2021

MATLAB Speech Synthesizer

MID-TERM REPORT

LACHLAN DOW

Introduction

Voice assistants, speech systems (the most famous example of which is Stephen Hawking), train announcements. Synthesizers are all around us every day, but often we don't even notice them.

Aims

The aims of this project are to produce a MATLAB voice synthesizer that takes inputted text and runs it through procedures and algorithms to allow the system to "speak" the text out loud. The system will allow for people who have vision impairments to have sections of text read out to them. There are many systems out there that provide Text-To-Speech features, but the aim of this system is to provide an interface that has few to no technical jargon and clear and simple instructions. Providing this cut down interface means that the system itself will need to be proficient in the translation of text to make the sound as intelligible as possible. The system aims to do this through incorporating parts of speech that improve naturalness including but not limited to, pausing, intonation, stressing consonants etc.

Background

A large amount of work and research has gone into the field of speech synthesis through people using mechanical devices, whether in the form of "speaking heads" in 1003 AD or in 1779 when a Danish scientist-built models of the human vocal tract that could produce 5 long vowel sounds.

It was not until the late 1950s that early speech synthesis through electronic devices was created. It would not be another decade until the first full text to speech synthesis system was created in 1961 by John Larry Kelly.

There are many ways of approaching speech synthesis – these include concatenative synthesis -which include unit selection synthesis, diphone synthesis, domain specific synthesis, articulatory synthesis, HMM-based synthesis, sinewave synthesis, and formant synthesis. Due to the scope of this report, the only examples of these that will be explored in more detail is concatenative synthesis (including a couple of the sub-forms) and Formant synthesis. Information on the other forms can be found in ((Speech synthesis, 2021))

Concatenative synthesis consists of stringing together segments of recorded speech to produce the synthesis of inputted text. This form of synthesis is known to produce the most natural sounding speech, but because of the variances in how people speak, and the way segments of speech are extracted, there can be glitches when piecing together the different sections. This model of synthesis can be performed in multiple ways mentioned below:

- Unit selections synthesis, which uses a large database of recorded speech when each recorded utterance is segmented in different ways using a specially designed speech recognition system. These segments are indexed based off defining factors e.g., fundamental frequency, duration etc.
- Diphone synthesis, which uses a database containing all the diphones occurring in a language. Only one example of each diphone is contained in the speech database, then at runtime the prosody of a sentence is superimposed on the units, usually through a linear predictive coder (LPC).
- Domain-specific synthesis, which concatenates pre-recorded words and phrases to create complete utterances, is generally only used in speech variance as it will be very limited, for example a train announcement system or weather reports.

Formant synthesis on the other hand, does not use human speech samples but instead is created using a model to produce speech sounds. Different variables such as fundamental frequency and voicing noise levels are changed over time to create a waveform that represents speech. Formant based synthesis systems generally generate robotic sounding speech, for example the system that Stephen Hawking used. This results in the speech never really being mistakeable for human speech, but because of the way these synthesizers are made, they are very intelligible consistently without the glitches found in concatenative systems. Formant systems also have the advantage of being able to produce a wide variety of prosodies and intonations to be outputted, allowing them to produce questions, statements, emotions and many voice tones.

The main features of the MATLAB speech synthesizer are as follows: –

- The system will be able to produce word sounds.
- The system will be able to produce these word sounds and piece them together into intelligible sentences.
- The system will take user input in the form of text and process it into synthesised speech.
- The system will be able to produce the outputted voice in 5 seconds or less.
- The interface will be accessible and easy to understand.
- The system should output words and sentences that are easily understood by an English speaker.
- The system should always output speech for any inputted text.
- The system should be built in a way that is maintainable and can be adjusted with additional features added that improve functionality.

Progress

The main section of the project from the 28/10/2020 to the 15/12/2020 was conducting research into the different types of speech synthesis systems already in use, including those mentioned above. From this research, construction of a basic concatenation synthesizer was undertaken, one that could stitch together basic diphone speech units to create words. Through this process many glitches and errors were created in the production of the sound, resulting in words being difficult to understand. In a meeting with the project advisor. It was brought up that this was not the most effective way to produce comprehensible speech sounds so a formant synthesizer was introduced as a better alternative. After this meeting research was conducted into these speech systems, including the MITalk system and the Klatt formant synthesizer using the cascade /parallel system (Klatt, 1980). Production was then retargeted to producing a speech synthesizer based on this model. An object-oriented approach has been taken, producing different parts of the synthesizer setup through objects in MATLAB. So far in the construction, the objects of the resonator, the anti-resonator, and the low pass resonator have been created, as well as the white noise generator in the system Parts of this system can been seen in Figure 1.Highlighted are areas that have been completed.

Personal reflection-

The project has been challenging, largely because speech synthesis is such a broad subject, that knowing where to start was a huge task and it has taken a large amount of time feel like progress has been heading in the right direction.

Since a great amount of time was lost due to the initial focus on the diphone synthesizer, work on the redesign and production of a formant synthesizer will take time but

progress has been quick and testing has been positive.

Plans for the Remainder of the Project

Finishing the construction of the Klatt formant synthesizer in MATLAB, the main building blocks of this system are in place but need to be combined into a fully functioning system.

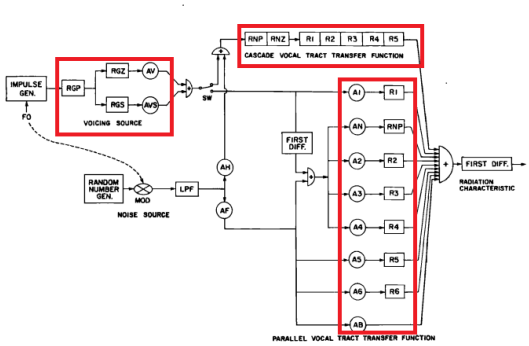

Figure 1 Block diagram of Klatt cascade/parallel formant synthesizer with completed in red outline.

Creating the input text pre-processor:
Input text will need to be processed so that it can be used by the synthesizer. There are 39 parameter values, and on any given step, 20 may need to be changed to produce the formant. The values for this will need to be stored and referenced when required, based on the pre-processed data obtained from the text.

Going forward, an interface will also be constructed for any user to use the synthesizer. Designs for this will be made and then implemented in MATLAB.

Testing of the system will also need to be conducted to ensure that the speech emitted from the system is understandable. These tests include but are not limited to – MOS, Rhyme, AB test based on testing standards document (2021).

A progress plan has also been created to approach the tasks that need to be completed to ensure that the project is delivered on time:

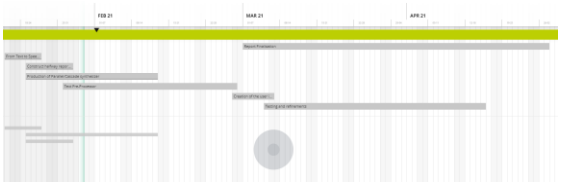

Figure 2: Gantt Chart for the remainder of the project

Working Klatt formant synthesizer – 12/2/21

Text Pre-processor – 26/2/2021

Creation of the user interface linking of Text pre-processor and formant synthesizer– 5/3/2021.

Testing and refinements – 15/4/2021

Report finalisation- 27/4/2021

With this schedule in mind the creation and completion of the synthesizer should be completed on time with all the main features mentioned before implemented.

References

Cs.mcgill.ca. 2021. *Speech Synthesis*. [online] Available at: <https://www.cs.mcgill.ca/~rwest/wikispeedia/wpcd/wp/s/Speech_synthesis.htm#:~:text=The%20first%20computer%2Dbased%20speech,the%20history%20of%20Bell%20Labs.> [Accessed 26 January 2021].

Klatt, D., 1980. *Software For A Cascade/Parallel Formant Synthesizer*. [online] Fon.hum.uva.nl. Available at: <https://www.fon.hum.uva.nl/david/ma_ssp/doc/Klatt-1980-JAS000971.pdf> [Accessed 20 January 2021].

Wiki.inf.ed.ac.uk. 2021. [online] Available at: <https://wiki.inf.ed.ac.uk/twiki/pub/CSTR/Speak14To15/evaluation.pdf> [Accessed 27 January 2021].

Jonathan Allen, M. Sharon Hunnicutt, Dennis H. Klatt, Robert C. Armstrong, and David B. Pisoni. 1987. *From text to speech: the MITalk system*. Cambridge University Press, USA.