

音频编解码speex库的使用方法

转载

newHung

于 2013-05-11 12:51:29 发布

阅读量1.5k

收藏 6

点赞数

文章标签:

C++

编码

Speex是近年来开发出的一套功能强大的语音引擎，能够实现高质量和低比特率的编码。它不仅提供了基于码激励线性预测（CELP）算法的编/解码；在其最新发布版本中还提供了声音预处理和声学回声消除模块，为保障IP网络中的语音通信质量提供了技术手段。此外，Speex还具有压缩后的比（2~44 kbps）的特点，并支持多种比特率。这些特点使得Speex特别适合 VoIP 的系统。

虽然是开源的，但是使用的人还不是很多，网上的资料也很少，下面简单给大家介绍一下如何使用speex的API。

一：编码流程

使用Speex的API函数对音频数据进行压缩编码要经过如下步骤：

- 1、定义一个SpeexBits类型变量bits和一个Speex编码器状态变量enc_state。
- 2、调用speex_bits_init(&bits)初始化bits。
- 3、调用speex_encoder_init(&speex_nb_mode)来初始化enc_state。其中speex_nb_mode是SpeexMode类型的变量，表示的是窄带模式。还有speex_wb_mode、speex_uwb_mode表示超宽带模式。
- 4、调用函数int speex_encoder_ctl(void *state, int request, void *ptr)来设定编码器的参数，其中参数state表示编码器的状态；参数request表示要定义的参数；SPEEX_GET_FRAME_SIZE表示设置帧大小，SPEEX_SET_QUALITY表示量化大小，这决定了编码的质量；参数ptr表示要设定的值。可通过speex_encoder_ctl(enc_state, SPEEX_GET_FRAME_SIZE, &frame_size)和speex_encoder_ctl(enc_state, SPEEX_SET_QUALITY, &quality)来设定数。
- 5、初始化完毕后，对每一帧声音作如下处理：调用函数speex_bits_reset(&bits)再次设定SpeexBits，然后调用函数speex_encode(enc_state, input_frame, &bits)中保存编码后的数据流。
- 6、编码结束后，调用函数speex_bits_destroy (&bits)， speex_encoder_destroy (enc_state)来

二：解码流程

同样，对已经编码过的音频数据进行解码要经过以下步骤：

- 1、 定义一个SpeexBits类型变量bits和一个Speex解码状态变量enc_state。
- 2、 调用speex_bits_init(&bits)初始化bits。
- 3、 调用speex_decoder_init (&speex_nb_mode)来初始化enc_state。
- 4、 调用函数speex_decoder_ctl (void *state, int request, void *ptr)来设定编码器的参数。
- 5、 调用函数 speex_decode(void *state, SpeexBits *bits, float *out)对参数bits中的音频数据进行解编码，参数out中保存解码后的数据流。
- 6、 调用函数speex_bits_destroy(&bits), speex_decoder_destroy (void *state)来关闭和销毁SpeexBits和解码器。

下面是一段实例代码：

```
01. #include <speex.h>
02. #include <stdio.h>
03. /*帧的大小在这个例程中是一个固定的值,但它并不是必须这样*/
04. #define FRAME_SIZE 160
05.
06. int main(int argc, char **argv)
07. {
08.     char *inFile;
09.     FILE *fin;
10.     short in[FRAME_SIZE];
11.     float input[FRAME_SIZE];
12.     char cbits[200];
13.     int nbBytes;
14.     /*保存编码的状态*/
15.     void *state;
16.     /*保存字节因此他们可以被speex常规读写*/
17.     SpeexBits bits;
18.     int i, tmp;
19.     //新建一个新的编码状态在窄宽(narrowband)模式下
20.     state = speex_encoder_init(&speex_nb_mode);
21.     //设置质量为8(15kbps)
22.     tmp=8;
23.     speex_encoder_ctl(state, SPEEX_SET_QUALITY, &tmp);
24.     inFile = argv[1];
25.
26.     fin = fopen(inFile, "r");
27.     //初始化结构使他们保存数据
28.     speex_bits_init(&bits);
29.     while (1)
30.     {
31.         //读入一帧16bits的声音
32.         fread(in, sizeof(short), FRAME_SIZE, fin);
```



newHung

关注

```
33.     if (feof(fin))
34.         break;
35.     //把16bits的值转化为float,以便speex库可以在上面工作
36.     for (i=0;i<FRAME_SIZE;i++)
37.         input[i]=in[i];
38.
39.     //清空这个结构体里所有的字节,以便我们可以编码一个新的帧
40.     speex_bits_reset(&bits);
41.     //对帧进行编码
42.     speex_encode(state, input, &bits);
43.     //把bits拷贝到一个利用写出的char型数组
44.     nbBytes = speex_bits_write(&bits, cbits, 200);
45.     //首先写出帧的大小,这是sampledec文件需要的一个值,但是你的应用程序中可能不一样
46.     fwrite(&nbBytes, sizeof(int), 1, stdout);
47.     //写出压缩后的数组
48.     fwrite(cbits, 1, nbBytes, stdout);
49. }
50.
51. //释放编码器状态量
52. speex_encoder_destroy(state);
53. //释放bit_packing结构
54. speex_bits_destroy(&bits);
55. fclose(fin);
56. return 0;
57. }
```

关于speex使用 (入门笔记) Zmn的排
未经本人允许禁止转载

音频编解码-speex库的使用方法 wmnmtm的排
Speex是近年来开发出的一套功能强大的语音引擎，能够实现高质量和低比特率的编码。它不仅提供了基于码激励线性预测（CELP）算法的编/解码模块，而且在其最新发

Speex编解码 了悟生死の排
以前在应用中使用到了Speex编解码，近来总结了一下Speex在android上的实现。Speex是一套主要针对语音的开源免费，无专利保护的音频压缩格式。Speex工程着力于

[总结]FFMPEG视音频编解码零基础学习方法 qq_1283679500的排
在CSDN上的这一段日子，接触到了很多同行业的人，尤其是使用FFMPEG进行视音频编解码的人，有的已经是有多年经验的“大神”，有的是刚开始学习的初学者。在和大

语音编解码speex，c代码iOS和Android平台。 qq_17524035的排
android ios speex编解码

android使用speex库对音频流进行编解码
本文将详细介绍如何在Android上使用AudioRecord获取音频流，并结合Speex库进行编解码。 1. **音频录制与处理** Android系统提供了`AudioRecord`类用于捕获设备麦

音频编解码库Speex源码 的排
Speex是一套主要针对语音的开源免费，无专利保护的音频压缩格式。Speex工程着力于通过提供一个可以替代高性能语音编解码来降低语音应用输入门槛。

STM32音频编解码 的排
5. **解码**：当需要播放时，STM32通过Speex库对压缩的音频数据进行解码，恢复成原始的音频样本。 6. **后处理**：解码后的数据可能还需要经过一些后处理，如滤波

STM32音频采集播放speex解压缩.zip
主控用的是STM32F103RCT6，用自带的AD/DA进行音频的采集播放，通过DMA传送，使用了speex对采集的数据进行解码编码

基于STM32和Speex编解码的录音设计与实现.pdf 的排
基于Speex算法的音频编码压缩部分是指使用Speex编解码算法来对采集到的音频信号进行编码和压缩。Speex编解码算法是一种开源的语音编解码算法，具有高质量、低


Speex C# 封装 的排
C# 封装Speex

speex API 使用手册 的排
The Speex API Reference Manual provides information on programming with libspeex.

speex的下载、编译、测试和在android推流端的集成 wensan911的排
目录 一、概述 二、实例 三、总结 一、概述 二、实例 https://github.com/xiph/speex/tree/master，这个是codec库 https://github.com/xiph/speexdsp，现在只是测试一下s

安卓使用speex开源库 里昂的排
Speex是一套主要针对语音的开源免费，无专利保护的音频压缩格式。Speex工程着力于通过提供一个可以替代高性能语音编解码来降低语音应用输入门槛。另外，相对

speex库处理音频编解码in即时通讯开发 u011473714的排
最近在做一个PC端即时通讯软件，类似RTX，在音频数据处理的过程中用到了Speex库。最近需要做一个基于udp的实时语音聊天的应用，语音流的压缩方面，我选择了

 newHung

关注