

音频隐写术：两种具体的实现方法

原创

唠嗑! 于 2022-04-06 18:27:39 发布 260 收藏 2

文章标签: [学习](#) [网络安全](#) [音视频](#) [视频编解码](#) [语音识别](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/forest_LL/article/details/123979165

版权

目录

前言

一. 码表索引值替换方法

二. 窗口类型转换方法

2.1 举例MP3编码器相关的窗口

2.2 窗口类型的转换规则

2.3 隐写嵌入规则

系列文章

前言

本文章介绍两种音频隐写技术。

一. 码表索引值替换方法

音频编码器在熵编码过程中使用多个哈夫曼表来编码量化DCT系数, 所以相同的量化系数可以采用不同的哈夫曼码表进行编码, 此处可以理解为哈夫曼码表的选择存在冗余性。毫无疑问, 码表在选择时, 有最优选和次优选。因此, 压缩标准选择最优的哈夫曼表有两个判定依据, 如下:

1. 所选择码表中的码字必须能对编码区中数值最大的QMDCT系数进行编码(此为必要条件)。
2. 编码的比特总数要最小。

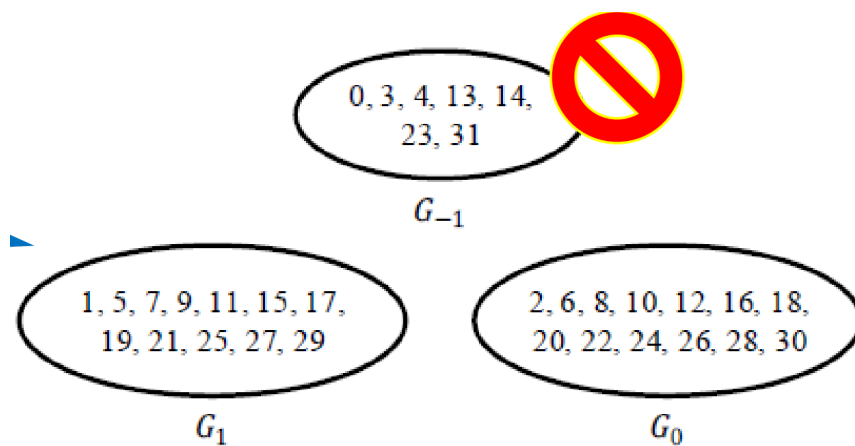
从这个方向出发, 可以利用标准中哈夫曼码表的冗余性来隐藏秘密信息而不引起明显的听觉变化。与此同时, 也可以构造特定的哈夫曼码表集合来编码比特信息。哈夫曼码表集合在划分时会有如下两个难题:

- 哪些是不可被利用的码表?
- 哪些码表可以用来嵌入比特“1”或比特“0”?

为了解决如上两个问题, 可以将MP3的大值区码表分成如下三类

1. 第一类 G_{-1} : 码表不用于消息嵌入
2. 第二类 G_1 : 码表用于嵌入消息比特“1”
3. 第三类 G_0 : 码表用于嵌入消息比特“0”

算法在分配码表时保持了码表索引的奇偶性与嵌入比特的一致性。据此哈夫曼码表可以分为如下三类:



当然码表集合的划分其实并不是唯一的，现举出分类 G_{-1} 的方法：

- 编码标准中第0号码表不含码字，且第4和14号码表均未被使用，所以应该被划归于 G_{-1} 。
- 按照码表映射规则，选择相邻的码表做替换，第3号和13号码表对应的替换码表应该为第4号和14号码表，在标准中他们属于保留码表，所以第3号和13号码表应该也归于 G_{-1} 。
- 第23号码表对应的替换码表为24号码表，但是由于第24号码表的linbits位比23号码表的短，可以编码的最大系数值更小，所以也应该归于 G_{-1} 。
- 第31号码表对应的替换码表为32号码表，但是第32号码表为小值区编码使用的码表，因此31号码表也不会被用来嵌入信息。

综上0号、3号、4号、13号、14号、23号、31号都属于 G_{-1} 。

除了分类码表外，还需要构建嵌入编码规则，可以理解作为一种码表映射规则。这个过程的主旨可以看成利用表中的替换规则对码表索引进行更改，实现消息比特的嵌入。

若当前码表索引所属的集合为 G_0 ，然而待嵌入的消息比特为“1”，那么就需要从 G_1 集合中选择对应的码表来做替换。在接收方，消息的提取过程则比较简单，只需要通过判定码表的索引值所属的集合，就可以获得所嵌入的消息比特。

码表映射的规则可见如下：

码表索引	替换后码表索引		码表索引	替换后码表索引	
	嵌入比特 “1”	嵌入比特 “0”		嵌入比特 “1”	嵌入比特 “0”
0	不变	不变	16	17	16
1	1	2	17	17	18
2	(3)	2	18	19	18
3	不变	不变	19	19	20
4	保留	保留	20	21	20
5	5	6	21	21	22
6	7	6	22	(23)	22
7	7	8	23	不变	不变
8	9	8	24	25	24
9	9	10	25	25	26
10	11	10	26	27	26
11	11	12	27	27	28
12	(13)	12	28	29	28
13	不变	不变	29	29	30
14	保留	保留	30	(31)	30
15	15	16	31	不变	不变

圆括号标记的地方则代表嵌入失效的问题：原因是码表映射规则是不封闭的。当然可以通过重复嵌入来解决此缺陷。

二. 窗口类型转换方法

在时频变换过程中，编码器需要根据信号变化的强烈程度来选择不同时宽的窗函数，由此来获得最佳的时域分辨率和频率分辨率。一般情况下，编码器可以选择很多种不同类型的窗函数，从这个角度出发可以利用窗口类型的冗余特点，在窗口类型和隐藏信息之间建立映射关系，通过窗口类型的替换实现信息的嵌入。

2.1 举例MP3编码器相关的窗口

MP3编码器主要采用了4种窗类型，包含：长窗、短窗、起始窗和结束窗。

- 长窗：用于变化平缓的音频帧，用较高的频域分辨率获得信号的能量分布
- 短窗：用于变化剧烈的音频帧，用较高的时域分辨率捕捉信号的瞬时变化
- 起始窗和结束窗：两种过渡窗，实现长窗和短窗之间的平滑切换

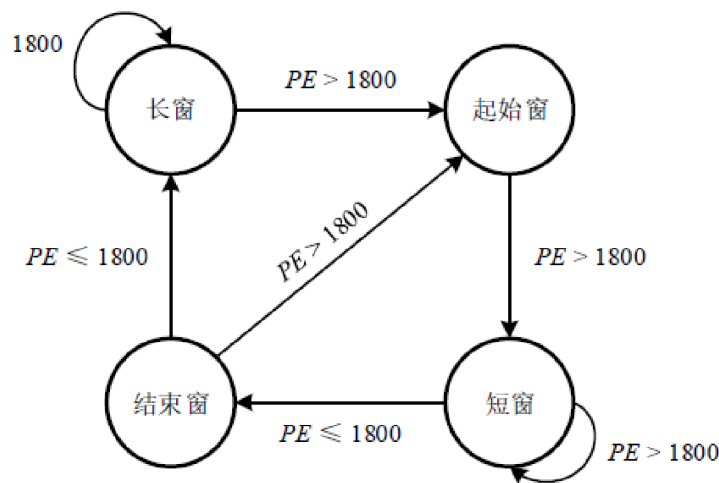
长窗的时宽是短窗的三倍，由此长窗的频域分辨率比短窗高三倍，但是时域分辨率比短窗低三倍。当然两者的过渡窗的时宽是一样的。

2.2 窗口类型的转换规则

在编码过程中，窗口类型的选择是依据心理声学模型（PAM），通过计算感知熵值（PE)来确定的。PE值的性质如下两点：

- PE值反应了信号频谱的平坦性，如果PE值越大则信号包含能量较强的高频分量，可以理解为时域信号存在瞬时的剧烈变化。
- MP3的标准规定，当PE值大于阈值1800时，窗口类型需要切换到短窗

窗口类型转换的规则示意图可见如下：



综合此图的理解，有两个重点：

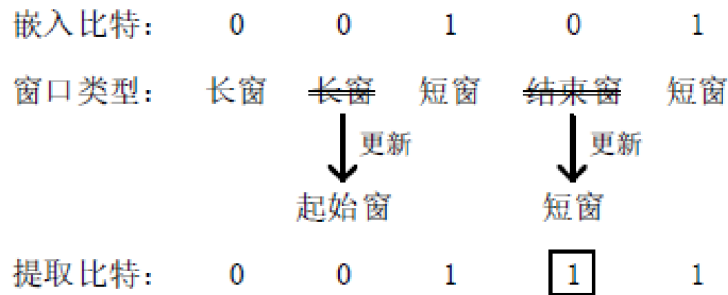
1. 当 $PE \leq 1800$ 时，若前状态为长窗或结束窗时，下一阶段选择长窗；若当前状态为短窗，下一阶段应该选择结束窗。
2. 当 $PE > 1800$ 时，下一阶段只能选择短窗。不仅如此，前一状态的窗口类型还需要进行更新，使用起始窗替换长窗、结束窗替换短窗。

2.3 隐写嵌入规则

隐写算法通过修改窗口类型的方式可实现消息的嵌入，由此需要约定嵌入消息与窗口类型之间的映射规则，如下两点：

- 当嵌入比特为1时，则当前窗口类型应该选择短窗，同时根据窗口类型的转换规则同时需要更新前一状态的窗口类型
- 当嵌入比特为0时，此时需要根据窗口类型的转换规则和前一状态的窗口类型选择不同的长窗、起始窗或者结束窗。

实际上，此算法在使用过程中是有算法缺陷的。若前一状态是结束窗时，则需要更新成短窗，由此就会导致嵌入比特由0变成1而发生提取错误。算法嵌入失效的实例如下：



为了解决算法的缺陷，需要尽量避免使用结束窗。当嵌入窗口类型变为结束窗时则判定此次嵌入失效，当前比特需要重新嵌入直到嵌入成功。在接收方端，在信息提取时如果遇到结束窗则跳过信息提取。

系列文章

音频隐写术：分析剑桥大学提出的MP3Stego算法_唠嗑！的博客-CSDN博客前言从信号处理的角度来看，隐写思想的本质就是将载密的噪声信号叠加到载体信号上，并且使得处理后的载体信号仍然保持感知透明性和统计不可检测性。音频隐写技术的发展经历了五个阶段，朴素的语音频隐写、编码内联的语音频隐写、自适应的语音频隐写和语音频隐写的新方向。如下图：朴素的语音频隐写阶段。第一阶段主要是解决语音频的听觉不可感知和隐藏容量等问题，使用的方法包括时域的低有效位隐写、回声隐藏、相位编码隐藏、扩频隐写，以及变换域隐写。编码内联的语音频隐写阶段。为了节省网络传输和存储的带宽，语音频数据一般..

https://blog.csdn.net/forest_LL/article/details/123960992?spm=1001.2014.3001.5501

隐写术基础_唠嗑！的博客-CSDN博客前言隐写术是一门关于信息隐藏的技巧与科学，所谓信息隐藏指的是不让第三者知晓信息的传递。隐写术的英文名叫做Steganography，起源于德国的一位修道士特里特米乌斯的著作《steganographia》。隐写技术提供对秘密信息存在性的保护，可以看成是一种保密通信技术和安全存储技术。数字隐写的载体包括音频、图像、视频、文本、网络包。一. 隐写系统模型音频隐写对抗模型包括三个实体和两个系统。三个实体：隐写者、接收方和隐写分析者。隐写者和接收方利用隐写系统来传递信息。隐写分析者利用隐写分析系https://blog.csdn.net/forest_LL/article/details/123953611?spm=1001.2014.3001.5501