Journal of Electronic Design Technology
ISSN: 2229-6980

www.stmjournals.com

Text-To-Speech (TTS) Conversion System for Gujarati Language

Jayesh Tanna**, Vijay Savani, Amit Degada
Department of Electronics and Communication Engineering, Nirma University, Ahmedabad, India

Abstract

Text-to-speech (TTS) conversion system enables user to enter text in Gujarati language and as an output it generates the equivalent sound. This type of system will be greatly useful for illiterate and vision-impaired people to hear and understand the content. TTS systems are still suffering from the problem of producing emotional speech like that of human being. Scientists are trying to give emotions and feelings to it. This shows that research work can be done to enhance the efficiency of TTS system. There are TTS systems that are under development for many languages other than Gujarati language. So, we are proposing the TTS system, which will work on Gujarati language. The input of our desired system is typed or scanned Gujarati text and equivalent Gujarati speech with smooth flow will be generated as an output. We are trying to add a new feature through which we can hear our own voice by superimposing our own voice frequency on the pre-recorded synthesized speech, so that we can listen to any text in our own voice. This paper starts with the introduction to the fundamental concepts of TTS synthesis. So, it will be useful for the readers who are less familiar in this area of research.

Keywords: *TTS, synthesis, phoneme, phonetics, diphone, intonation, pitch*

****Author for Correspondence*** E-mail: tanna.jayesh27@gmail.com

INTRODUCTION

The fundamental function which has to be performed by a text-to-speech (TTS) system is that it has to generate a clear linguistic sound. This process of generating artificial sound is known as speech synthesis and the system, which can be used for this purpose, is known as speech synthesizer. This can be implemented by using software and also hardware as per our requirement and resources available. We are going to implement the software (MATLAB) first and afterwards we will implement a standalone device for this system.

A TTS system converts a normal language text into phonetic speech. Phone is the smallest unit of sound in any language and phonetics is the branch of acoustics concerned with speech process including its production and perception. There are basically two parts in a TTS system: front end and back end. The front-end converts the input text, which contains symbols, numbers and abbreviations into equivalent pronunciation of those words. This process is known as text normalization. Back-end performs basically two operations: text-to-phoneme conversion and phoneme-to-sound conversion. The process of mapping each word to its equivalent phone is known as text-to-phoneme conversion. The process of mapping each phoneme to its equivalent sound is known as phoneme-to-sound conversion [1].

There are two main characteristics for measuring the performance of any TTS system: naturalness and intelligibility. Naturalness shows how closely the TTS system-generated sound seems like human speech while intelligibility is how easily the TTS understands the input text and generates the sound. Thus, the ideal speech synthesizer should have both the characteristics of naturalness and intelligibility. So, our main goal is to maximize both the characteristics of TTS at the level of acceptance [2].

The paper begins with a brief introduction of the speech synthesis and the fundamentals of

TTS conversion. Section 2 describes about the block diagram of the TTS system in general. Section 3 provides text normalization process in Gujarati language and the proposed work in Gujarati TTS converter system. Section 4 describes the comparative analysis of different types of synthesizer technologies. Section 5 gives brief idea about applications of the TTS system.

BLOCK DIAGRAM OF TTS SYSTEM

The overview of the TTS system is shown in Figure 1. After having glance on this block diagram, anyone can understand the working and function of TTS system. In our TTS system, scanned or typed text is applied as an input and at the output we get sound/pronunciation/speech equivalent to the related word which is the heart of the system linguistic analysis. This part is purely responsible for the pitch and intonation of the output speech. To build a very good linguistic analysis module has been the biggest challenge for researchers till now, though research is going on since last many years. Because of this limitation of linguistic module accuracy, we cannot generate pure natural sound as of a human being from the TTS system. Researchers are trying to give emotions and feelings in the generated speech by TTS systems.

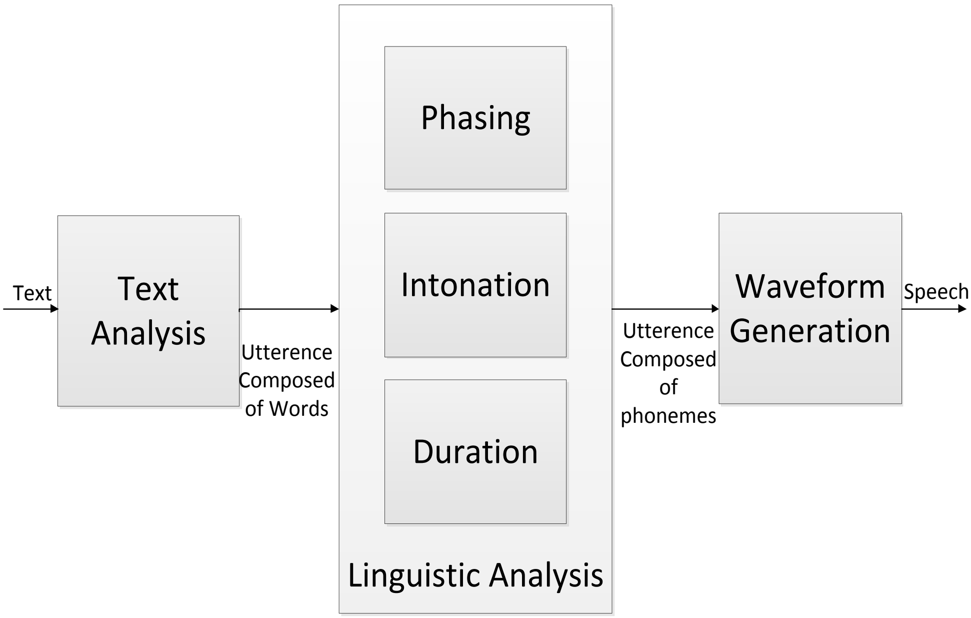

Fig. 1: *Block Diagram of TTS System.*

The main function of text analysis block is to perform text normalization, which deals with symbols, numbers and abbreviations. For example, in Gujarati language “ચિ.”is written but it is pronounced as “ચિરંજીવી” which means “having longevity.” But, in Gujarati language the number of this type of abbreviations is very few, so it is an easy task for TTS. Sometimes “૧૨૩૪” is pronounced as “એક બે ત્રણ ચાર” and sometimes as “એક હજાર બસો ચોત્રીસ”. All these issues have to be taken care of by the text analysis block. After that, linguistic analysis block comes into the picture, whose main function is to perform text-to-phoneme conversion in which it maps all the Gujarati text characters to their equivalent phonemes. It also provides prosody (the patterns of stress and intonation in a language), intonation (rise and fall of the voice speech) and pitch (fundamental frequency) to the individual phonemes. At last, the waveform generation block performs filtering operation to smoothen out the generated speech by system.

TEXT NORMALIZATION PROCESS IN GUJARATI LANGUAGE

In the text normalizing part, all the pixel values are converted to ‘0’ and ‘1.’ Pixel having gray value above threshold is converted to ‘1’ and below threshold is converted to ‘0.’ For perfect result of TTS system, first, we need to seperate out lines from the scanned document, then words from lines and at last characters from the words. Again, maybe these characters have heads and tails, so again we need to separate them out [3].

For seperating lines in a document, horizontal histogram of the document is useful. The blank space between the two lines indicates the seperation of the lines. To separate down the words from the lines, we need to take vertical histogram. For example, The histograms of characters ‘અ’ and ‘હ’ are shown in Figures 2 and 3 respectively [3].

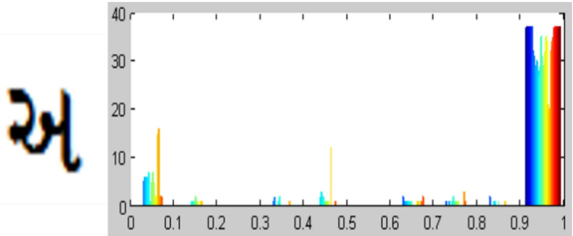

Fig. 2: Histogram of Character

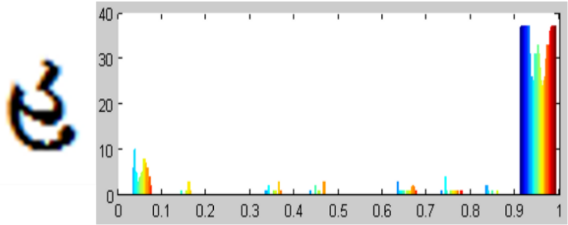

Fig. 3: Histogram of Character ‘હ’.

All the characters of Gujarati language can be classified in the following four groups:

- Groups of characters having vertical line at the end: ‘અ’, ‘ખ’, ‘ગ’, ‘ઘ’, ‘ચ’, ‘ણ’, ‘ત’, ‘થ’, ‘ધ’, ‘ન’, ‘બ’, ‘ભ’, ‘મ’, ‘ય’, ‘લ’, ‘વ’, ‘શ’, ‘ષ’, ‘સ’, ‘ળ’, ‘ક્ષ’, ‘જ્ઞ’
- Group of Characters with no vertical line: ‘ક’, ‘છ’, ‘જ’, ‘ઝ’, ‘ટ’, ‘ઠ’, ‘ડ’, ‘ઢ’, ‘દ’, ‘ફ’, ‘ર’, ‘હ’
- Group of heads: ‘િ’, ‘ી’
- Group of tails: ‘ુ’, ‘ૂ’

After seperating the characters from lines, first it is to be decided to which group each character belongs. After identifying the proper group, we have to identify the exact character by using different methods. When TTS system comes to know the exact character, it will be assigned a unique number, which has already been stored in the database, and by using that number our TTS system will do mapping to its equivalent phoneme[3].

TTS SYNTHESIZER TECHNOLOGIES

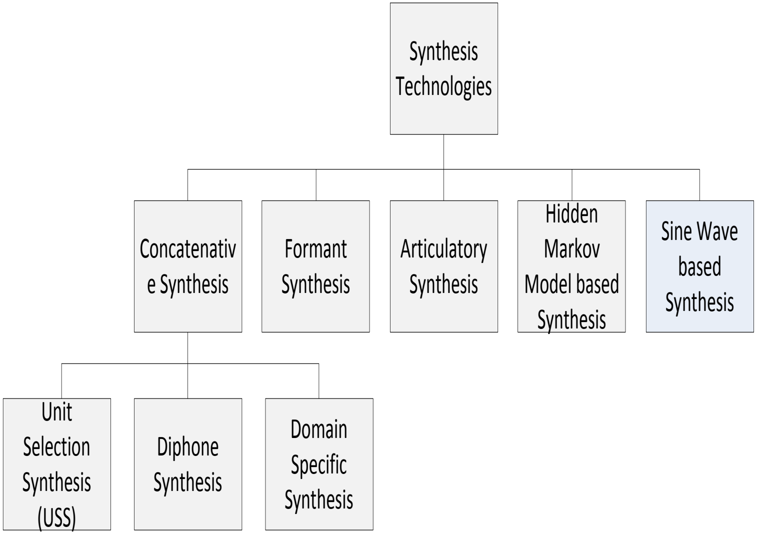
Synthesis Technologies
Concatenative Synthesis
Formant Synthesis
Articulatory Synthesis
Hidden Markov Model based Synthesis
Sine Wave based Synthesis
Unit Selection Synthesis (USS)
Diphone Synthesis
Domain Specific Synthesis

Fig. 4: Hierarchy of Synthesizer Technologies.

Figure 4 shows the hierarchy of TTS synthesizer technology. Mainly there are five types of TTS synthesizer technologies: concatenative synthesis, formant (a main band of frequency) synthesis, articulatory synthesis, hidden Markov model (HMM)-based synthesis and sine-wave based synthesis. Concatenative synthesis technology is widely accepted and used to develop any language-dependent TTS system. Concatenative synthesis uses the pre-recorded sounds of each and every word and maps them with the equivalent input text words. This type requires very large memory space for storing all the pre-recorded sounds of each and every word, maybe in terms of gigabytes (GBs). This technique gives the most natural sound in the output of TTS system because the recorded clips are already in voice of some person [4].There are three subparts of concatenative synthesis technology: unit selection synthesis (USS), diphone synthesis and domain-specific synthesis. USS uses large database of pre-recorded speech. The smallest recorded segment in the database is known as "unit" which can be a phoneme, diphone, syllable, word or even a sentence. After applying text as an input, the system searches out for the best candidate unit (which is done by giving each unit a unique index) and concatenates at the output to generate full natural speech (Table 1).

Table 1: Comparison of Different Synthesis Technologies.

Synthesis technology	Pros	Cons
Concatenative synthesis	Most natural sound, almost 100% efficiency if sound of word is stored in the database	Large speech database, tedious method, time consuming-method
Unit selection synthesis (USS)	Highest accuracy and most natural sound	Large database of different units like phoneme, diphone, syllable, etc.
Diphone synthesis	Minimal speech database containing all the diphones occuring in a language	Lesser accuracy and naturalness than USS, suffers from glitches
Domain-specific synthesis	Fast, responsive and natural, less memory requirement due to predecided words	Application is only that particular domain
Formant synthesis	Least memory and power requirement, used in embedded system, efficiency is 70–75%, program size is smaller	Robotic sound in output, very low naturalness
Articulatory synthesis	Memory requirement is almost nothing	Computation intensive (high complexity)
HMM-based synthesis	Very little memory, sound with better prosody, produces natural sound, efficiency more than 90%	Consumes largae CPU resource, complex in structure
Sine-wave-based synthesis	Replace formants with pure tone whistles	Rarely used

Diphone synthesis has the database of all the diphones in a language which is much more less than that of USS. The number of diphones depends upon the language itself. This method suffers from glitches of concatenative synthesis. But it requires less memory space than the USS. Domain-specific synthesis concatenates the pre-recorded words or sentences from the database to create complete utterances of the input text. This technique is used where a limited number of words or sentences are used, e.g., at airports, railway stations or bus rapid transport services (BRTS), where instructions have to be announced repeatedly with limited words and sentences. The level of naturalness in this technique is very high because of the limited number of pre-recorded speech [4]. Formant synthesis produces the robotic speech and does not use human speech samples at runtime. Thus, it requires very less memory and power; so this technique can be used in embedded systems, where memory and power are big constraints. The drawback of this system is the complexity and lower level of naturalness. This technique is reliable and intelligent enough to work at higher speeds too. This type of synthesizer can be created by smaller programs than concatenative systems because the absence of the pre-recorded database of speech [4]. Articulatory synthesis is the technique, in which human articulatory system is modeled (jaw, tongue, teeth, etc.) and simulated how the airflow passes through them and produces the sound as per our requirement. This system is not used due to its very high complexity level and low performance and efficiency level. Memory requirement is almost nothing for this type of technique. HMM-based synthesis system is also referred to as a "statistical parametric synthesis." In this technique, frequency spectrum (vocal tract), fundamental frequency (vocal source) and duration (prosody) of speech are modeled simultaneously by HMMs. Speech waveforms are generated from these models based on maximum probability criterion. This technique consumes large CPU resource but very little memory. In addition, this technique gives better prosody without glitches and still produces the most natural sound [5].In sine-wave synthesis, formants (main bands of energy) of the synthesized speech are replaced by pure tone whistles (sine waves with specific frequency) to produce the synthesized speech. This technique is generally used as an internal module of another device because as a standalone unit, it cannot produce human understandable sound.

APPLICATIONS OF TTS SYSTEM

TTS conversion is mainly useful for the blind people or physically handicapped people to communicate with others. Stephen Hawking is a famous personality, who is paralyzed for decades by Lau Gehrig’s disease. Currently, braille language is available for the blind people to read something, but to convey the information in braille language is very difficult and is equivalent to 4 pages as compared to normal 1 A4 size page in English language. So, very few books are available in braille language. If they want to read the normal printed book, then TTS system can be useful.

If someone does not have time to read the e-mails, they can listen to them directly or by recording them in mp3 or other file formats.

TTS is used for continuous announcements at the airports, railway stations, etc. For making computer or any electronics device conversational, TTS is very useful.

FUTURE SCOPE

We thought to develop a device which has the entire speech database in one particular frequency (single person). Now, when in the regular TV serials, an actor/actress suffers from cough and their voice changes, we can superimpose the original voice’s frequency on the synthesized speech, generated by TTS module. Thus, in the same way, when celebrities and politicians do not have time for recording their speech, we can again superimpose their original voice’s frequency on the synthesized speech, generated by our TTS system. Thus, we can also listen to any songs in our own voice. We want to develop a system which can recognize every type of Gujarati fonts. It means TTS should be font independent. In future, it can be implemented with different types of handwritten Gujarati text also.

CONCLUSIONS

At the end, we can say that TTS system is very useful in many application areas. Here, we have proposed the idea for developing the TTS system which can generate the synthesis speech in our own voice. This system is very useful for the celebrities or politicians, who even do not have time to record their speeches. So, by recording just a template speech in their voice, we can superimpose that voice’s frequency on our TTS-based generated speech. We can get the entire speech in their voice by using our proposed device.

REFERENCES

1. John F. Pitrelli. The IBM Text-to-Speech synthesis system for American English. In: *IEEE Transactions on Audio, Speech and Language Processing* July 2006; 14(4).
2. Ian McLoughlin. Applied Speech and Audio Processing with MATLAB Examples. Cambridge Publications.
3. Prajakta S. Rathod. Script to speech conversion for Hindi language by using artificial neural network. In: *NUiCONE-*2011.
4. Aimilios Chalamandaris, Sotiris Karabetsos. A unit selection text-to-speech synthesis system optimized for use with screen readers. In: *IEEE Transactions on Consumer Electronics* August 2010; 56(3).
5. Chien-Liang Liu, Wen-Hoar Hsaio, Chia-Hoang Lee, et al. An HMM-based algorithm for content ranking and coherence-feature extraction. In: *IEEE Transactions on Systems, Man and Cybernetic Systems* March 2013; 43(2).