The Design and Implementation of Speech Engine based on Speed

Fan Zhang

Hubei Open University, Wuhan, China

Keywords: speech coding; Speed; speech engine.

Abstract. This paper will take the foundation of speech coding as the breakthrough point, by means of the interpretation of Speed, it discusses the design of the speech engine as well as the route of implementation so as to reduce tedious work in voice testing.

Introduction

Speech coding is the basic technology of digital speech transmission and storage, by means of the compressed digital, it can represent the speech signals and make the expression of these signals with the minimum number of the required bits. Compared with the stimulated voice, digital voice transmission and storage system of using speech coding technology, has the advantages of high reliability, strong anti-interference ability, easy to be quickly exchanged, easy for the realization of confidentiality, multiplexing, packaging as well as the advantage of low price, etc. The compressed voice is used for transmission, which can reduce the required bandwidth of each route, thus it can transmit mote voice transmission in the same bandwidth; it can be used for storage, which can save space and improve the storage of speech length as well as reduce cost.

The Interpretation of Speex. Speex is a multi-mode, multi-rate, speech code, based on CELP algorithm, it can provide narrow band, wide band and ultra-wide band three speech codec modes, which are respectively corresponding to the speech signals with the bandwidth of 4kHz (sampling rate is 8000), 8kHz (sampling rate is 16000), 16kHz (sampling rate is 32000). Among them, the narrow band speech coding only adopts narrow band sub-pattern coding; wide band speech can be divided into two sub bands,wide band speech adopts broadband sub-pattern coding, low band voice adopts narrow band sub-pattern coding; ultra-wide band will repeat decomposition two times which adopts the wide band sub-pattern coding twice and narrow band sub-pattern coding once . Thus, we can see, throughout the algorithm of Speex, it is composed by two types of sub-pattern encoding: narrow band sub-pattern and wide band sub-pattern.

The Structure of the Functional Module of the System. In this paper, VOIP system adopts Speed coding. The whole system consists of two parts, the server side and client side.

The server of the database saves the data of all registered users. Each user must firstly log on the server used by the client side, and access to the list if online friends, then it can make a voice call to the online friends. The call between the client sides adopts the mode of point to point. It can avoid the excessive delay of voice packet caused by the transmission of the server side.

The Design of the Server Side. The server side can not only save the user's information in addition to maintaining a database, but also can command and manage each client side. Once the server is started to start the service, it is started to listen for the requests of users. The server receives the message sent by the client side, firstly, it sends back the confirmed information and then it can establish a separate thread to deal with the received data. In this separate thread, according to the category of the received data it has the corresponding treatment.

The Design of the Client Side. The function of the client side can be divided into two parts: one part is interacted with the server side, which can obtain the relevant information from the server; the other part can complete the communication between the different clients point to point. Among them, the second part is the core of VOIP system.
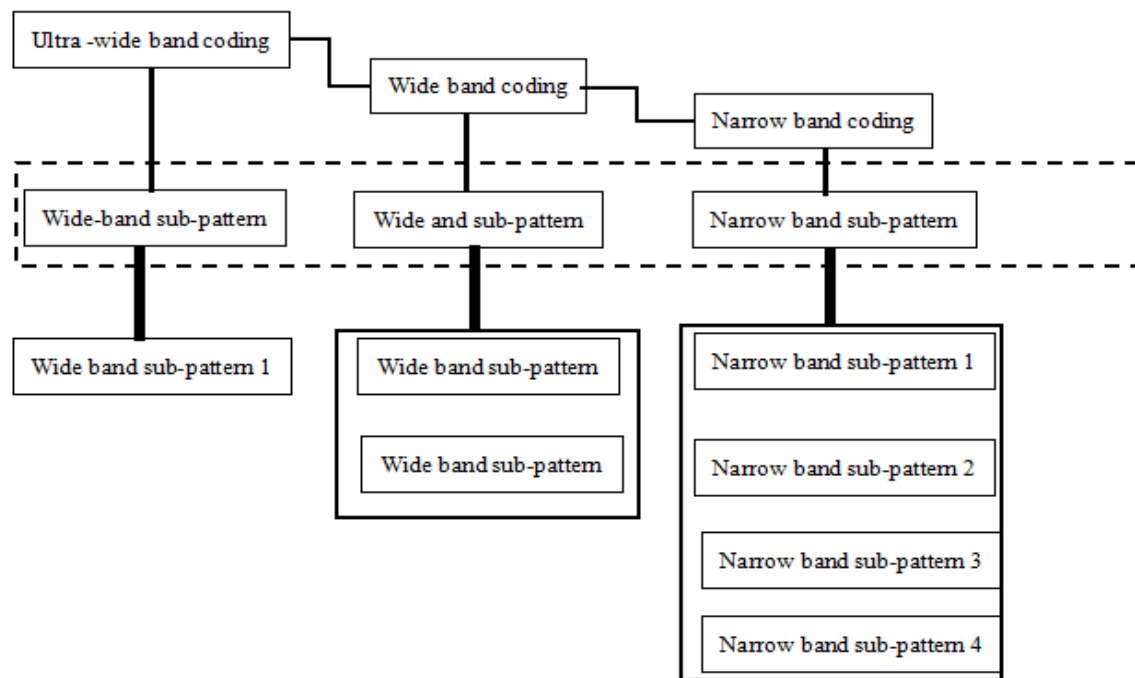
Fig 1. Decomposition of coding mode

After starting the program of the client side, if there is local users' information, then it can load the local users' information, and display the login window; if not, then it can display the user's registration window (in the login window, it also can choose the user's registration). Users can receive the request of voice call sent by friends, at the same time according to the user's operation, the client side can give response to friends, answer or reject.

The Implementation of Speech Engine

The Hierarchical Design of Speech Engine. The Inst Audio layer is located at the upper layer, involved with the interaction of interface; Voice Engine layer is in the middle layer, which is the middle part to play the function of calling different kinds of speech database; Speex base layer is codec library, which is the specific location that can implement the Speex codec as well as the process of mixing operation.

    The Introduction of Components. The starting function of the local voice. As the entry function of the local voice in the conversation of each window, it has speech right, here, modifying the corresponding speech right of the local voice in chain P1 to be true, at the same time, finding out if there is the corresponding element of chain P2, if not, joining the corresponding element in chain P2, so as to check whether the thread is opened or not, if not, open the thread, otherwise skip it without processing. This function is in Voice Engine layer.

    Encoding thread function. This function as the implementation of encoding function, namely encoding thread function, appeared as an independent running thread. This function is in Speex layer.

    The termination function of the local voice.

    Such function is responsible for dealing with voice conversation between the local voice and either end of the opposite terminal when the local voice loses the speech rights. This function is in Voice Engine layer.

    Data sending function. This function will be responsible for sending the compiled data by the callback method. This function is in Voice Engine layer.

    The starting function of the opposite terminal. As the entry function in the conversation of each window, the opposite terminal has the speech right, her modifying the corresponding speech right of the opposite terminal in chain P1 to be true, at the same time, finding out if there is the corresponding element of chain P3, if not, joining the corresponding element in chain P3, so as to check whether the

thread is opened or not, if not, open the thread, otherwise skip it without processing. This function is in Voice Engine layer.

Decoding thread function. This function as the implementation of decoding function, namely decoding thread function, appeared as an independent running thread. This function is in Speex layer.

Data receiving function. This function is responsible for dealing with the data sent by the opposite terminal, after acquiring the data, it can check whether the data has the corresponding data sub-chain in chain P4, if it has, add it at the end of the data sub-chain, if not, add the data sub-chain in chain P4.This function is in Voice Engine layer.

The Relationship between Components

The call graph of the relationship between components can be shown in Figure 2.
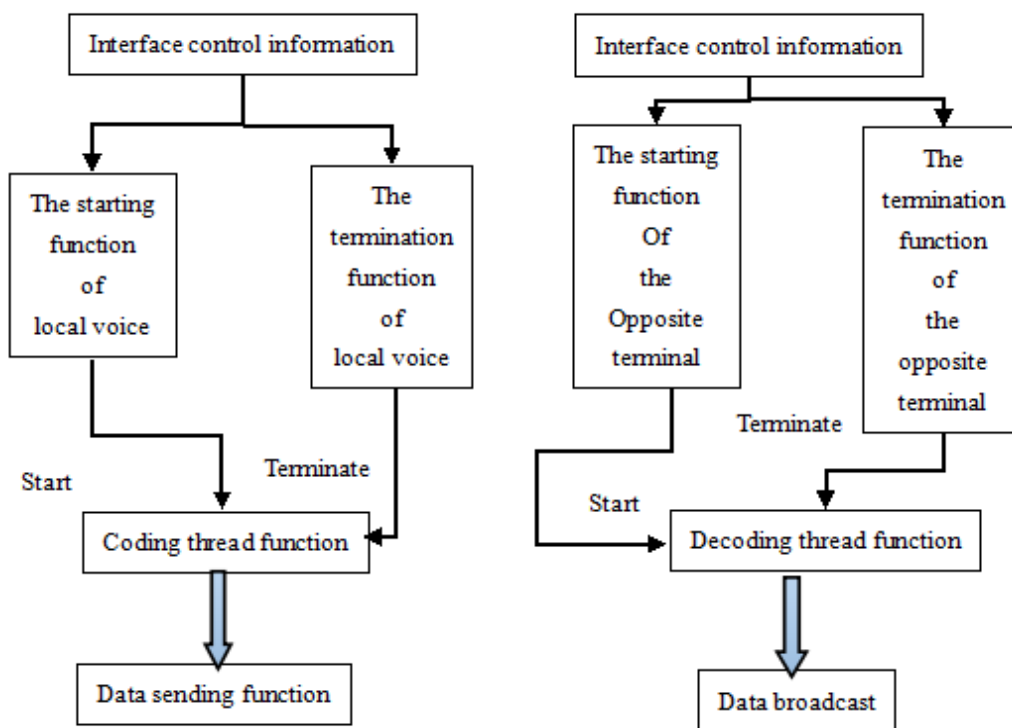

Figure 2 The call graph of the relationship between components

When the voice call begins, first of all, it should accept interface control information. This information can call the starting function of local voice and the starting function of the opposite terminal, judging whether it has opened coding thread or decoding thread, if it is not opened, then open it. The data after the coding thread is dealt with the data sending function, the data solved by decoding thread is dealt with the sound card to play.

When the voice call ends, first of all, it should accept interface control information. This information can call the termination function of local voice and the termination function of the opposite terminal, judging whether it has opened coding thread or decoding thread, if it is not opened, then open it.

Conclusion

In this paper, the implementation of the speech engine based on Speex has already been operated, and it has performed well at this stage under the environment. It has showed the efficiency of Speex as the voice codec, as well as the processing ability in voice communication.

Reference

[1]SAVOJIM H.1989, A robust algorithm for accurate endpointing of speech. Speech Communications, vol. 8, p 45-60.

[2]John M. Paulett, Curtis P. Langlotz. 2009, Improving language models for radiology speech recognition. Journal of Biomedical Informatics,vol.42, p53–58.

[3]Rabiner L,Juang B. 1993,Fundamentals of Speech Recognition. Englewood Cliff New Jersey Prentice-Hall.

[4]Pitton J,Atlas L,Loughlin P. 1994,Aplications of PossitiveTime-frequeney Distributions to Speeeh Proeessing. Speeeh and Audio Proeessing,vol.24, p544-566.