t) \rangle \quad (4.23)$$

$$D_j x(t) = \sum_k d_k^{(j)} \psi_{jk}(t), \quad d_k^{(j)} = \langle x(t), \psi_{jk}(t) \rangle \quad (4.24)$$

又

$$P_{j-1} x(t) = P_j x(t) + D_j x(t) \quad (4.25)$$

$P_j x(t)$ 是 $x(t)$ 在 V_j 中的投影，也就是 $x(t)$ 在分辨率 j 下的平滑逼近， $x_k^{(j)}$ 是其离散逼近。

$D_j x(t)$ 是 $x(t)$ 在 W_j 中的投影，反映 $P_{j-1} x(t)$ 、 $P_j x(t)$ 两个平滑逼近间的细节差异。而其离散值 $d_k^{(j)}$ 就是小波变换 $WT_x(j=j, k)$ 。

以上就是从函数空间剖分角度讨论多分辨率分析的基本框架。

4.1.3 可分离的二维多分辨率分析

这里所说的“可分离”指的是在二维信号分析中，沿 x_1 和 x_2 两个方向分先后两步作串行处理。本节的思路是将一维多分辨率分析推广到二维来。

假设 $f(x_1, x_2)$ 是 $L^2(R^2)$ 二维实空间 $V_0^{(2)}$ 内的二维函数。再假设二维空间 $V_j^{(2)}(x_1, x_2)$ 是可分离的，即它可以分解成两个一维空间 $V_j^{(1)}(x_1)$ 和 $V_j^{(1)}(x_2)$ 的张量乘积：

$$V_j^{(2)}(x_1, x_2) = V_j^{(1)}(x_1) \otimes V_j^{(1)}(x_2) \quad (4.26)$$

如果低通的尺度函数 $\phi(x_1, x_2)$ 是 $V_j^{(2)}(x_1, x_2)$ 空间内的整数位移正交归一基，则 $\phi(x_1, x_2)$ 也必可分解为： $\phi(x_1, x_2) = \phi(x_1) \otimes \phi(x_2)$ 。

同理，如果小波函数 $\psi(x_1, x_2)$ 是补空间 $W_j^{(2)}(x_1, x_2)$ 中的整数位移正交归一基，则 $\psi(x_1, x_2)$ 必可分解成： $\psi(x_1, x_2) = \psi(x_1) \otimes \psi(x_2)$ 。

如果把 $V_j^{(2)}(x_1, x_2)$ 的补空间记作 $W_j^{(2)}(x_1, x_2)$ ， $V_j^{(1)}(x_i)$ [$i=1, 2$] 的补空间记作 $W_j^{(1)}(x_i)$ ，则有：

$$\begin{aligned} V_{j-1}^{(2)}(x_1, x_2) &= V_j^{(2)}(x_1, x_2) \oplus W_j^{(2)}(x_1, x_2) \\ V_{j-1}^{(1)}(x_i) &= V_j^{(1)}(x_i) \oplus W_j^{(1)}(x_i) \end{aligned} \quad (4.27)$$

由式 (4.26) 及 (4.27) 可得：

$$\begin{aligned} V_{j-1}^{(2)}(x_1, x_2) &= V_{j-1}^{(1)}(x_1) \otimes V_{j-1}^{(1)}(x_2) \\ &= [V_j^{(1)}(x_1) \oplus W_j^{(1)}(x_1)] \otimes [V_j^{(1)}(x_2) \oplus W_j^{(1)}(x_2)] \end{aligned}$$

展开后得：

$$\begin{aligned} V_{j-1}^{(2)} &= [V_j^{(1)}(x_1) \otimes V_j^{(1)}(x_2)] \oplus [W_j^{(1)}(x_1) \otimes V_j^{(1)}(x_2)] \\ &\quad \oplus [V_j^{(1)}(x_1) \oplus W_j^{(1)}(x_2)] \oplus [W_j^{(1)}(x_1) \oplus W_j^{(1)}(x_2)] \end{aligned} \quad (4.28)$$

由上式可见，它共分为四部分，第一部分为平滑逼近的低通空间，即：

$$V_j^{(2)}(x_1, x_2) = V_j^{(1)}(x_1) \otimes V_j^{(1)}(x_2)$$

其余三部分的组合反映的都是高通细节，即：

$$\begin{aligned} W_j^{(2)}(x_1, x_2) &= [W_j^{(1)}(x_1) \otimes V_j^{(1)}(x_2)] \oplus [V_j^{(1)}(x_1) \oplus W_j^{(1)}(x_2)] \\ &\quad \oplus [W_j^{(1)}(x_1) \oplus W_j^{(1)}(x_2)] \end{aligned}$$

以上分析结果说明，在可分离的情况下，二维多分辨率可分两步进行。先沿 x_1 方向分

别用 $\phi(x_1)$ 和 $\psi(x_1)$ 作分析, 把 $f(x_1, x_2)$ 分解成平滑逼近和细节这两部分, 然后对这两部分再沿 x_2 方向分别用 $\phi(x_2)$ 和 $\psi(x_2)$ 作类似的分析, 如图 4.1 所示:

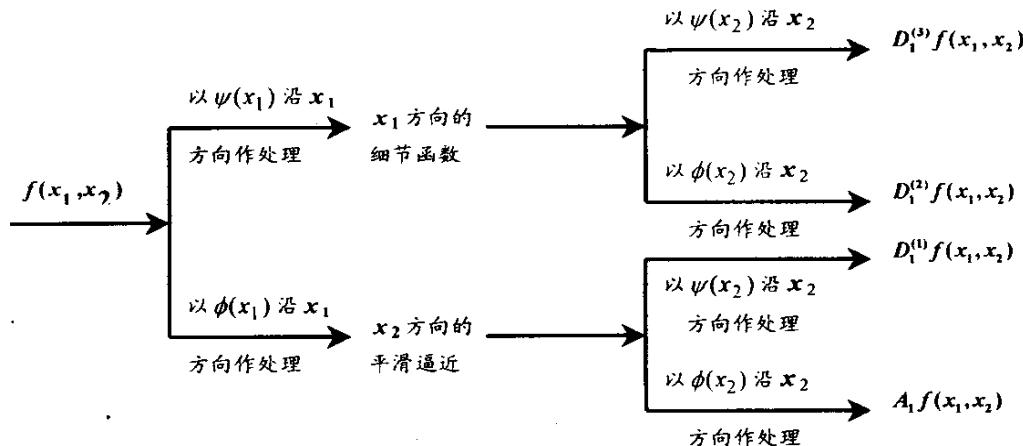


图 4.1 可分离情况下的多分辨率分析

对于图像的分解, 二维小波函数和尺度函数是通过一维小波函数和尺度函数经过张量积变换得到。二维小波分解是把尺度 j 的低频部分分解成四个部分: 尺度 $j+1$ 的低频部分和三个方向(水平、垂直、斜线)的高频部分, 如图 4.2 所示, 这个图的本质与图 4.1 相同。

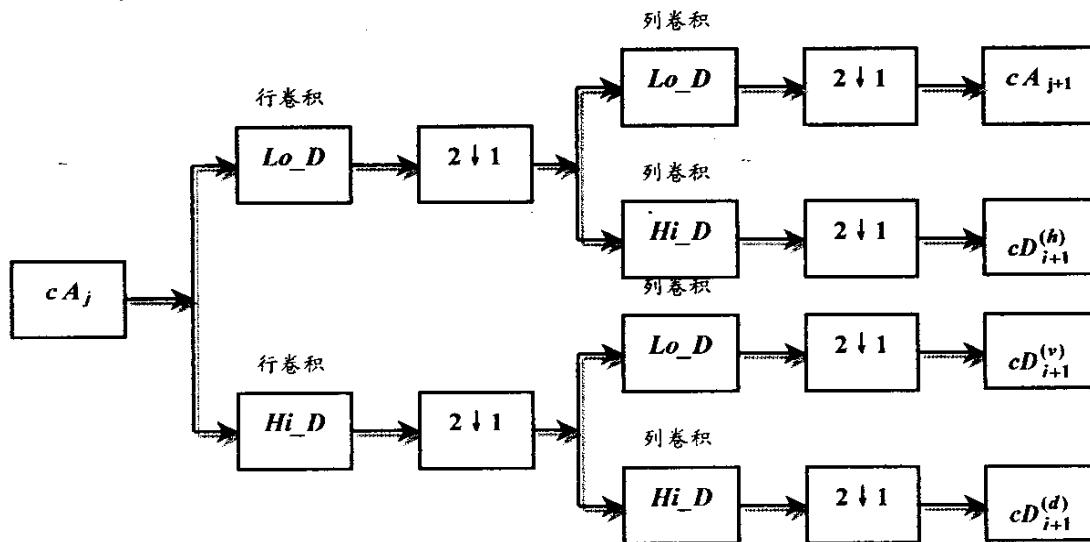


图 4.2 二维 DWT 的滤波器组结构

图 4.3 还给出了一幅图像经过了两级小波分解后的结果。



图 4.3 图像两级小波分解后的结果

4.1.4 视频图像序列的三维小波分解

本文所提出的水印嵌入算法是将水印信息嵌入到彩色视频图像序列的小波变换域中。这里使用的小波变换是三维可分离的离散小波变换（见文献[50]）。上一节介绍了多分辨率分析和小波变换的基本概念，也介绍了静态图像的二维可分离小波变换的实现。其实，在对视频图像序列做三维小波变换时，采取的方法本质上与二维小波变换相同，不同的只是除了分别在行(x_1 坐标方向)和列(x_2 坐标方向)上做一维小波变换之外，还要对不同帧相同位置的对应像素做一维小波变换。如图 4.4 所示，经过一次三维小波变换后的视频信号被分成 8 个波段，每次对变换所得的低频波段(LLL 波段)做一次变换又可得到 8 个新的波段。如果对视频信号做 N 次三维小波变换可得 $7 \times N + 1$ 个波段，经不同变换次数的波段内的系数数目是不同的，第 $k+1$ 次变换后，每波段的系数数目是第 k 次变换后每波段系数数目的 $\frac{1}{8}$ 。

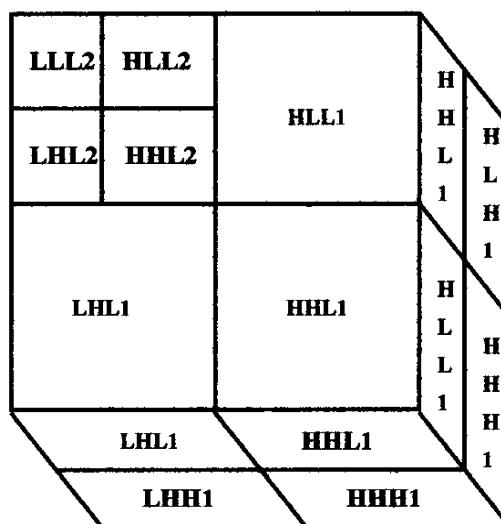


图 4.4 三维小波变换示意图

4.2 水印信息处理

4.2.1 水印图像的压缩与编码

本文的算法选用的是 24 bit 的真彩色图像作为水印。如图 4.5, 展示的是三幅可被用作水印的图像。



图 4.5 水印图像

由于水印技术的主要应用目的是版权保护, 所以这里选取了一些公司的商标或者组织的标志等。水印图像 W 的大小是 64×64 , 考虑到算法不可见性的要求, 其实它的信息量是相当大的。于是, 为了减少嵌入的数据量, 必须对 W 进行压缩。从压缩率和图像恢复质量两方面作衡量, 算法借鉴了 JPEG 压缩方案对图像进行压缩。具体步骤如下:

(1) 颜色分离: 将水印图像 W 做颜色分离, 它的每一个像素都可以分为 R (红色)、 G (绿色)、 B (蓝色) 三种基色分量, 那么 W 可分解成三幅基色图像 WR 、 WG 和 WB 。对每幅基色图像都采用相同的压缩和编码方法, 以 WR 为例, 我们实际上处理的是 256 级的灰度图像。

通过研究人类视觉系统 (HVS) 发现, 人眼对绿色 (G) 最敏感, 其次是红色 (R), 再次是蓝色 (B), 这一点从亮度方程上也可见一斑: $Y = 0.299 \times R + 0.587 \times G + 0.114 \times B$ 。在此处, 对水印图像进行颜色分离, 也是为了在嵌入过程中, 根据要求对 WR 、 WG 和 WB 等三幅基色图像采取不同的嵌入策略。

(2) 图像分块: 将水印图像划分成互不重叠的 8×8 的块, 记为 $B_m = f_m(x, y), m = 0, 1, \dots, 63$, 即:

$$WR = \sum_{m=0}^{63} f_m(x, y), \quad 0 \leq x, y \leq 7 \quad (4.29)$$

(3) DCT 变换: 对 B_m 作 DCT 变换, 二维 DCT 变换公式可参考 3.7.2.1 节。变换的结果是:

$$B'_m = F(u, v) = DCT\{f_m(x, y), 0 \leq x, y \leq 7\}, \quad m = 0, 1, 2, \dots, 63$$

8×8 的图像块经过了 DCT 变换后, 其低频分量都集中在左上角, 高频分量分布在右下角 (可以认为, DCT 变换起到了空间域低通滤波器的作用)。由于低频分量包含了图像的概貌信息, 而高频分量包含的是细节信息, 为了得到近似图像, 我们忽略高频分量, 从而达

到压缩的目的。

(4) **量化:** 在 JPEG 压缩中, 使用的是线性均匀量化器。DCT 输出矩阵通过量化来减小系数的精度, 从而减少了数据比特率。我们采用 JPEG 常用的针对亮度分量的量化表 Q 对 DCT 输出矩阵进行量化。

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

图 4.6 JPEG 亮度分量的量化表

量化与反量化过程为:

$$B_m^Q(u, v) = \text{IntergerRand} \left(\frac{F(u, v)}{Q(u, v)} \right), \quad B_m^Q(u, v) = B_m^Q(u, v) \times Q(u, v)$$

$$0 \leq u, v \leq 7, m = 0, 1, 2 \dots 63 \quad (4.30)$$

量化表 Q 含有量化值, 它们也被称为步长。如上式所示, 其中 $Q(u, v)$ 是量化值, 做分母, 如果它是 1, 量化后的系数就有较高的精度。随着量化值增大, 量化后的系数的精度就降低。 Q 的右下半部分具有较大的值, 这就导致在量化矩阵的右下半部分有许多接近零值的量化系数, 也因此降低了高频分量的精度。

(5) **取出主要量化系数:** 按照从上到下, 从左到右的顺序取出每个块主要的量化系数。提取的方式是在 B_m^Q 中, 按照 Z 扫描方式 (关于 Z 扫描可参阅文献[18]), 取出最前面的 16 个系数, 即为 P_m 。然后做:

$$P = \sum_{m=1}^{63} P_m = \{d_1, d_2, \dots, d_n, n = 1024\} \quad (4.31)$$

上式的含义是把提取出的系数排成一行, 一共有 1024 个。

(6) 将 P 做行堆叠, 形成 32×32 的系数矩阵 P_s , 然后将 P_s 分成互不重叠的大小为 16×16 的块, 每块记为 $WRD_i, i = 1, 2, 3, 4$ 。对于 WRD_i 矩阵来说, 其中的每个系数都能分解为 8

位的两进制码。每个系数的二进制码的第 j 位组合在一起构成二进制位平面，可见， WRD_i 能分解成 8 个二进制的矩阵，它们记为 $WRB_{i,j}$ ， $(1 \leq i \leq 4, 1 \leq j \leq 8)$ 。如图 4.7 所示：

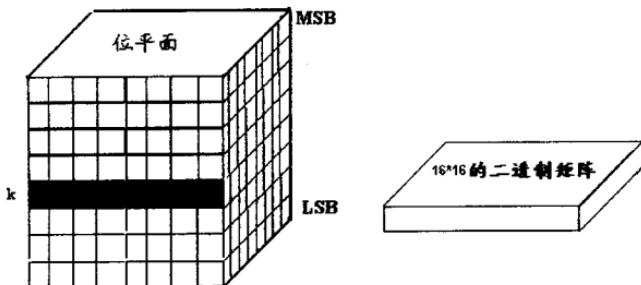


图 4.7 分离的位平面示意图

综上所述，一幅 64×64 的灰度图像经过六个步骤变成了 32 个大小为 16×16 的二进制矩阵。对另外两个基色图像 WG 和 WB 也做同样的操作，结果就得到：

$$\begin{aligned} WR &\rightarrow \{WRB_{i,j}, 1 \leq i \leq 4, 1 \leq j \leq 8\}; \\ WG &\rightarrow \{WGB_{i,j}, 1 \leq i \leq 4, 1 \leq j \leq 8\}; \\ WB &\rightarrow \{WBB_{i,j}, 1 \leq i \leq 4, 1 \leq j \leq 8\}. \end{aligned} \quad (4.32)$$

为了保证水印提取的可靠性，还可以对上面的二进制矩阵做差错编码的处理。方法是，将 $WRB_{i,j}$ ， $WGB_{i,j}$ ， $WBB_{i,j}$ 每组二进制矩阵对应位置上的二进制数当作一组信息码，进行差错编码。编码方式可以是线性的分组码，也可以是非线性的卷积码等，这里不再详细讨论。水印嵌入前对之做差错编码，可以提高水印的抗误码能力（参看文献[12]），但要注意的是这样做无疑会增加嵌入的信息量，因此还应该谨慎为之。

4.2.2 置乱技术

本节介绍的“置乱技术”实际上是一种图像加密技术，上文的 3.1 节已经提到，不少水印算法将传统的加密技术引入其中，目的是进一步增强算法的安全性。也就是说，即使攻击者窃取了整个水印方案，从数据中取出了水印，但如果水印信息在嵌入前进行了加密，他依然无法知道水印的确切信息。这对于含有秘密信息的水印来说是十分必要的，但即使水印信息是公开的，加密技术的采用也能进一步提高它的安全性和可靠性。当然，不单可以直接对水印信息采取加密，也可以在水印嵌入过程中采用加密的技术，如可以对水印的嵌入位置、嵌入强度等作加密。为了全面讨论与水印的相关技术，在本节首先介绍对水印信息进行加密的“置乱技术”。下一节将介绍如何对水印的嵌入位置做加密。

置乱技术是随着信息的安全与保密被重视而发展起来的图像加密技术。该技术早期是对模拟图像的位置空间做置换，它可以看作是从经典密码学中的单表系统扩展而来的。对于数字化的图像，置乱过程不仅可以在数字图像的空间域上进行，还可以在其频率域上进行。数字图像置乱即是一种加密方法，算法设计者可以自由控制算法的选择，参数的选择以及使用随机数技术，从而达到非法攻击者无法破译图像内容的目的。

置乱技术的方法主要有以下几种：*Arnold* 变换^[52]、幻方、*Hilbert* 曲线、*Conway* 游戏、*Tangram* 算法和 *IFS* 模型等。在水印算法中使用置乱技术应考虑两个要求：(1) 尽可能小的计算量；(2) 尽可能大的置乱度。所以本文采用 *Arnold* 变换对上节产生的 32×3 个二进制矩阵作置乱。

Arnold 变换是 *A. I. Arnold* 在研究环面上自同态时提出来的。它的具体方法是，设有图像上的像素点坐标 (x, y) ，将坐标 (x, y) 变到另一个坐标 (x', y') 的变换为：

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \pmod{N} \quad (4.33)$$

其中 $x, y \in (0, 1, 2, \dots, N-1)$ ，表示某一像素点的坐标，而 N 是图像矩阵的阶数。

经过 *Arnold* 变换后的图像会变得“面目全非”，但继续使用 *Arnold* 变换，一定会出现一幅与原图相同的图像。也就是说，*Arnold* 变换具有周期性。对于不同的 N ，*Arnold* 变换有不同的周期，如表 1 所示。

N	2	4	6	8	10
周期	3	3	12	6	30
N	12	16	24	32	48
周期	12	12	12	24	12

表 1 不同阶数 N 下二维 *Arnold* 变换的周期

图 4.8 展示了对一幅二值图像（左图）以二维 *Arnold* 变换进行置乱的实例。图像经过了 13 次 *Arnold* 变换后成为了中间那幅“混乱不堪”的图像，然后再经过了 11 次 *Arnold* 变换后成为与原始图像一样的图像（右图）。



图 4.8 图像置乱的例子

4.3 嵌入水印算法设计

上节详尽地介绍了对水印图像的操作，把一幅 24 bit，大小为 64×64 的真彩色水印图

像变成了 32×3 个 16×16 的二进制矩阵。根据 4.2.2 节的方法还能对它们做置乱，以进一步加强水印的安全性。本节将论述如何把这些二进制信息嵌入到视频中去。

我们采用的彩色视频图像序列也由 RGB 三个色彩频道表示，每帧图像大小为 352×240 。嵌入水印的过程如下：

第一步，将视频图像序列进行场景 (scene) 的分割，整个水印嵌入过程都是针对同一场景的一段图像序列进行的。场景分割的方法很多，我们在这里不作详细的讨论。只是假设同一场景中的 32 帧图像组成一个图像组 (GOP)，记作： V ，其中的每帧图像记作 $F_i, 1 \leq i \leq 32$ 。

第二步，对 F_i 做颜色空间的转换，从 RGB 颜色空间转变为 YCbCr (亮度色度色度) 空间，转换的公式为：

$$\begin{aligned} Y &= 0.299000 \times R + 0.587000 \times G + 0.114000 \times B \\ Cb &= -0.168736 \times R - 0.331264 \times G + 0.500000 \times B \\ Cr &= 0.500000 \times R - 0.418688 \times G - 0.081312 \times B \end{aligned} \quad (4.34)$$

嵌入了水印之后，要将 YCbCr 表示的视频图像序列转换回 RGB 表示，所以这里将这组转换的公式也列出：

$$\begin{aligned} R &= Y + 1.402000 \times Cr \\ G &= Y - 0.344136 \times Cb - 0.714136 \times Cr \\ B &= Y + 1.772000 \times Cb \end{aligned} \quad (4.35)$$

我们将水印信息嵌入视频图像序列的亮度图像中，即 Y 频道的图像中。原因是，针对人眼的视觉特性来说，亮度分量往往比两个色度分量更加重要，因此大部分有损图像压缩方案就会对色度分量采取精度较低的量化，这样会导致嵌入其中的水印遭到更严重的破坏。出于这个原因，我们选择了亮度图像嵌入水印。视频的每帧亮度图像记作： $FY_i, 1 \leq i \leq 32$ 。

第三步：将每帧图像 FY_i 分成互不重叠的，大小为 64×64 的块。帧图像的大小是 352×240 ，这样共可分成 15 块。32 帧图像对应位置的块就构成了一个三维立体的块，立体块的大小为 $64 \times 64 \times 32$ ，每块记作 $B_k, 1 \leq k \leq 15$ ，如下图所示。

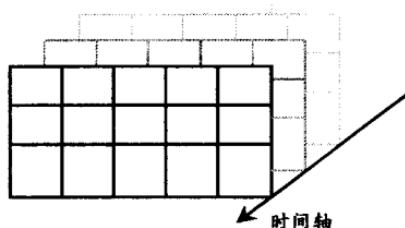


图 4.9 图像组分块

第四步，以人类视觉系统（Human Visual System, HVS）的特性为依据，对上一步产生的立体块 $B_k, 1 \leq k \leq 15$ 进行排序。HVS 的两个重要特性在这里得到应用，它们是：

- 人眼在物体发生运动时，空间敏感度有所下降，因此在视频的运动区域所做的改动更不容易被察觉；
- 根据 HVS 的对比度特性，图像的背景越亮，纹理越复杂（或有边缘存在），视觉系统的敏感度就越差，这个特性也被称为“照度掩蔽性”和“纹理掩蔽性”。

所以，我们可以通过计算各个块的运动参数、照度参数和均匀度（纹理）参数，并把它们加以综合，得到“水印容忍指数”，根据这个指数来对各块排序，以及调整水印嵌入方案，实现自适应。具体的方法是，

首先计算 B_k 的运动参数，沿时间轴相邻两帧对应像素的变化率为：

$$V_t^k(m_j, n_j) = \frac{f_t^k(m_{j+1}, n_{j+1}) - f_t^k(m_j, n_j)}{t_{j+1} - t_j} \quad (4.36)$$

其中， $1 \leq m_j, n_j \leq 64$ ， $1 \leq j < 32$ ， $1 \leq k \leq 15$ 。

对每个位置 (m, n) 的像素值，找出最大的 $V_t^k(m_j, n_j)$ ，即：

$$V_{\max}^k(m, n) = \max_j \{V_t^k(m_j, n_j)\} \quad 1 \leq j < 32 \quad (4.37)$$

将 B_k 所有的 $V_{\max}^k(m, n)$ 求和并取平均，得到：

$$V_{sum_max}^k = \frac{1}{64 \times 64} \sum_{m=1}^{64} \sum_{n=1}^{64} V_{\max}^k(m, n) \quad 1 \leq k \leq 15 \quad (4.38)$$

$V_{sum_max}^k$ 作为运动参数，可以用来测度立体块 B_k 的运动属性。

然后，我们要计算照度参数和均匀度参数。它们在每帧图像中单独计算，然后取平均值。

B_k 是由 32 幅 64×64 的子图像组成的，每幅子图像的照度参数就是它的均值，有

$$L_i^k = \frac{1}{64 \times 64} \sum_{m=1}^{64} \sum_{n=1}^{64} f_i^k(m, n), \quad 1 \leq i \leq 32 \quad (4.39)$$

将 32 个 L_i^k 取平均值，得到 L^k ，它就是照度参数。

根据 Weber 定律，设背景照度为 I ，在均匀背景下，人眼刚好可以识别的物体照度为 $I + \Delta I$ ， ΔI 满足

$$\Delta I \approx 0.02 \times I \quad (4.40)$$

对视觉系统的进一步研究表明 ΔI 与 I 的关系更接近指数关系；参考文献[53]提出了更准

确的对比度敏感函数 (CSF, *Contrast Sensibility Function*)

$$\Delta I = I_0 \cdot \max\{1, (I/I_0)^\alpha\} \quad (4.41)$$

其中, I_0 为当 $I=0$ 时的对比度门限, α 为常数, 一般 $\alpha=0.6-0.7$ 。

对块 B_k 的每幅子图, 根据上式可定义如下衡量均匀度 (纹理) 的参数:

$$D_i^k = \frac{1}{64 \times 64} \sum_{m=1}^{64} \sum_{n=1}^{64} w(L_i^k) \cdot \frac{|f_i^k(m, n) - L_i^k|}{L_i^k} \quad (4.42)$$

加权系数 $w(L_i^k)$ 作为修正因子, 可由下式计算:

$$w(L_i^k) = (1/L_i^k)^\alpha \quad (4.43)$$

将 32 个 D_i^k 取平均值, 得到 D^k , 它就是均匀度参数。用来衡量图像的纹理复杂度。

计算出了三个参数: $V_{sum_max}^k$, L^k 和 D^k , 我们把它们综合起来就能得到所谓的“水印

容忍度”参数 T^k 了, 如下式所示。

$$T^k = p_c \cdot V_{sum_max}^k + p_b \cdot (L^k + D^k), \quad 1 \leq k \leq 15 \quad (4.44)$$

式中的加权系数 p_c , p_b 可以根据视频的情况来指定, 只要满足 $p_c + p_b = 1$ 。对于运动部分为主的视频可以取 $p_c > p_b$, 而对于静止部分为主的视频可以取 $p_c < p_b$ 。最后, 按照 T^k 从大到小将 B_k 做排序。记录下排序的结果(*Blocks_Order*), 以备水印提取时使用。

第五步, 排在前面的四个立体块 $\{B_1, B_2, B_3, B_4\}$ 用来嵌入 $WGB_{i,j}, (1 \leq i \leq 4, 1 \leq j \leq 8)$ 。

方法是: 先对 B_1 做两次三维小波变换 (详见 4.1.4), 我们选择 $LHL2$ 和 $HLL2$ (如图 4.4) 这两个三维子带来嵌入 $WGB_{1,j}, 1 \leq j \leq 8$ 等 8 个二进制矩阵。 $LHL2$ 与 $HLL2$ 的大小都是 $16 \times 16 \times 8$, 每帧小波系数矩阵用 $LHL2^l$ 及 $HLL2^l$, $1 \leq l \leq 8$ 表示, 它们的大小与水印矩阵相同。

我们将 $WGB_{1,1}$ 中的二进制信息嵌到 $LHL2^l$ 或 $HLL2^l$ 对应位置的小波系数中。至于到底嵌入到哪里, 由一个随机的二进制矩阵 P^l 决定, P^l 的大小也为 16×16 , 如果 $P(u, v)=1$, 则对应位置的 $WGB_{1,1}(u, v)$ 嵌入 $LHL2^l(u, v)$ 中; 如果 $P(u, v)=0$, 则对应位置的 $WGB_{1,1}(u, v)$ 嵌入

¹随机二进制矩阵 P^l 的产生方法是, 采用反馈移位寄存器产生一维的 m 序列, 我们知道 m 序列的特点是具有周期性和均衡性, 而且它仅由移位寄存器的结构和初值决定, 所以在接收端提取水印时, 只需要这两个参数就能够产生与嵌入时相同的 m 序列。产生了长度为 256 的一维 m 序列后, 再经过行堆叠将其变成 16×16 的二维矩阵。 m 序列的产生方法见参考文献 [35]。

$HLL2^1(u, v)$ 中。

确定了嵌入位置后，我们按如下的方法嵌入水印：

$$\begin{aligned}
 &\text{当 } w(u, v) = 1 \text{ 且 } c(u, v) \geq 0 \text{ 时, } c'(u, v) = c(u, v) - c(u, v) \bmod S + T_1; \\
 &\text{当 } w(u, v) = 1 \text{ 且 } c(u, v) < 0 \text{ 时, } c'(u, v) = c(u, v) + |c(u, v)| \bmod S - T_1; \\
 &\text{当 } w(u, v) = 0 \text{ 且 } c(u, v) \geq 0 \text{ 时, } c'(u, v) = c(u, v) - c(u, v) \bmod S + T_2; \\
 &\text{当 } w(u, v) = 0 \text{ 且 } c(u, v) < 0 \text{ 时, } c'(u, v) = c(u, v) + |c(u, v)| \bmod S - T_2.
 \end{aligned} \tag{4.45}$$

上式中 $w(u, v)$ 为水印信息，此时也就是 $WGB_{1,1}(u, v)$ 。 $c(u, v)$ 为小波变换系数，它可能是 $LHL2^1(u, v)$ ，也可能是 $HLL2^1(u, v)$ ，当然由 P 决定。 $c'(u, v)$ 为经过修改的小波系数，也就是嵌入了水印的小波系数。而 S 、 T_1 和 T_2 为由算法指定的参数。 S 是非负的整数，我们要求 $T_1 = S/4$ ， $T_2 = (S/4) \times 3$ 。

S 、 T_1 、 T_2 三者的取值与嵌入水印后视频图像的质量有很大关系，试验表明， S 越大， T_1 与 T_2 的距离越远，而嵌入水印后的图像失真就越厉害。然而，提取水印的错误率也越低。所以，我们采用第四步选择出来的 $\{B_1, B_2, B_3, B_4\}$ 来嵌入最重要的 $WGB_{i,j}$ （可参看 4.2.1），HVS 对这个区域的变化相对来说最不敏感（可参看上文的“第四步”）；因此我们可以将 S 取数取得大一些，这样鲁棒性更强，保证绿色基色图恢复的质量。而在其它区域嵌入红色基色图与蓝色基色图时，可将 S 取得小一些，保证了不可见性的需要。

按照上述的方法，把另外七个矩阵 $\{WGB_{1,j}(u, v), 2 \leq j \leq 8\}$ 按次序嵌入到 B_1 的 $\{(LHL2^l, HLL2^l), 2 \leq l \leq 8\}$ 之中。

对另三个立体块 B_2 、 B_3 、 B_4 来说，做法与 B_1 相同。总之，将 $WGB_{i,j}, (1 \leq i \leq 4, 1 \leq j \leq 8)$ 全部嵌入为止。

第六步：将嵌入了水印的块做三维小波反变换。做反变换时要注意正变换的顺序，假设正变换的顺序是，第一做时间维上的 DWT，第二做行方向的 DWT，第三做列方向的 DWT，那么反变换时，要采用相反的顺序。

第七步：按照第五步和第六步的方法，将 $WRB_{i,j}, (1 \leq i \leq 4, 1 \leq j \leq 8)$ 嵌入 $\{B_5, B_6, B_7, B_8\}$ 中，将 $WBB_{i,j}, (1 \leq i \leq 4, 1 \leq j \leq 8)$ 嵌入 $\{B_9, B_{10}, B_{11}, B_{12}\}$ 中。只是必须注意参数 S 、 T_1 和 T_2 取值的不同， S 应该取得越来越小才对。

第八步：将视频序列的每帧图像进行颜色空间的转换，从 $YCbCr$ 空间转换到 RGB 空间，

转换公式见 4.35 式。

至此，水印嵌入过程结束。

4.4 水印提取算法设计

本算法的一大优势是，水印提取时无需原始视频数据的参与。但依然需要一些参数的帮助。它们是：

1. 嵌入过程第四步产生的块 B_k 的排序结果，由此我们可以得知水印的大致嵌入位置；
2. 嵌入过程第五步产生 m 序列的移位寄存器的结构（可用一个 8 进制数表示）及初值；由此我们可以确定水印嵌入的确切位置；
3. 嵌入时指定的参数 S ，对应不同的水印基色图有三个不同的 S ；
4. 如果嵌入时采用了 4.2.2 节的置乱技术，那么还需要 Arnold 变换的次数和周期。

获得了上面的各项参数，水印的提取就十分简单了。步骤如下：

第一步：找到视频中嵌入了水印的那个图像组 (GOP)，其中的每幅帧图像记作 $F_i, 1 \leq i \leq 32$ 。

第二步：对 F_i 做颜色空间的转换，从 RGB 空间转变到 YCbCr 空间，转换公式见式 4.34。

第三步：将由 Y 频道的帧图像组成的图像序列分割成互不重叠的、大小为 $64 \times 64 \times 32$ 的三维立体块，记作 $B_k, 1 \leq k \leq 15$ 。

第四步：根据先前记录的 *Blocks_Order*，首先对排在前面的 4 个立体块提取嵌入其中的水印图像的绿色基色图。

对 $\{B_1, B_2, B_3, B_4\}$ 做三维小波变换，使用的小波变换要与嵌入过程的一模一样。然后，根据记录下的移位寄存器的结构和初始值产生与嵌入过程相同的 m 序列，对之做行堆叠，生成二维的二进制随机矩阵 P 。由 P 我们得到了水印信息在小波子带中确切的嵌入位置。

第五步：每个被确定嵌入了水印的小波系数 $c''(u, v)$ ，作如下的判断：

$$\begin{aligned} \text{如果 } |c''(u, v)| \bmod S < \frac{T1 + T2}{2}, \text{ 那么 } wd(u, v) = 1; \\ \text{如果 } |c''(u, v)| \bmod S \geq \frac{T1 + T2}{2}, \text{ 那么 } wd(u, v) = 0 \end{aligned} \quad (4.46)$$

式中, T_1 和 T_2 可以为我们记录的系数 S 所求出, 即 $T_1 = S/4, T_2 = (S/4) \times 3$ 。这里要注意对不同的基色图像来说, S 取的值是不同的。 $wd(u, v)$ 为提取出的水印信息。

如果在嵌入过程中使用了置乱技术, 那么还需对每个立体块中提取出来的 16×16 的矩阵作“反置乱”操作。方法是根据记录下的置乱时 Arnold 变换的次数 r 和周期 T , 再继续做 $T - r$ 次 Arnold 变换即可。

按照上式, 将所有 32 个矩阵从 4 个立体块中提取出来, 并按原先的位置放好(参看 4.2.1 节), 组成了 8 个 32×32 的二进制矩阵。将对应位置上的 8 个二进制数“合成”一个十进制数。于是我们就得到了重构绿色基色图所需的 DCT 系数。

第六步: 进行绿色基色图的重构, 由于我们压缩水印图像时使用的是分块的 DCT 压缩。因此, 我们首先要把这些 DCT 系数安排到每个图像块合适的位置, 所谓合适的位置由乙扫描决定(参看 4.2.1 节)。然后对每个图像块做逆 DCT 变换, 可见式 3.5。最后我们得到了绿色基色图的重构图像。

第七步: 按照第四到六步的做法, 分别在 $\{B_5, B_6, B_7, B_8\}$ 和 $\{B_9, B_{10}, B_{11}, B_{12}\}$ 中提取出红色基色图与蓝色基色图的 DCT 系数, 并重构两幅基色图, 然后将三幅基色图像合成一幅彩色的水印图像。

至此, 水印图像的提取过程结束。

本章是论文的核心。在第三章论述的数字水印技术的基本框架和现有的研究方法的基础上, 作者在本章提出了一种新的视频水印嵌入和提取算法。它是一种基于视频局部三维小波变换的盲水印方案, 并结合了人类视觉系统的特性, 在视频图像的不同区域采取不同的嵌入策略, 实现了算法的自适应, 以及水印不可见性和鲁棒性等要求。下一章将用实验来检验该算法的性能。

第五章 实验结果与分析

本文的水印算法可用于视频的版权保护,因此衡量算法优劣的指标主要是不可见性和鲁棒性。如第三章所言,视频水印算法(也包括静态图像的水印算法)的不可见性要求视频或者图像在嵌入水印后在视觉上无法感知水印的存在。实验发现,水印嵌入后,完全做到与原始视频无丝毫差异是很困难的,但是可以做到在视觉上无明显的差异,图像的质量得到保证,因而并不影响其使用价值和商业价值。

水印算法的鲁棒性是指嵌入视频中的水印信息不易遭到破坏,对常见的信号处理以及有意的攻击具备“免疫”的能力。但说水印算法鲁棒性的好坏,只能是一个相对的评价。任何一个水印算法都无法做到对所有有意或无意的破坏和攻击全部具备抵抗能力。本章所做的是在几个基本的方面对水印算法进行检验,如MPEG压缩,对帧图像叠加噪声,丢失若干帧,颠倒帧序和线性滤波等。

本文算法的一个显著特点是有效地利用了人类视觉系统(HVS),根据视频图像的内容来自适应地选择水印信息的嵌入位置和相应的嵌入策略。为什么要考虑HVS?原因是不可见性和鲁棒性实际上是两个相互矛盾的指标。不可见性要求水印嵌入的强度越小越好,对图像的改动也越小越好。因为水印一旦被发现,那么它就更容易被破坏和删除。但是用于版权保护的水印又不能是脆弱的,它还要有一定的强度来抵抗攻击。所以一些水印算法利用人类视觉系统的一些特性来指导水印的嵌入过程,对人眼感知比较迟钝的地方嵌入强度较大的水印,而在人眼感知较为敏感的地方嵌入强度较小的水印。本文的算法就具备这样的特点。我们在这一章给出了自适应算法与非自适应算法(也就是,不考虑HVS,对图像序列采取统一的嵌入策略,详见下文)的比较,从而有说服力地凸现了本文算法的优越性。

5.1 实验说明

我们实验所采用的水印图像是大小为 64×64 的24 bit真彩色图像。如图5.1所示:



图 5.1 水印图像 (Olympics)

实验用的是三组视频图像序列,它们是:

- (1) *Calendar* 图像序列,共32帧;

- (2) *Bicyclers* 图像序列, 共 64 帧;
- (3) *Flowers Garden* 图像序列, 共 96 帧。

以上视频每帧图像的大小都是 352×240 , 且都是真彩色图像。为什么要选择这三组视频呢? 因为它们都是 MPEG 标准的测试图像序列, 多为研究者所采用, 而且它们蕴含的内容也各有特色。第一组 *Calendar* 图像序列运动区域和静止区域分布比较均匀, 让人最感兴趣的是那列玩具火车和滚动的皮球, 挂在墙上的挂历也在作自上而下的运动, 而背景几乎是静止的; 第二组是 *Bicyclers* 图像序列, 它描绘了自行车追逐赛冲刺的镜头, 蕴含了丰富的运动场景; 第三组的 *Flowers Garden* 图像序列几乎是静止的, 只有远处的风车在风中缓慢地转动, 另外还有树枝微微地颤动, 都不易为人察觉。由此可见, 这三组视频给我们实验提供了典型的场景内容, 利用它们做实验可以说具有一定的代表性。

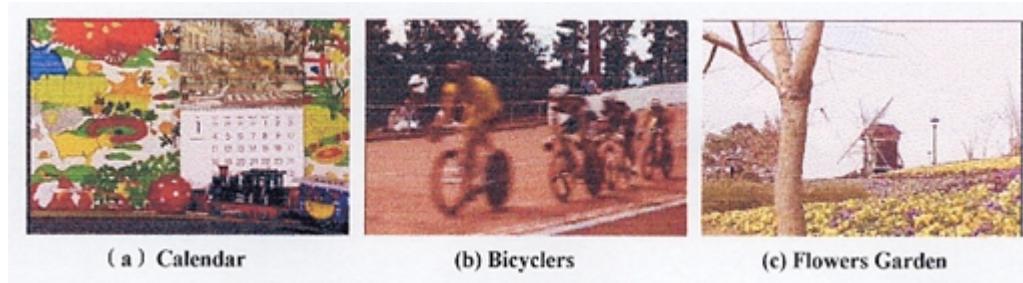


图 5.2 实验用的视频图像组

在实验中, 我们将水印图像周期性地嵌入到视频中去, 每 32 帧图像为一个周期, 那么 *Calendar* 图像序列中嵌入了一幅 *Olympics*, *Bicyclers* 图像序列中嵌入了两幅 *Olympics*, *Flowers Garden* 图像序列中嵌入了三幅 *Olympics*。

本文的水印算法在嵌入过程中需要约定一些参数, 它们是 4.44 式中的 p_c 和 p_b , 4.45 式中的 S 、 T_1 、 T_2 。根据视频内容的不同, 我们经过反复实验选定了这些参数的值。如表 5.1 所示。

视频序列	选定的参数				
	p_c	p_b	S	T_1	T_2
<i>Calendar</i>	0.5	0.5			
<i>Bicyclers</i>	0.8	0.2			
<i>Flowers Garden</i>	0.25	0.75			
			嵌入水印图像的绿色基色图时, $S = 48$, $T_1 = 12$, $T_2 = 36$;		
			嵌入红色基色图时, $S = 36$, $T_1 = 9$, $T_2 = 27$, 嵌入蓝色基色图时, $S = 24$, $T_1 = 6$, $T_2 = 18$ 。		

表 5.1 实验的参数值列表

5.2 水印算法不可见性的检验与比较

使用本文的水印算法将水印图像嵌入视频中，从主观上判断，图像质量几乎没有下降。如图 5.3 所示：



图 5.3 嵌入水印前后图像质量的比较

除了通过主观来判断水印的不可见性之外，我们还引入了客观评价的标准，即嵌入了水印的视频图像的峰值信噪比（PSNR）。峰值信噪比的计算公式可参阅式 3.1。

表 5.1, 5.2, 5.3 分别列出了三组视频嵌入水印后，计算得出的峰值信噪比（PSNR）。

之后，我们将本文的自适应算法与非自适应算法作了比较。这里的非自适应算法的做法是在水印嵌入过程中，省去根据图像的内容选择嵌入区域的步骤，而是随机地选择嵌入水印的立体块。另外，水印图像还是分为三幅基色图，但每幅基色图嵌入时的参数 S 、 T_1 、 T_2 都是一样的。也就是，嵌入水印的强度是不变的。

图 5.4 给出了几种算法的结果，即嵌入了水印的视频帧图像的峰值信噪比。以及对它们的分析。

图像帧号	峰值信噪比 (单位: DB)						
1	31.81	9	32.25	17	32.03	25	32.05
2	32.37	10	31.51	18	31.22	26	32.52
3	31.90	11	32.53	19	32.07	27	32.27
4	32.54	12	31.89	20	32.35	28	32.60
5	31.72	13	32.20	21	32.23	29	32.19
6	32.11	14	32.72	22	32.88	30	32.10
7	32.05	15	31.34	23	32.74	31	31.82
8	31.98	16	32.05	24	31.87	32	31.28

表 5.2 Calendar 视频的峰值信噪比

平均信噪比为 32.09 DB

☆说明: 表 5.2 的帧号是相应图像组的帧号, 由于 Bicyclers 视频有 64 帧, 被分为了两组, 所以每个帧号对应有两个峰值信噪比, 分别对应于第 1 组与第 2 组的帧图像。

表 5.3 的情况同表 5.2, 因为 Flowers 视频有 96 帧图像组成, 它被分为三组, 所以每个帧号对应了三个峰值信噪比。

图像帧号	峰值信噪比 (单位: DB)										
1	32.18	32.26	9	32.69	31.73	17	31.09	32.32	25	32.76	31.02
2	32.17	31.02	10	31.08	32.35	18	30.97	31.18	26	31.58	32.04
3	31.29	31.29	11	32.54	30.75	19	31.08	31.21	27	32.07	32.97
4	31.90	32.43	12	31.78	31.86	20	31.96	32.62	28	32.70	31.22
5	32.22	32.79	13	30.65	32.92	21	32.31	32.55	29	31.17	32.21
6	31.92	31.81	14	32.62	31.96	22	30.33	31.60	30	32.43	31.16
7	32.95	32.18	15	32.05	31.91	23	31.70	32.46	31	31.25	31.18
8	32.90	31.97	16	31.84	31.84	24	32.59	31.42	32	31.31	32.04

表 5.3 Bicyclers 视频的峰值信噪比
第一组平均信噪比为 31.84 DB, 第二组平均信噪比为 31.97 DB

图像 帧号	峰值信噪比 (单位: DB)			图像 帧号	峰值信噪比 (单位: DB)			图像 帧号	峰值信噪比 (单位: DB)		
1	31.31	32.20	32.53	9	32.41	32.61	31.79	17	31.88	31.80	32.21
2	32.48	31.99	32.89	10	32.43	32.88	32.33	18	32.10	31.71	32.50
3	32.45	32.13	31.99	11	32.49	30.98	32.07	19	31.75	32.44	32.52
4	31.99	31.72	32.77	12	32.86	32.25	31.95	20	32.34	32.45	32.85
5	32.18	32.06	32.41	13	32.01	32.06	31.99	21	32.07	32.50	31.09
6	32.27	32.74	32.58	14	32.73	31.94	31.67	22	31.67	32.87	32.34
7	31.94	32.37	31.09	15	31.20	31.88	32.03	23	32.48	31.92	32.07
8	31.81	32.54	32.13	16	31.90	32.15	32.72	24	32.82	32.71	31.82

图像 帧号	峰值信噪比 (单位: DB)			图像 帧号	峰值信噪比 (单位: DB)		
25	31.96	31.14	31.95	29	31.16	31.79	32.09
26	32.72	31.91	31.96	30	32.13	32.28	32.44
27	32.35	31.83	31.75	31	31.75	32.07	32.47
28	32.58	31.98	31.68	32	32.26	31.70	32.79

表 5.4 Flowers 视频的峰值信噪比
第一组平均信噪比为 32.14 DB,
第二组平均信噪比为 32.11 DB,
第三组平均信噪比为 32.17 DB.

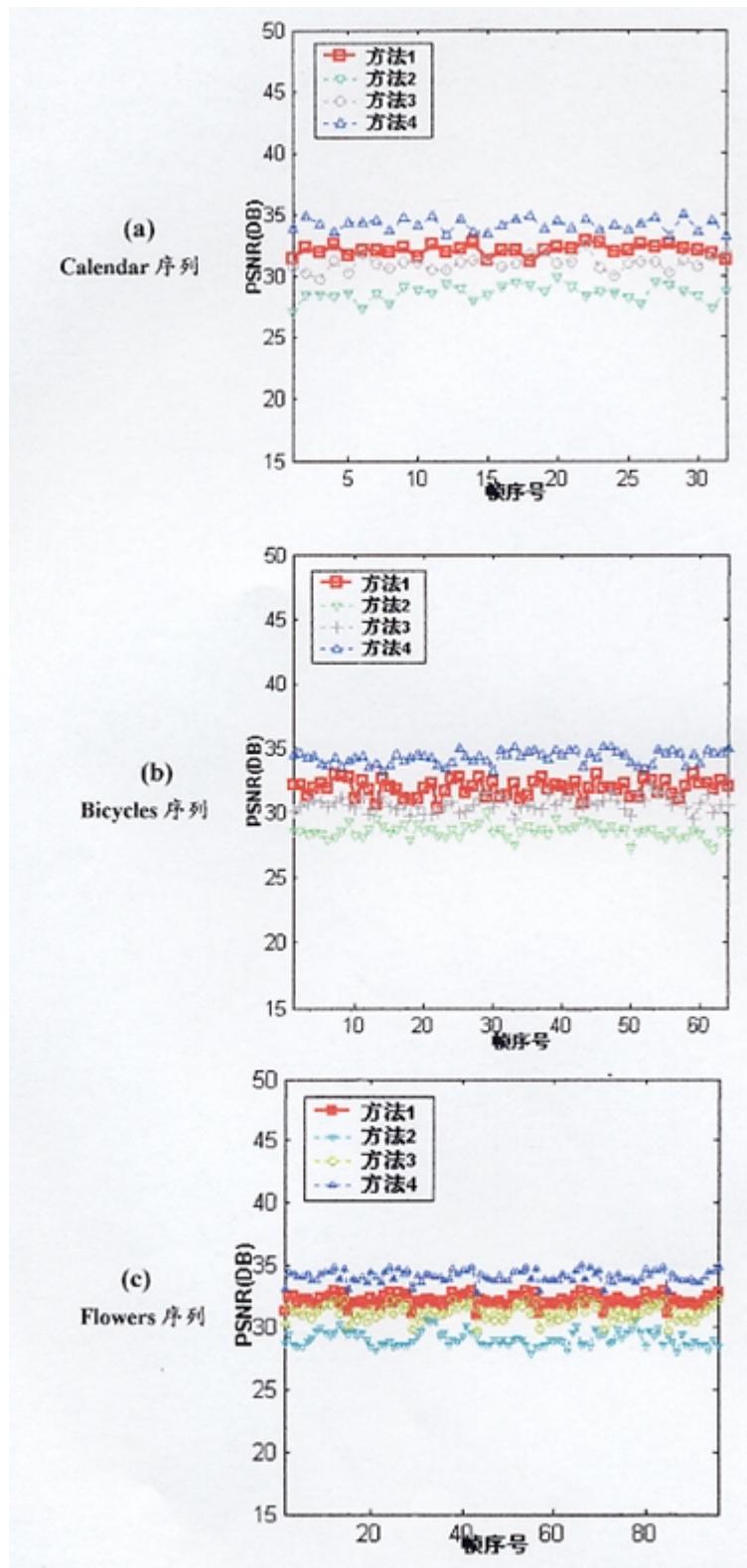


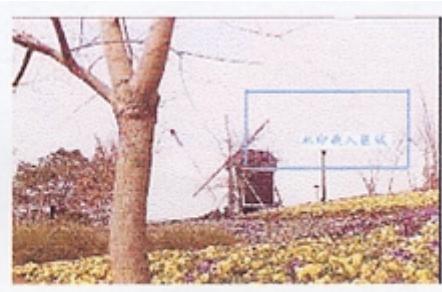
图 5.4 自适应算法与非自适应算法的比较

☆说明：上图中“方法 1”指的是自适应算法的结果；“方法 2、3、4”为非自适应算法的结果。注意，方法 2 中 $S=48$ ，方法 3 中 $S=36$ ，方法 4 中 $S=24$ 。

上图表明, 从峰值信噪比的角度来看, 方法 1 的信噪比并非最优, 它明显劣于方法 4, 但是方法 4 的 T_1 、 T_2 取得太小, 水印的鲁棒性不佳, 无法达到要求。在下文中, 可以看到。而方法 2 中, T_1 、 T_2 取得太大, 水印的鲁棒性是最好的, 因为嵌入的水印强度最大, 但是其不可见性无法满足。水印很容易被发现, 所以用牺牲不可见性来换取鲁棒性的提高, 使被保护视频产品的质量受到严重影响, 这样做不可取。方法 3 中的 T_1 、 T_2 取的较为适中, 它的峰值信噪比只是略低于方法 1, 看来可以取代复杂性较高的方法 1, 但从主观的角度判断, 方法 1 要好于方法 3。原因是, 方法 3 对视频的所有区域“一视同仁”, 但有些如静止的背景, 且亮度适中的区域, 人眼的感觉是最敏锐的。在这些区域嵌入强度如此的水印就显得太冒险了。如下图, 我们可以凭借肉眼感觉图像有明显的失真, 从而判断出水印的存在。



(a) 方法 1



(b) 方法 3

图 5.5 方法 1 与方法 3 水印嵌入后结果的比较

5.3 提取出的水印图像评价

对于本算法提取出的水印图像, 我们也给出了一种衡量其质量的标准, 就是相似度 Sim , 它的计算公式如下:

$$Sim(W, W^*) = \frac{\sum \sum W(i, j)W^*(i, j)}{\sum \sum [W(i, j)]^2} \quad (5.1)$$

W 为原始水印图像, W^* 为提取出来的水印图像。 $Sim(W, W^*)$ 越接近 1 说明 W 与 W^* 越相似。

在度量提取出来的水印图像与原始水印图像的相似性时, 我们考虑到实际嵌入视频的其实是水印图像的重要的 DCT 参数, 而一部分 DCT 参数已经在嵌入之前删除了(参看 4.2 节), 这种处理本身将引起图像的失真, 它与整个嵌入过程都无关。因此我们分两步考虑相似性, 第一步是原始图像与经过 DCT 压缩后的图像的比较; 第二步是压缩后的图像与它经过了嵌入和提取过程后的图像之间的比较。如图 5.6、5.7、5.8、5.9 所示。



PSNR: 22.45 DB SIM: 0.89

图 5.6 水印压缩后的图像



PSNR: 33.53 DB SIM: 0.98

图 5.7 从 Calendar 视频中提取出来的水印图像



(a)

PSNR: 34.37 DB SIM: 0.97



(b)

PSNR: 33.97 DB SIM: 0.99

图 5.8 (a) 从 Bicyclers 视频的第一组图像中提取出来的水印图像

(b) 从 Bicyclers 视频的第二组图像中提取出来的水印图像



a) PSNR: 33.57 DB SIM: 0.98



(b) PSNR: 33.72 DB SIM: 0.98



(c) PSNR: 34.20 DB SIM: 0.97

图 5.9 (a) 从 Flowers Garden 视频的第一组图像中提取出来的水印图像

(b) 从 Flowers Garden 视频的第二组图像中提取出来的水印图像

(c) 从 Flowers Garden 视频的第三组图像中提取出来的水印图像

5.4 水印算法鲁棒性的检验与比较

检验了视频水印的不可见性和提取出水印图像的相似度后,我们要对水印算法的鲁棒性做进一步的检验。鲁棒性的好坏对于水印算法也十分重要。攻击者对含水印产品进行攻击的主要目的是想破坏它体现版权所有的能力。对含水印图像的常见攻击方法分为无意攻击和有意攻击两大类。我们在此用到的无意攻击主要指一些保持视觉相似性的处理,如 MPEG 压缩、叠加各类噪声以及线性滤波等,有意攻击指删除若干帧图像,颠倒帧序等。

5.4.1 无意攻击的检验与比较

MPEG-1 是较为常用的视频压缩算法，支持的数据率为 1.5 Mbit/s 。嵌入了水印的视频图像序列经过了 MPEG-1 的压缩，能否提取出水印信息？提取出的水印信息质量如何？我们对算法作了检验。



图 5.10 MPEG 压缩后还原的帧图像

嵌入了水印的 *Calendar* 序列经过了 MPEG 压缩后，再解码还原。之后，我们使用本文算法（见 4.4 节）提取出其中的水印图像，如下图所示：

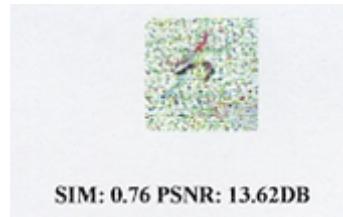


图 5.11 使用本文算法，MPEG 压缩后，提取出来的水印图像

由上图可见，提取出来的水印图像其本来面目可以辨认，这说明本文算法对 MPEG 压缩具有一定的抵御能力。

为了比较非自适应算法之方法 4 的性能，我们也对使用方法 4 嵌入水印的视频做了 MPEG 压缩，然后解码还原，提取出水印，结果如下图：



图 5.12 使用方法 4（非自适应算法），MPEG 压缩后，提取出来的水印图像

方法 4 嵌入水印时，强度取得太小，虽然它的不可见性最佳，但是它抵御有损压缩（MPEG 压缩）的能力明显不如本文的自适应算法。

接下来，我们验证水印算法对加性噪声的鲁棒性。这里使用了两种噪声，即盐和胡椒噪声与高斯白噪声。我们对视频图像序列中的每帧图像都叠加噪声，然后提取水印，结果如

下图所示：



图 5.13 叠加了噪声的视频帧图像以及提取出的水印图像

同样，我们与非自适应的方法 3、方法 4 比较，对它们的帧图像叠加了相同能量的噪声后，提取出来的水印如下图：

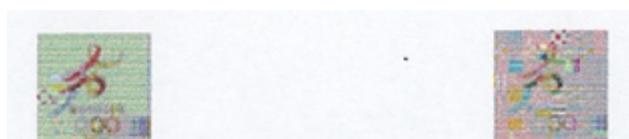


图 5.14 叠加了盐和胡椒噪声后提取出的水印图像



图 5.15 叠加了高斯白噪声后提取出的水印图像

从上面两图可以看出，方法 4 加入了两类噪声后，提取出来的水印图像质量明显下降，

几乎难以分辨，说明它抗噪声的能力很差，这与它嵌入水印的强度太小有直接的关系。方法 3 抗噪声的性能比之有较大的提高，但与图 5.13 相比，水印图像的质量（尤其是主观上）还是差了一些，原因是，方法 1 嵌入水印的绿色基色图时，嵌入的强度要大于红色基色图和蓝色基色图，所以提取出的水印图像绿色分量保留的较为完好，最终主观评价下图像的质量较好。但是方法 3 对三个基色图嵌入的策略是一样的，遭到噪声破坏的程度也大致相同，所以图像质量，尤其从主观上看，不如方法 1。

最后，我们考察了线性滤波对该水印算法的影响，如下面的表 5.5 所示：

线性滤波器	提取出的水印图像
高斯低通滤波器	
模糊对比度	
增强滤波器	

表 5.5 线性滤波后提取出来的水印图像

5.4.2 有意攻击的检验与比较

对视频水印的有意攻击中，典型的是随机地删除视频图像序列中的若干帧图像，试图用这种方法来破坏水印信息。我们模拟了这种攻击，分别对 *Calendar* 序列随机地删除 1 帧，2 帧和 4 帧图像，删除的图像由它的前一帧和后一帧图像相加取平均来填补，提取出的水印图像如表 5.6 所示：

删除了第 21 帧		峰值信噪比 (PSNR): 23.39 dB 相似度 (SIM): 0.81
删除了第 14、23 帧		峰值信噪比 (PSNR): 21.35 dB 相似度 (SIM): 0.76
删除了第 12、17、 22、29 帧		峰值信噪比 (PSNR): 17.61 dB 相似度 (SIM): 0.59

表 5.6 随机删除了帧图像后提取出来的水印及其 PSNR 与 SIM

从上表可以看出，这种攻击方法对本算法有一定的作用。删除了若干帧图像后，我们虽然对视频图像序列做了填补，但从中提取出来的水印图像有了较为明显的失真，而且随着删除图像数量的增加，失真表现得愈发明显。但是不管怎么说，水印的本来面目还是可以辨认的，说明本算法对这种攻击仍有一定的抵抗能力。

另一种攻击方法是更改视频图像序列中图像帧的顺序。如我们交换了 *Calendar* 序列第 23 帧与第 24 帧的次序，结果提取出来的水印图像如下图所示：



图 5.16 更改帧序后提取出的水印图像 (1)

进一步，我们更改了 *Calendar* 序列中连续 6 帧图像的顺序，即将第 22、23、24、25、26、27 帧的顺序改成：25、24、23、22、27、26。结果提取出的水印图像如下：

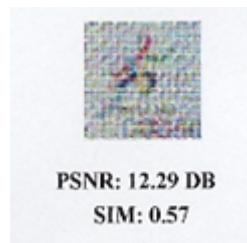


图 5.17 更改帧序后提取出的水印图像 (2)

第六章 结论和今后研究方向

数字水印技术是近几年才发展起来的一项用来保护数字媒体版权的新技术。虽然它已经成为了信息安全和图像处理领域的热点，但很多方面还有待进一步地研究。

目前提出的许多数字水印算法大多针对静止灰度图像，对彩色图像、视频图像的研究较少。而且，使用的水印大多由随机序列组成，对有意义水印（如图像、序列号和文字等）研究得还远远不够。这是由于有意义的水印信息不允许有大的检测错误率，当嵌入数据量较大时，水印编码和检测将存在较大的难度。还有一个弱点在于大多数的算法尚未很好地利用人类视觉系统（HVS）的特性，对视觉特性应用的定性/定量描述尚未见到。所以在保证水印不可见性的问题上存在不小的难度。

本文针对数字水印研究的薄弱环节，所提出的算法具有一定的创新性和实用性。该水印算法的特点在于：（1）将 24 bit 的真彩色图像嵌入同样是彩色的视频图像序列中，所用的水印信息量大，表现力强；（2）算法借鉴了 JPEG 图像压缩的思想，对水印图像进行压缩，因此所嵌入视频中的其实是图像的部分 DCT 系数；（3）算法根据人类视觉系统（HVS）的特性，创建了“水印容忍指数”，按照它来对图像块做排序，并根据排序结果选择水印嵌入位置和相应的嵌入策略。在计算水印容忍指数时，我们既考虑了视频图像序列运动的特点，又考虑了各帧图像的照度和纹理，方法具有一定的代表性；（4）算法使用基于局部立体块的三维小波变换，使算法的运算复杂度较小；（5）在水印提取时，实现了无需原始视频数据的“盲水印”要求，提取水印方法简单，错误率极小，具备实用性；（6）经过实验的仿真和检验，该水印算法的不可见性和鲁棒性均达到一定的水平。

应该说，本文所做的仅是对彩色视频水印的初步研究。今后的研究方向应该是：（1）水印嵌入算法运用了人类视觉系统（HVS）的特性，但还只是给出了初步的定量准则，从实际情况看，对视频图像块的分类还不是十分准确。如何利用 HVS 来更好地找出适宜嵌入水印的区域，如何根据 HVS 来调整水印嵌入强度等问题依然有待深入研究；（2）水印算法抗攻击性的能力还不够，因此水印在解决版权争议方面的效力还待进一步加强。

本篇论文仅仅作为笔者对水印技术研究的起点，今后必将如这句古训所言，“路漫漫其修远兮，吾将上下而求索”。

致 谢

本篇论文的写作历时七个月，可以说是一个不断学习、实践和总结的过程。回首这段经历，我要首先感谢导师张文龙教授。因为，在不同的阶段，我都受益于张老师的悉心指导和帮助。在最初的论文选题期间，张老师给予了我极大的鼓励，使我满怀信心地选择了视频水印这个颇具挑战性的课题，同时，他还提供给我不少学习和参考的资料。在算法的设计和编写期间，张老师的建议和提醒总能帮助我克服一个又一个的困难。在论文的撰写阶段，张老师更是耐心地审阅，使我最终能顺利地完成了毕业论文的工作。

同时，我还要衷心感谢学院里的多位老师，他们的教导和帮助也令我受益匪浅。他们是王耀明老师、尤韦彦老师、李莉老师、董建萍老师、马燕老师和王学刚老师。他们不仅在学习上给予我宝贵的指导，还在生活中为我树立了为人师表的榜样，所以在这里，我要对他们表示最诚挚的谢意。

参考文献

- [1] *Mitchell D. Swanson, Mei Kobayashi, Ahmed H. Tewfik, Multimedia Data-Embedding and Watermarking Technologies, Proceedings of the IEEE, Vol. 86, No.6, June 1998*
- [2] 刘挺, 基于离散小波变换 (DWT) 和 HVS 的彩色数字图像混合型水印嵌入技术算法研究, 上海师范大学硕士学位论文
- [3] 易开祥, 石教英, 孙鑫, 数字水印技术研究进展, 中国图象图形学报, *Vol.6(A), No.2, Feb. 2001*
- [4] 张彤, 王育民, 信息隐藏技术及其在信息安全中的应用, 中兴通信研究报告
- [5] 戴元军, 信息隐藏与数字水印技术, IBM 技术开发中心
- [6] 冯晖, 来凤琪等, 《计算机密码学》, 中国铁道出版社, 1999 年
- [7] *Deepak Kundur, Multiresolution Digital Watermarking: Algorithms and Implications for Multimedia Signals, PhD. Thesis*
- [8] *Ingemar J. Cox, Joe Kilian, Tom Leighton, Talal Shamoon, Secure Spread Spectrum Watermarking for Multimedia, NEC Research Institute, Tech. Rep. 95-10*
- [9] 黄继武, 谭铁牛, 图像隐性水印综述, 自动化学报, *Vol.26, No.5, Sep. 2000*
- [10] 纪震, 张基宏, 蒋一峰, 小波域的扩频数字图像水印算法, 电子与信息学报, *Vol.23, No.8 Aug. 2001*
- [11] 王慧琴, 王志雄, 李人厚, 变换域数字水印技术的研究, 计算机科学, *Vol.28, No. 6, 2001*
- [12] 冯德锦, 张妙兰, 李象霖, 基于卷积码的数字水印算法, 计算机工程与应用, 2002, 8
- [13] *Raymond B. Wolfgang, Christine I. Podilchuk, Edward J. Delp, Perceptual Watermarks for Digital Images and Video, Bell Lab. Tech. Rep.*
- [14] *Ingemar J. Cox, Matt L. Miller, A review of watermarking and the importance of perceptual modeling, Proc. Of Electronic Imaging '97, Feb. 1997*
- [15] 张贤达, 现代信号处理, 清华大学出版社, 1995
- [16] 徐立中, 数字图像的智能信息处理, 国防工业出版社, 2001
- [17] *Deepa Kundur, Dimitrios Hatzinakos, A Robust Digital Image Watermarking Method using Wavelet-Based Fusion, University of Toronto*
- [18] 吴乐南, 数据压缩, 电子工业出版社, 2000

- [19] Puate J., Jordan F., *Using fractal compression scheme to embed a digital signature into an image*, *Proceedings of SPIE Photonics East '96 Symposium, Boston, Massachusetts*, 1996
- [20] Yao Zhao, Qiang Ma, Baozong Yuan, *Digital watermark based on LIFS*, *2000 5th International Conference on Signal Processing Proceedings*, 2000
- [21] 李冠华, 赵耀, 基于分形图象编码的数字水印技术的研究, *铁道学报*, Vol. 23, No.4, Aug. 2001
- [22] Min-Jen Tsai, Kuang-Yao Yu, Yi-Zhang Chen, *Joint Wavelet and Spatial Transformation For Digital Watermark*, *IEEE 2000*
- [23] 伯晓晨, 沈林成, 常文森, 一种新的盲图像水印检测算法, *计算机学报*, Vol.24 No.12, Dec. 2001
- [24] 陈明奇, 钮心忻, 杨义先, 数字水印的攻击方法, *电子与信息学报*, Vol.23, No.7, July 2001
- [25] R. Barnett, D. Pearson, *Attack operator for digitally watermarked images*, *IEE Proc. Vis. Image Signal Process*, Vol. 145, No.4 August 1998
- [26] Scott Craver, Nasir Memon, Boon-Lock Yeo, Minerva Yeung, *Can Invisible Watermark Resolve Rightful Ownerships*, *IBM Research Report*, 1996
- [27] Scott Craver, Nasir Memon, Boon-Lock Yeo, *Resolving Rightful Ownerships with Invisible Watermarking Techniques: Limitations, Attacks, and Implications*, *IEEE Journal on Selected Area in Communications*, Vol.16, No.4, May, 1998
- [28] <http://www.cl.cam.ac.uk/users/fapp2>
- [29] Schyndel R. G Van, Tirkel A. Z., Osborne C. F., *A digital watermark*, *Proc. IEEE Int. Conf. On Image Processing, Piscataway*, IEEE Press, 1994,2
- [30] Bruyndoncky O, Quisquater J J, Marq B., *Spatial method for copyright labeling of digital images*, *Proc. IEEE Workshop on Nonlinear Signal Processing, Piscataway*: IEEE Press, 1995
- [31] Nikolaidis N, Pitas I., *Robust image watermarking in the spatial domain*, *Signal Processing*, 1998,66 (3)
- [32] Kenneth. R. Castleman, *数字图像处理*, 电子工业出版社, 1998
- [33] Langelaar G C, Der Lubbe J C A Van, Legendijk R I., *Robust labeling methods for copy protection of images*, *Proc. SPIE 3022: EP'97, Storage and Retrieval for Image and Video Database, San Jose: SPIE*, 1997
- [34] John G Proakis, *Digital Communications*, McGraw Hill, 1995

- [35] 樊昌信, 廖道庸等, 通信原理, 国防工业出版社, 1995
- [36] 李华, 朱光喜, 朱耀庭, 基于视觉特性和小波分解的数字水印隐藏方法, 通信学报, Vol.21, No.6, June 2000
- [37] 马杜祥, 刘贵忠, 曾召华, 基于小波变换的数字水印及版权保护, 电子与信息学报, Vol.23, No.11, Nov. 2001
- [38] 韦志辉, 秦鹏, 欧阳宏彬, 基于小波域中视觉门限模型的数字水印技术, 东南大学学报, Vol.28, No.5, Sept. 1998
- [39] Mitchell D. Swanson, Bin Zhu, Ahmed H. Tewfik, *Multiresolution Scene-Based Video Watermarking Using Perceptual Models*, IEEE Journal on Selected Areas in Communications, Vol. 16, No.4, May 1998
- [40] Wenwu Zhu, Zixiang Xiong, Ya-Qin Zhang, *Multiresolution Watermarking for Images and Video*, IEEE Trans. on Circuits and Systems for Video Technology, Vol.9, No.4, June 1999
- [41] Hisashi INOUE, Akio MIYAZAKI, Takashi ARAKI, Takashi KATSURA, *A Digital Watermark Method Using the Wavelet Transform for Video Data*, IEICE Trans. Fundamentals, Vol. E83-A, No. 1, January 2000
- [42] Masataka EJIMA, Akio MIYAZAKI, *A Wavelet-Based Watermarking for Digital Images and Video*, IEICE Trans. Fundamentals, Vol. E83-A, No. 3, March 2000
- [43] Xiamu Niu, Shenghe Sun, Wenjun Xiang, *Multiresolution Watermarking for Video Based on Gray-Level Digital Watermark*, IEEE Transactions on Consumer Electronics, Vol. 46, No. 2, May 2000
- [44] 杨福生, 小波变换的工程分析与应用, 科学出版社, 2001
- [45] 陈武凡, 小波分析及其在图像处理中的应用, 科学出版社, 2002
- [46] 秦前清, 杨宗凯, 实用小波分析, 西安电子科技大学出版社, 1994
- [47] 徐佩霞, 孙功宪, 小波分析与应用实例, 中国科学技术大学出版社, 1996
- [48] 张益贞, 刘滔, Visual C++ 实现 MPEG/JPEG 编解码技术, 人民邮电出版社, 2002
- [49] 胡昌华, 张军华等, 基于 MATLAB 的系统分析与设计 - 小波变换, 西安电子科技大学出版社, 1999
- [50] 伍鹏, 张妙兰, 李象霖, 一种彩色视频图像序列的三维小波变换编码方法, 电子与信息学报, Vol. 23, No.5, May 2001
- [51] 易开祥, 石教英, 自适应二维数字水印系统, 中国图象图形学报, Vol. 6(4), No. 5, May

2001

[52] 张华熊, 仇佩亮, 置乱技术在数字水印中的应用, 电路与系统学报, Vol. 6, No. 3, September, 2001

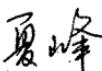
[53] 黄继武, SHI Yun Q, 一种自适应图象水印算法, 自动化学报, Vol. 25, No. 4, Jul., 1999

[54] Marc Antonini, Michel Barlaud, Pierre Mathieu, Ingrid Daubechies, *Image Coding Using Wavelet Transform, IEEE Transactions on Image Processing*, Vol. 1, No.2, April 1992

[55] 毛琼, 陈明奇, 夏光升, 杨义先, 谭铁牛, 安全数字水印体系的研究, 电子与信息学报, 2001 年 9 月

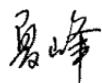
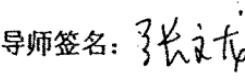
论文独创性声明

本论文是我个人在导师指导下进行的研究工作及取得的研究成果。论文中除了特别加以标注和致谢的地方外，不包含其他人或机构已经发表或撰写过的研究成果；其他同志对本研究的启发和所做的贡献均已在论文中做了明确的声明并表示了谢意。

作者签名：  日期： 2003/5/8

论文使用授权声明

本人完全了解上海师范大学有关保留、使用学位论文的规定，即：学校有权保留送交论文的复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容，可以采用影印、缩印或其它手段保存论文。保密的论文在解密后遵守此规定。

作者签名：  导师签名：  日期： 2003/5/8