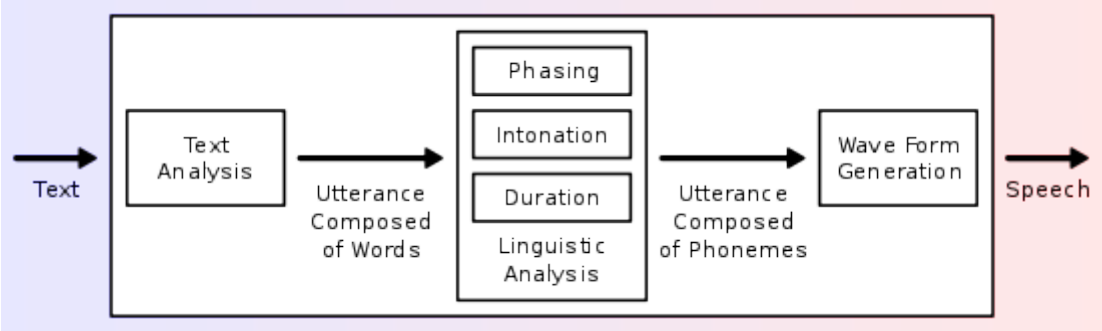
Speech synthesis

Speech synthesis is the artificial production of human speech. A computer system used for this purpose is called a **speech computer** or **speech synthesizer**, and can be implemented in software or hardware products. A **text-to-speech** (**TTS**) system converts normal language text into speech; other systems render symbolic linguistic representations like phonetic transcriptions into speech.[1] The reverse process is speech recognition.

Synthesized speech can be created by concatenating pieces of recorded speech that are stored in a database. Systems differ in the size of the stored speech units; a system that stores phones or diphones provides the largest output range, but may lack clarity. For specific usage domains, the storage of entire words or sentences allows for high-quality output. Alternatively, a synthesizer can incorporate a model of the vocal tract and other human voice characteristics to create a completely "synthetic" voice output.[2]

The quality of a speech synthesizer is judged by its similarity to the human voice and by its ability to be understood clearly. An intelligible text-to-speech program allows people with visual impairments or reading disabilities to listen to written words on a home computer. Many computer operating systems have included speech synthesizers since the early 1990s.



Overview of a typical TTS system

A text-to-speech system (or "engine") is composed of two parts:[3] a front-end and a back-end. The front-end has two major tasks. First, it converts raw text containing symbols like numbers and abbreviations into the equivalent of written-out words. This process is often called *text normalization*, *pre-processing*, or *tokenization*. The front-end then assigns phonetic transcriptions to each word, and divides and marks the text into prosodic units, like phrases, clauses, and sentences. The process of assigning phonetic transcriptions to words is called *text-to-phoneme* or *grapheme-to-phoneme* conversion. Phonetic transcriptions and prosody information together make up the symbolic linguistic representation that is output by the front-end. The back-end—often referred to as the *synthesizer*—then converts the symbolic linguistic representation into sound. In certain systems, this part includes the computation of the *target prosody* (pitch contour, phoneme durations),[4] which is then imposed on the output speech.

Contents

History

Long before the invention of electronic signal processing, some people tried to build machines to emulate human speech. Some early legends of the existence of "Brazen Heads" involved Pope Silvester II (d. 1003 AD), Albertus Magnus (1198–1280), and Roger Bacon (1214–1294).

In 1779 the German-Danish scientist Christian Gottlieb Kratzenstein won the first prize in a competition announced by the Russian Imperial Academy of Sciences and Arts for models he built of the human vocal tract that could produce the five long vowel sounds (in International Phonetic Alphabet notation: [aː], [eː], [iː], [oː] and [uː]).[5] There followed the bellows-operated "acoustic-mechanical speech machine" of Wolfgang von Kempelen of Pressburg, Hungary, described in a 1791 paper.[6] This machine added models of the tongue and lips, enabling it to produce consonants as well as vowels. In 1837, Charles Wheatstone produced a "speaking machine" based on von Kempelen's design, and in 1846, Joseph Faber exhibited the "Euphonia". In 1923 Paget resurrected Wheatstone's design.[7]

In the 1930s Bell Labs developed the vocoder, which automatically analyzed speech into its fundamental tones and resonances. From his work on the vocoder, Homer Dudley developed a keyboard-operated voice-synthesizer called The Voder (Voice Demonstrator), which he exhibited at the 1939 New York World's Fair.

Dr. Franklin S. Cooper and his colleagues at Haskins Laboratories built the Pattern playback in the late 1940s and completed it in 1950. There were several different versions of this hardware device; only one currently survives. The machine converts pictures of the acoustic patterns of speech in the form of a spectrogram back into sound. Using this device, Alvin Liberman and colleagues discovered acoustic cues for the perception of phonetic segments (consonants and vowels).

Electronic devices

The first computer-based speech-synthesis systems originated in the late 1950s. Noriko Umeda *et al.* developed the first general English text-to-speech system in 1968, at the Electrotechnical Laboratory in Japan.[8] In 1961, physicist John Larry Kelly, Jr and his colleague Louis Gerstman[9] used an IBM 704 computer to synthesize speech, an event among the most prominent in the history of Bell Labs. Kelly's voice recorder synthesizer (vocoder) recreated the song "Daisy Bell", with musical accompaniment from Max Mathews. Coincidentally, Arthur C. Clarke was visiting his friend and colleague John Pierce at the Bell Labs Murray Hill facility. Clarke was so impressed by the demonstration that he used it in the climactic scene of his screenplay for his novel *2001: A Space Odyssey*,[10] where the HAL 9000 computer sings the same song as astronaut Dave Bowman puts it to sleep.[11] Despite the success of purely electronic speech synthesis, research into mechanical speech-synthesizers continues.[12]

Computer and speech synthesiser housing used by Stephen Hawking in 1999

Linear predictive coding (LPC), a form of speech coding, began development with the work of Fumitada Itakura of Nagoya University and Shuzo Saito of Nippon Telegraph and Telephone (NTT) in 1966. Further developments in LPC technology were made by Bishnu S. Atal and Manfred R. Schroeder at Bell Labs during the 1970s.[13] LPC was later the basis for early speech synthesizer chips, such as the Texas Instruments LPC Speech Chips used in the Speak & Spell toys from 1978.

In 1975, Fumitada Itakura developed the line spectral pairs (LSP) method for high-compression speech coding, while at NTT.[14][15][16] From 1975 to 1981, Itakura studied problems in speech analysis and synthesis based on the LSP method.[16] In 1980, his team developed an LSP-based speech synthesizer

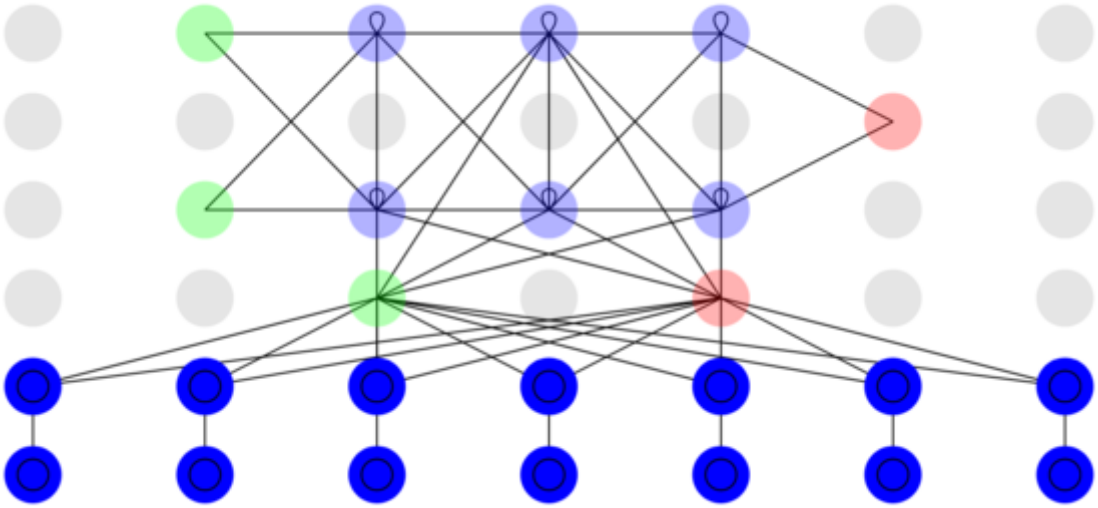
chip. LSP is an important technology for speech synthesis and coding, and in the 1990s was adopted by almost all international speech coding standards as an essential component, contributing to the enhancement of digital speech communication over mobile channels and the internet.[15]

In 1975, MUSA was released, and was one of the first Speech Synthesis systems. It consisted of a stand-alone computer hardware and a specialized software that enabled it to read Italian. A second version, released in 1978, was also able to sing Italian in an "a cappella" style.[17]

Dominant systems in the 1980s and 1990s were the DECtalk system, based largely on the work of Dennis Klatt at MIT, and the Bell Labs system;[18] the latter was one of the first multilingual language-independent systems, making extensive use of natural language processing methods.

0:00 / 0:00

DECtalk demo recording using the Perfect Paul and Uppity Ursula voices



Handheld electronics featuring speech synthesis began emerging in the 1970s. One of the first was the Telesensory Systems Inc. (TSI) *Speech+* portable calculator for the blind in 1976.[19][20] Other devices had primarily educational purposes, such as the Speak & Spell toy produced by Texas Instruments in 1978.[21] Fidelity released a speaking version of its electronic chess computer in 1979.[22] The first video game to feature speech synthesis was the 1980 shoot 'em up arcade game, *Stratovox* (known in Japan as *Speak & Rescue*), from Sun Electronics.[23][24] The first personal computer game with speech synthesis was *Manbiki Shoujo* (*Shoplifting Girl*), released in 1980 for the PET 2001, for which the game's developer, Hiroshi Suzuki, developed a "*zero cross*" programming technique to produce a synthesized speech waveform.[25] Another early example, the arcade version of *Berzerk*, also dates from 1980. The Milton Bradley Company produced the first multi-player electronic game using voice synthesis, *Milton*, in the same year.

Fidelity Voice Chess Challenger (1979), the first talking chess computer

0:00 / 0:00

Speech output from Fidelity Voice Chess Challenger

Early electronic speech-synthesizers sounded robotic and were often barely intelligible. The quality of synthesized speech has steadily improved, but as of 2016 output from contemporary speech synthesis systems remains clearly distinguishable from actual human speech.

Synthesized voices typically sounded male until 1990, when Ann Syrdal, at AT&T Bell Laboratories, created a female voice.[26]

Kurzweil predicted in 2005 that as the cost-performance ratio caused speech synthesizers to become cheaper and more accessible, more people would benefit from the use of text-to-speech programs.[27]

Synthesizer technologies

The most important qualities of a speech synthesis system are *naturalness* and *intelligibility*.[28] Naturalness describes how closely the output sounds like human speech, while intelligibility is the ease with which the output is understood. The ideal speech synthesizer is both natural and intelligible. Speech synthesis systems usually try to maximize both characteristics.

The two primary technologies generating synthetic speech waveforms are *concatenative synthesis* and *formant synthesis*. Each technology has strengths and weaknesses, and the intended uses of a synthesis system will typically determine which approach is used.

Concatenation synthesis

Concatenative synthesis is based on the concatenation (stringing together) of segments of recorded speech. Generally, concatenative synthesis produces the most natural-sounding synthesized speech. However, differences between natural variations in speech and the nature of the automated techniques for segmenting the waveforms sometimes result in audible glitches in the output. There are three main sub-types of concatenative synthesis.

Unit selection synthesis

Unit selection synthesis uses large databases of recorded speech. During database creation, each recorded utterance is segmented into some or all of the following: individual phones, diphones, half-phones, syllables, morphemes, words, phrases, and sentences. Typically, the division into segments is done using a specially modified speech recognizer set to a "forced alignment" mode with some manual correction afterward, using visual representations such as the waveform and spectrogram.[29] An index of the units in the speech database is then created based on the segmentation and acoustic parameters like the fundamental frequency (pitch), duration, position in the syllable, and neighboring phones. At run time, the desired target utterance is created by determining the best chain of candidate units from the database (unit selection). This process is typically achieved using a specially weighted decision tree.

Unit selection provides the greatest naturalness, because it applies only a small amount of digital signal processing (DSP) to the recorded speech. DSP often makes recorded speech sound less natural, although some systems use a small amount of signal processing at the point of concatenation to smooth the waveform. The output from the best unit-selection systems is often indistinguishable from real human voices, especially in contexts for which the TTS system has been tuned. However, maximum naturalness typically require unit-selection speech databases to be very large, in some systems ranging into the gigabytes of recorded data, representing dozens of hours of speech.[30] Also, unit selection algorithms have

been known to select segments from a place that results in less than ideal synthesis (e.g. minor words become unclear) even when a better choice exists in the database.[31] Recently, researchers have proposed various automated methods to detect unnatural segments in unit-selection speech synthesis systems.[32]

Diphone synthesis

Diphone synthesis uses a minimal speech database containing all the diphones (sound-to-sound transitions) occurring in a language. The number of diphones depends on the phonotactics of the language: for example, Spanish has about 800 diphones, and German about 2500. In diphone synthesis, only one example of each diphone is contained in the speech database. At runtime, the target prosody of a sentence is superimposed on these minimal units by means of digital signal processing techniques such as linear predictive coding, PSOLA[33] or MBROLA.[34] or more recent techniques such as pitch modification in the source domain using discrete cosine transform.[35] Diphone synthesis suffers from the sonic glitches of concatenative synthesis and the robotic-sounding nature of formant synthesis, and has few of the advantages of either approach other than small size. As such, its use in commercial applications is declining, although it continues to be used in research because there are a number of freely available software implementations. An early example of Diphone synthesis is a teaching robot, Leachim, that was invented by Michael J. Freeman.[36] Leachim contained information regarding class curricular and certain biographical information about the students whom it was programmed to teach.[37] It was tested in a fourth grade classroom in the Bronx, New York.[38][39]

Domain-specific synthesis

Domain-specific synthesis concatenates prerecorded words and phrases to create complete utterances. It is used in applications where the variety of texts the system will output is limited to a particular domain, like transit schedule announcements or weather reports.[40] The technology is very simple to implement, and has been in commercial use for a long time, in devices like talking clocks and calculators. The level of naturalness of these systems can be very high because the variety of sentence types is limited, and they closely match the prosody and intonation of the original recordings.

Because these systems are limited by the words and phrases in their databases, they are not general-purpose and can only synthesize the combinations of words and phrases with which they have been preprogrammed. The blending of words within naturally spoken language however can still cause problems unless the many variations are taken into account. For example, in non-rhotic dialects of English the *"r"* in words like *"clear"* /ˈklɪə/ is usually only pronounced when the following word has a vowel as its first letter (e.g. *"clear out"* is realized as /ˌklɪəɹˈʌʊt/). Likewise in French, many final consonants become no longer silent if followed by a word that begins with a vowel, an effect called liaison. This alternation cannot be reproduced by a simple word-concatenation system, which would require additional complexity to be context-sensitive.

Formant synthesis

Formant synthesis does not use human speech samples at runtime. Instead, the synthesized speech output is created using additive synthesis and an acoustic model (physical modelling synthesis).[41] Parameters such as fundamental frequency, voicing, and noise levels are varied over time to create a waveform of artificial speech. This method is sometimes called *rules-based synthesis*; however, many concatenative systems also have rules-based components. Many systems based on formant synthesis technology generate artificial, robotic-sounding speech that would never be mistaken for human speech. However, maximum naturalness is not always the goal of a speech synthesis system, and formant synthesis systems have advantages over concatenative systems. Formant-synthesized speech can be reliably intelligible, even at very high speeds,

avoiding the acoustic glitches that commonly plague concatenative systems. High-speed synthesized speech is used by the visually impaired to quickly navigate computers using a screen reader. Formant synthesizers are usually smaller programs than concatenative systems because they do not have a database of speech samples. They can therefore be used in embedded systems, where memory and microprocessor power are especially limited. Because formant-based systems have complete control of all aspects of the output speech, a wide variety of prosodies and intonations can be output, conveying not just questions and statements, but a variety of emotions and tones of voice.

Examples of non-real-time but highly accurate intonation control in formant synthesis include the work done in the late 1970s for the Texas Instruments toy Speak & Spell, and in the early 1980s Sega arcade machines[42] and in many Atari, Inc. arcade games[43] using the TMS5220 LPC Chips. Creating proper intonation for these projects was painstaking, and the results have yet to be matched by real-time text-to-speech interfaces.[44]

Articulatory synthesis

Articulatory synthesis refers to computational techniques for synthesizing speech based on models of the human vocal tract and the articulation processes occurring there. The first articulatory synthesizer regularly used for laboratory experiments was developed at Haskins Laboratories in the mid-1970s by Philip Rubin, Tom Baer, and Paul Mermelstein. This synthesizer, known as ASY, was based on vocal tract models developed at Bell Laboratories in the 1960s and 1970s by Paul Mermelstein, Cecil Coker, and colleagues.

Until recently, articulatory synthesis models have not been incorporated into commercial speech synthesis systems. A notable exception is the NeXT-based system originally developed and marketed by Trillium Sound Research, a spin-off company of the University of Calgary, where much of the original research was conducted. Following the demise of the various incarnations of NeXT (started by Steve Jobs in the late 1980s and merged with Apple Computer in 1997), the Trillium software was published under the GNU General Public License, with work continuing as gnuspeech. The system, first marketed in 1994, provides full articulatory-based text-to-speech conversion using a waveguide or transmission-line analog of the human oral and nasal tracts controlled by Carré's "distinctive region model".

More recent synthesizers, developed by Jorge C. Lucero and colleagues, incorporate models of vocal fold biomechanics, glottal aerodynamics and acoustic wave propagation in the bronqui, traquea, nasal and oral cavities, and thus constitute full systems of physics-based speech simulation.[45][46]

HMM-based synthesis

HMM-based synthesis is a synthesis method based on hidden Markov models, also called Statistical Parametric Synthesis. In this system, the frequency spectrum (vocal tract), fundamental frequency (voice source), and duration (prosody) of speech are modeled simultaneously by HMMs. Speech waveforms are generated from HMMs themselves based on the maximum likelihood criterion.[47]

Sinewave synthesis

Sinewave synthesis is a technique for synthesizing speech by replacing the formants (main bands of energy) with pure tone whistles.[48]

Deep learning-based synthesis

Deep learning speech synthesis uses deep neural networks (DNN) to produce artificial speech from text (text-to-speech) or spectrum (vocoder). The deep neural networks are trained using a large amount of recorded speech and, in the case of a text-to-speech system, the associated labels and/or input text.

The DNN-based speech synthesizers are approaching the naturalness of the human voice. Examples of disadvantages of the method are low robustness when the data are not sufficient, lack of controllability and low performance in auto-regressive models. Some of the limitations (like lack of controllability) can be solved by future research.

Currently, Tacotron2 + Waveglow requires only a few dozen hours of training material on recorded speech to produce a very high quality voice. However, for tonal languages, such as Chinese or Taiwanese language, there are different levels of tone sandhi required and sometimes the output of speech synthesizer may result in the mistakes of tone sandhi.

Audio deepfakes

The audio deepfake is a type of artificial intelligence used to create convincing speech sentences that sound like specific people saying things they did not say.[49][50] This technology was initially developed for various applications to improve human life. For example, it can be used to produce audiobooks,[51] and also to help people who have lost their voices (due to throat disease or other medical problems) to get them back.[52][53] Commercially, it has opened the door to several opportunities. This technology can also create more personalized digital assistants and natural-sounding speech translation services.

Audio deepfakes, recently called audio manipulations, are becoming widely accessible using simple mobile devices or personal PCs.[54] Unfortunately, these tools have also been used to spread misinformation around the world using audio,[50] and their malicious use has led to fears of the audio deepfake. This has led to cybersecurity concerns among the global public about the side effects of using audio deepfakes. People can use them as a logical access voice spoofing technique,[55] where they can be used to manipulate public opinion for propaganda, defamation, or terrorism. Vast amounts of voice recordings are daily transmitted over the Internet, and spoofing detection is challenging.[56] However, audio deepfake attackers have targeted not only individuals and organizations but also politicians and governments.[57] In early 2020, some scammers used artificial intelligence-based software to impersonate the voice of a CEO to authorize a money transfer of about $35 million through a phone call.[58] Therefore, it is necessary to authenticate any audio recording distributed to avoid spreading misinformation.

Challenges

Text normalization challenges

The process of normalizing text is rarely straightforward. Texts are full of heteronyms, numbers, and abbreviations that all require expansion into a phonetic representation. There are many spellings in English which are pronounced differently based on context. For example, "My latest project is to learn how to better project my voice" contains two pronunciations of "project".

Most text-to-speech (TTS) systems do not generate semantic representations of their input texts, as processes for doing so are unreliable, poorly understood, and computationally ineffective. As a result, various heuristic techniques are used to guess the proper way to disambiguate homographs, like examining neighboring words and using statistics about frequency of occurrence.

Recently TTS systems have begun to use HMMs (discussed above) to generate "parts of speech" to aid in disambiguating homographs. This technique is quite successful for many cases such as whether "read" should be pronounced as "red" implying past tense, or as "reed" implying present tense. Typical error rates when using HMMs in this fashion are usually below five percent. These techniques also work well for most European languages, although access to required training corpora is frequently difficult in these languages.

Deciding how to convert numbers is another problem that TTS systems have to address. It is a simple programming challenge to convert a number into words (at least in English), like "1325" becoming "one thousand three hundred twenty-five." However, numbers occur in many different contexts; "1325" may also be read as "one three two five", "thirteen twenty-five" or "thirteen hundred and twenty five". A TTS system can often infer how to expand a number based on surrounding words, numbers, and punctuation, and sometimes the system provides a way to specify the context if it is ambiguous.[59] Roman numerals can also be read differently depending on context. For example, "Henry VIII" reads as "Henry the Eighth", while "Chapter VIII" reads as "Chapter Eight".

Similarly, abbreviations can be ambiguous. For example, the abbreviation "in" for "inches" must be differentiated from the word "in", and the address "12 St John St." uses the same abbreviation for both "Saint" and "Street". TTS systems with intelligent front ends can make educated guesses about ambiguous abbreviations, while others provide the same result in all cases, resulting in nonsensical (and sometimes comical) outputs, such as "Ulysses S. Grant" being rendered as "Ulysses South Grant".

Text-to-phoneme challenges

Speech synthesis systems use two basic approaches to determine the pronunciation of a word based on its spelling, a process which is often called text-to-phoneme or grapheme-to-phoneme conversion (phoneme is the term used by linguists to describe distinctive sounds in a language). The simplest approach to text-to-phoneme conversion is the dictionary-based approach, where a large dictionary containing all the words of a language and their correct pronunciations is stored by the program. Determining the correct pronunciation of each word is a matter of looking up each word in the dictionary and replacing the spelling with the pronunciation specified in the dictionary. The other approach is rule-based, in which pronunciation rules are applied to words to determine their pronunciations based on their spellings. This is similar to the "sounding out", or synthetic phonics, approach to learning reading.

Each approach has advantages and drawbacks. The dictionary-based approach is quick and accurate, but completely fails if it is given a word which is not in its dictionary. As dictionary size grows, so too does the memory space requirements of the synthesis system. On the other hand, the rule-based approach works on any input, but the complexity of the rules grows substantially as the system takes into account irregular spellings or pronunciations. (Consider that the word "of" is very common in English, yet is the only word in which the letter "f" is pronounced [v].) As a result, nearly all speech synthesis systems use a combination of these approaches.

Languages with a phonemic orthography have a very regular writing system, and the prediction of the pronunciation of words based on their spellings is quite successful. Speech synthesis systems for such languages often use the rule-based method extensively, resorting to dictionaries only for those few words, like foreign names and loanwords, whose pronunciations are not obvious from their spellings. On the other hand, speech synthesis systems for languages like English, which have extremely irregular spelling systems, are more likely to rely on dictionaries, and to use rule-based methods only for unusual words, or words that aren't in their dictionaries.

Evaluation challenges

The consistent evaluation of speech synthesis systems may be difficult because of a lack of universally agreed objective evaluation criteria. Different organizations often use different speech data. The quality of speech synthesis systems also depends on the quality of the production technique (which may involve analogue or digital recording) and on the facilities used to replay the speech. Evaluating speech synthesis systems has therefore often been compromised by differences between production techniques and replay facilities.

Since 2005, however, some researchers have started to evaluate speech synthesis systems using a common speech dataset.[60]

Prosodics and emotional content

A study in the journal *Speech Communication* by Amy Drahota and colleagues at the University of Portsmouth, UK, reported that listeners to voice recordings could determine, at better than chance levels, whether or not the speaker was smiling.[61][62][63] It was suggested that identification of the vocal features that signal emotional content may be used to help make synthesized speech sound more natural. One of the related issues is modification of the pitch contour of the sentence, depending upon whether it is an affirmative, interrogative or exclamatory sentence. One of the techniques for pitch modification[64] uses discrete cosine transform in the source domain (linear prediction residual). Such pitch synchronous pitch modification techniques need a priori pitch marking of the synthesis speech database using techniques such as epoch extraction using dynamic plosion index applied on the integrated linear prediction residual of the voiced regions of speech.[65]

Dedicated hardware

- Icophone
- General Instrument SP0256-AL2
- National Semiconductor DT1050 Digitalker (Mozer – Forrest Mozer)
- Texas Instruments LPC Speech Chips[66]

Hardware and software systems

Popular systems offering speech synthesis as a built-in capability.

Mattel

The Mattel Intellivision game console offered the Intellivoice Voice Synthesis module in 1982. It included the SP0256 Narrator speech synthesizer chip on a removable cartridge. The Narrator had 2kB of Read-Only Memory (ROM), and this was utilized to store a database of generic words that could be combined to make phrases in Intellivision games. Since the Orator chip could also accept speech data from external memory, any additional words or phrases needed could be stored inside the cartridge itself. The data consisted of strings of analog-filter coefficients to modify the behavior of the chip's synthetic vocal-tract model, rather than simple digitized samples.

SAM

Also released in 1982, Software Automatic Mouth was the first commercial all-software voice synthesis program. It was later used as the basis for Macintalk. The program was available for non-Macintosh Apple computers (including the Apple II, and the Lisa), various Atari models and the Commodore 64. The Apple version preferred additional hardware that contained DACs, although it could instead use the computer's one-bit audio output (with the addition of much distortion) if the card was not present. The Atari made use of the embedded POKEY audio chip. Speech playback on the Atari normally disabled interrupt requests and shut down the ANTIC chip during vocal output. The audible output is extremely distorted speech when the screen is on. The Commodore 64 made use of the 64's embedded SID audio chip.

0:00 / 0:00

A demo of SAM on the C64

Atari

Arguably, the first speech system integrated into an operating system was the 1400XL/1450XL personal computers designed by Atari, Inc. using the Votrax SC01 chip in 1983. The 1400XL/1450XL computers used a Finite State Machine to enable World English Spelling text-to-speech synthesis.[67] Unfortunately, the 1400XL/1450XL personal computers never shipped in quantity.

The Atari ST computers were sold with "stspeech.tos" on floppy disk.

Apple

The first speech system integrated into an operating system that shipped in quantity was Apple Computer's MacInTalk. The software was licensed from third-party developers Joseph Katz and Mark Barton (later, SoftVoice, Inc.) and was featured during the 1984 introduction of the Macintosh computer. This January demo required 512 kilobytes of RAM memory. As a result, it could not run in the 128 kilobytes of RAM the first Mac actually shipped with.[68] So, the demo was accomplished with a prototype 512k Mac, although those in attendance were not told of this and the synthesis demo created considerable excitement for the Macintosh. In the early 1990s Apple expanded its capabilities offering system wide text-to-speech support. With the introduction of faster PowerPC-based computers they included higher quality voice sampling. Apple also introduced speech recognition into its systems which provided a fluid command set. More recently, Apple has added sample-based voices. Starting as a curiosity, the speech system of Apple Macintosh has evolved into a fully supported program, PlainTalk, for people with vision problems. VoiceOver was for the first time featured in 2005 in Mac OS X Tiger (10.4). During 10.4 (Tiger) and first releases of 10.5 (Leopard) there was only one standard voice shipping with Mac OS X. Starting with 10.6 (Snow Leopard), the user can choose out of a wide range list of multiple voices. VoiceOver voices feature the taking of realistic-sounding breaths between sentences, as well as improved clarity at high read rates over PlainTalk. Mac OS X also includes say, a command-line based application that converts text to audible speech. The AppleScript Standard Additions includes a say verb that allows a script to use any of the installed voices and to control the pitch, speaking rate and modulation of the spoken text.

0:00 / 0:00

MacinTalk 1 demo

0:00 / 0:00

MacinTalk 2 demo featuring the Mr. Hughes and Marvin voices

Amazon

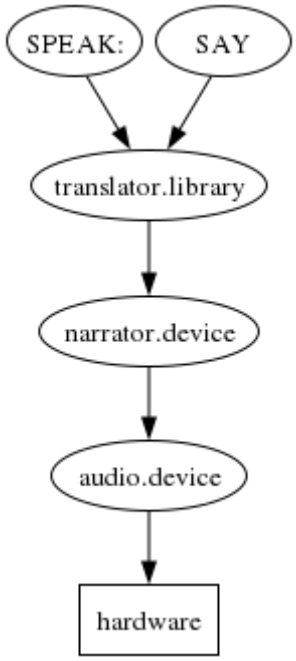
Used in Alexa and as Software as a Service in AWS[69] (from 2017).

AmigaOS

0:00 / 0:00

Example of speech synthesis with the included Say utility in Workbench 1.3

The second operating system to feature advanced speech synthesis capabilities was AmigaOS, introduced in 1985. The voice synthesis was licensed by Commodore International from SoftVoice, Inc., who also developed the original MacinTalk text-to-speech system. It featured a complete system of voice emulation for American English, with both male and female voices and "stress" indicator markers, made possible through the Amiga's audio chipset.[70] The synthesis system was divided into a translator library which converted unrestricted English text into a standard set of phonetic codes and a narrator device which implemented a formant model of speech generation.. AmigaOS also featured a high-level "Speak Handler", which allowed command-line users to redirect text output to speech. Speech synthesis was occasionally used in third-party programs, particularly word processors and educational software. The synthesis software remained largely unchanged from the first AmigaOS release and Commodore eventually removed speech synthesis support from AmigaOS 2.1 onward.



Despite the American English phoneme limitation, an unofficial version with multilingual speech synthesis was developed. This made use of an enhanced version of the translator library which could translate a number of languages, given a set of rules for each language.[71]

Microsoft Windows

Modern Windows desktop systems can use SAPI 4 and SAPI 5 components to support speech synthesis and speech recognition. SAPI 4.0 was available as an optional add-on for Windows 95 and Windows 98. Windows 2000 added Narrator, a text-to-speech utility for people who have visual impairment. Third-party programs such as JAWS for Windows, Window-Eyes, Non-visual Desktop Access, Supernova and System Access can perform various text-to-speech tasks such as reading text aloud from a specified website, email account, text document, the Windows clipboard, the user's keyboard typing, etc. Not all programs can use speech synthesis directly.[72] Some programs can use plug-ins, extensions or add-ons to read text aloud. Third-party programs are available that can read text from the system clipboard.

Microsoft Speech Server is a server-based package for voice synthesis and recognition. It is designed for network use with web applications and call centers.

Texas Instruments TI-99/4A

0:00 / 0:00

TI-99/4A speech demo using the built-in vocabulary

In the early 1980s, TI was known as a pioneer in speech synthesis, and a highly popular plug-in speech synthesizer module was available for the TI-99/4 and 4A. Speech synthesizers were offered free with the purchase of a number of cartridges and were used by many TI-written video games (notable titles offered with speech during this promotion were Alpiner and Parsec). The synthesizer uses a variant of linear predictive coding and has a small in-built vocabulary. The original intent was to release small cartridges that plugged directly into the synthesizer unit, which would increase the device's built-in vocabulary. However, the success of software text-to-speech in the Terminal Emulator II cartridge canceled that plan.

Votrax

From 1971 to 1996, Votrax produced a number of commercial speech synthesizer components. A Votrax synthesizer was included in the first generation Kurzweil Reading Machine for the Blind.

Text-to-speech systems

Text-to-speech (TTS) refers to the ability of computers to read text aloud. A TTS engine converts written text to a phonemic representation, then converts the phonemic representation to waveforms that can be output as sound. TTS engines with different languages, dialects and specialized vocabularies are available through third-party publishers.[73]

Android

Version 1.6 of Android added support for speech synthesis (TTS).[74]

Internet

Currently, there are a number of applications, plugins and gadgets that can read messages directly from an e-mail client and web pages from a web browser or Google Toolbar. Some specialized software can narrate RSS-feeds. On one hand, online RSS-narrators simplify information delivery by allowing users to listen to their favourite news sources and to convert them to podcasts. On the other hand, on-line RSS-readers are available on almost any personal computer connected to the Internet. Users can download generated audio files to portable devices, e.g. with a help of podcast receiver, and listen to them while walking, jogging or commuting to work.

A growing field in Internet based TTS is web-based assistive technology, e.g. 'Browsealoud' from a UK company and Readspeaker. It can deliver TTS functionality to anyone (for reasons of accessibility, convenience, entertainment or information) with access to a web browser. The non-profit project Pediaphon was created in 2006 to provide a similar web-based TTS interface to the Wikipedia.[75]

Other work is being done in the context of the W3C through the W3C Audio Incubator Group with the involvement of The BBC and Google Inc.

Open source

Some open-source software systems are available, such as:

- RHVoice with support for multiple languages.[76][77]
- Festival Speech Synthesis System which uses diphone-based synthesis, as well as more modern and better-sounding techniques.
- eSpeak which supports a broad range of languages.
- gnuspeech which uses articulatory synthesis[78] from the Free Software Foundation.
- MaryTTS, web based and open source.[79]

Others

- Following the commercial failure of the hardware-based Intellivoice, gaming developers sparingly used software synthesis in later games. Earlier systems from Atari, such as the Atari 5200 (Baseball) and the Atari 2600 (Quadrun and Open Sesame), also had games utilizing software synthesis.
- Some e-book readers, such as the Amazon Kindle, Samsung E6, PocketBook eReader Pro, enTourage eDGe, and the Bebook Neo.
- The BBC Micro incorporated the Texas Instruments TMS5220 speech synthesis chip,
- Some models of Texas Instruments home computers produced in 1979 and 1981 (Texas Instruments TI-99/4 and TI-99/4A) were capable of text-to-phoneme synthesis or reciting complete words and phrases (text-to-dictionary), using a very popular Speech Synthesizer peripheral. TI used a proprietary codec to embed complete spoken phrases into applications, primarily video games.[80]
- IBM's OS/2 Warp 4 included VoiceType, a precursor to IBM ViaVoice.
- GPS Navigation units produced by Garmin, Magellan, TomTom and others use speech synthesis for automobile navigation.
- Yamaha produced a music synthesizer in 1999, the Yamaha FS1R which included a Formant synthesis capability. Sequences of up to 512 individual vowel and consonant formants could be stored and replayed, allowing short vocal phrases to be synthesized.

Digital sound-alikes

At the 2018 Conference on Neural Information Processing Systems (NeurIPS) researchers from Google presented the work 'Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis', which transfers learning from speaker verification to achieve text-to-speech synthesis, that can be made to sound almost like anybody from a speech sample of only 5 seconds.[81]

Also researchers from Baidu Research presented a voice cloning system with similar aims at the 2018 NeurIPS conference,[82] though the result is rather unconvincing.

By 2019 the digital sound-alikes found their way to the hands of criminals as Symantec researchers know of 3 cases where digital sound-alikes technology has been used for crime.[83][84]

This increases the stress on the disinformation situation coupled with the facts that

- Human image synthesis since the early 2000s has improved beyond the point of human's inability to tell a real human imaged with a real camera from a simulation of a human imaged with a simulation of a camera.
- 2D video forgery techniques were presented in 2016 that allow near real-time counterfeiting of facial expressions in existing 2D video.[85]
- In SIGGRAPH 2017 an audio driven digital look-alike of upper torso of Barack Obama was presented by researchers from University of Washington. It was driven only by a voice track as source data for the animation after the training phase to acquire lip sync and wider facial information from training material consisting of 2D videos with audio had been completed.[86]

In March 2020, a freeware web application called 15.ai that generates high-quality voices from an assortment of fictional characters from a variety of media sources was released.[87] Initial characters included GLaDOS from *Portal*, Twilight Sparkle and Fluttershy from the show *My Little Pony: Friendship Is Magic*, and the Tenth Doctor from *Doctor Who*.

Speech synthesis markup languages

A number of markup languages have been established for the rendition of text as speech in an XML-compliant format. The most recent is Speech Synthesis Markup Language (SSML), which became a W3C recommendation in 2004. Older speech synthesis markup languages include Java Speech Markup Language (JSML) and SABLE. Although each of these was proposed as a standard, none of them have been widely adopted.

Speech synthesis markup languages are distinguished from dialogue markup languages. VoiceXML, for example, includes tags related to speech recognition, dialogue management and touchtone dialing, in addition to text-to-speech markup.

Applications

Speech synthesis has long been a vital assistive technology tool and its application in this area is significant and widespread. It allows environmental barriers to be removed for people with a wide range of disabilities. The longest application has been in the use of screen readers for people with visual impairment, but text-to-speech systems are now commonly used by people with dyslexia and other reading difficulties as well as by pre-literate children. They are also frequently employed to aid those with severe speech impairment usually through a dedicated voice output communication aid. A noted application, of speech synthesis, was the Kurzweil Reading Machine for the Blind which incorporated text-to-phonetics software based on work from Haskins Laboratories and a black-box synthesizer built by Votrax[88]

Speech synthesis techniques are also used in entertainment productions such as games and animations. In 2007, Animo Limited announced the development of a software application package based on its speech synthesis software FineSpeech, explicitly geared towards customers in the entertainment industries, able to generate narration and lines of dialogue according to user specifications.[89] The application reached maturity in 2008, when NEC Biglobe announced a web service that allows users to create phrases from the voices of characters from the Japanese anime series *Code Geass: Lelouch of the Rebellion R2*.[90]

In recent years, text-to-speech for disability and impaired communication aids have become widely available. Text-to-speech is also finding new applications; for example, speech synthesis combined with speech recognition allows for interaction with mobile devices via natural language processing interfaces.

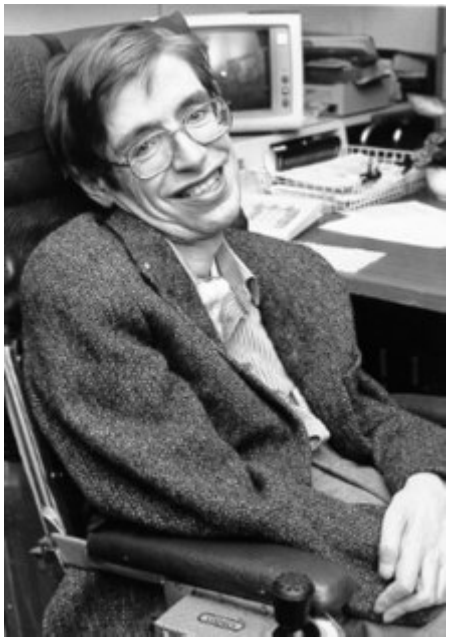
Text-to-speech is also used in second language acquisition. Voki, for instance, is an educational tool created by Oddcast that allows users to create their own talking avatar, using different accents. They can be emailed, embedded on websites or shared on social media. Another area of application is AI video creation with talking heads. Tools, like Elai.io are allowing users to create video content with AI avatars[91] who speak using text-to-speech technology.

In addition, speech synthesis is a valuable computational aid for the analysis and assessment of speech disorders. A voice quality synthesizer, developed by Jorge C. Lucero et al. at the University of Brasília, simulates the physics of phonation and includes models of vocal frequency jitter and tremor, airflow noise and laryngeal asymmetries.[45] The synthesizer has been used to mimic the timbre of dysphonic speakers with controlled levels of roughness, breathiness and strain.[46]

See also

- 15.ai
- Chinese speech synthesis

- Comparison of screen readers
- Comparison of speech synthesizers
- Euphonia (device)
- Orca (assistive technology)
- Paperless office
- Speech processing
- Speech-generating device
- Silent speech interface
- Text to speech in digital television



Stephen Hawking was one of the most famous people to use a speech computer to communicate.

References

1. Allen, Jonathan; Hunnicutt, M. Sharon; Klatt, Dennis (1987). *From Text to Speech: The MITalk system* (https://archive.org/details/fromtexttospeech00alle). Cambridge University Press. ISBN 978-0-521-30641-6.
2. Rubin, P.; Baer, T.; Mermelstein, P. (1981). "An articulatory synthesizer for perceptual research". *Journal of the Acoustical Society of America*. **70** (2): 321–328. Bibcode:1981ASAJ...70..321R (https://ui.adsabs.harvard.edu/abs/1981ASAJ...70..321R). doi:10.1121/1.386780 (https://doi.org/10.1121%2F1.386780).
3. van Santen, Jan P. H.; Sproat, Richard W.; Olive, Joseph P.; Hirschberg, Julia (1997). *Progress in Speech Synthesis* (https://archive.org/details/progressinspeech0000unse). Springer. ISBN 978-0-387-94701-3.
4. Van Santen, J. (April 1994). "Assignment of segmental duration in text-to-speech synthesis". *Computer Speech & Language*. **8** (2): 95–128. doi:10.1006/csla.1994.1005 (https://doi.org/10.1006%2Fcsla.1994.1005).
5. History and Development of Speech Synthesis (http://www.acoustics.hut.fi/publications/files/theses/lemmetty_mst/chap2.html), Helsinki University of Technology, Retrieved on November 4, 2006
6. *Mechanismus der menschlichen Sprache nebst der Beschreibung seiner sprechenden Maschine* ("Mechanism of the human speech with description of its speaking machine", J. B. Degen, Wien). (in German)
7. Mattingly, Ignatius G. (1974). Sebeok, Thomas A. (ed.). "Speech synthesis for phonetic and phonological models" (https://web.archive.org/web/20130512085755/http://www.haskins.yale.edu/Reprints/HL0173.pdf) (PDF). *Current Trends in Linguistics*. Mouton, The Hague. **12**: 2451–2487. Archived from the original (http://www.haskins.yale.edu/Reprints/HL0173.pdf) (PDF) on 2013-05-12. Retrieved 2011-12-13.

8. Klatt, D (1987). "Review of text-to-speech conversion for English". *Journal of the Acoustical Society of America*. **82** (3): 737–93. Bibcode:1987ASAJ...82..737K (https://ui.adsabs.harvard.edu/abs/1987ASAJ...82..737K). doi:10.1121/1.395275 (https://doi.org/10.1121%2F1.395275). PMID 2958525 (https://pubmed.ncbi.nlm.nih.gov/2958525).
9. Lambert, Bruce (March 21, 1992). "Louis Gerstman, 61, a Specialist In Speech Disorders and Processes" (https://www.nytimes.com/1992/03/21/nyregion/louis-gerstman-61-a-specialist-in-speech-disorders-and-processes.html). *The New York Times*.
10. "Arthur C. Clarke Biography" (https://web.archive.org/web/19971211154551/http://www.lsi.usp.br/~rbianchi/clarke/ACC.Biography.html). Archived from the original (http://www.lsi.usp.br/~rbianchi/clarke/ACC.Biography.html) on December 11, 1997. Retrieved 5 December 2017.
11. "Where "HAL" First Spoke (Bell Labs Speech Synthesis website)" (https://web.archive.org/web/20000407081031/http://www.bell-labs.com/news/1997/march/5/2.html). Bell Labs. Archived from the original (http://www.bell-labs.com/news/1997/march/5/2.html) on 2000-04-07. Retrieved 2010-02-17.
12. Anthropomorphic Talking Robot Waseda-Talker Series (http://www.takanishi.mech.waseda.ac.jp/top/research/voice/index.htm) Archived (https://web.archive.org/web/20160304034116/http://www.takanishi.mech.waseda.ac.jp/top/research/voice/index.htm) 2016-03-04 at the Wayback Machine
13. Gray, Robert M. (2010). "A History of Realtime Digital Speech on Packet Networks: Part II of Linear Predictive Coding and the Internet Protocol" (https://ee.stanford.edu/~gray/lpcip.pdf) (PDF). *Found. Trends Signal Process*. **3** (4): 203–303. doi:10.1561/2000000036 (https://doi.org/10.1561%2F2000000036). ISSN 1932-8346 (https://www.worldcat.org/issn/1932-8346).
14. Zheng, F.; Song, Z.; Li, L.; Yu, W. (1998). "The Distance Measure for Line Spectrum Pairs Applied to Speech Recognition" (http://www.work.caltech.edu/~ling/pub/icslp98lsp.pdf) (PDF). *Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP'98)* (3): 1123–6.
15. "List of IEEE Milestones" (https://ethw.org/Milestones:List_of_IEEE_Milestones). IEEE. Retrieved 15 July 2019.
16. "Fumitada Itakura Oral History" (https://ethw.org/Oral-History:Fumitada_Itakura). IEEE Global History Network. 20 May 2009. Retrieved 2009-07-21.

17. Billi, Roberto; Canavesio, Franco; Ciaramella, Alberto; Nebbia, Luciano (1 November 1995). "Interactive voice technology at work: The CSELT experience". *Speech Communication*. **17** (3): 263–271. doi:10.1016/0167-6393(95)00030-R (https://doi.org/10.1016%2F0167-6393%2895%2900030-R).
18. Sproat, Richard W. (1997). *Multilingual Text-to-Speech Synthesis: The Bell Labs Approach*. Springer. ISBN 978-0-7923-8027-6.
19. [TSI Speech+ & other speaking calculators]
20. Gevaryahu, Jonathan, ["TSI S14001A Speech Synthesizer LSI Integrated Circuit Guide"]
21. Breslow, et al. US 4326710 (https://worldwide.espacenet.com/textdoc?DB=EPODOC&IDX=US4326710): "Talking electronic game", April 27, 1982
22. Voice Chess Challenger (http://www.ismenio.com/chess_fidelity_vcc.html)
23. Gaming's most important evolutions (http://www.gamesradar.com/f/gamings-most-important-evolutions/a-20101008102331322035/p-2) Archived (https://web.archive.org/web/20110615221800/http://www.gamesradar.com/f/gamings-most-important-evolutions/a-20101008102331322035/p-2) 2011-06-15 at the Wayback Machine, GamesRadar
24. Adlum, Eddie (November 1985). "The Replay Years: Reflections from Eddie Adlum" (https://archive.org/details/re-play-volume-11-issue-no.-2-november-1985-600DPI/RePlay%20-%20Volume%2011%2C%20Issue%20No.%202%20-%20November%201985/page/162/mode/2up). *RePlay*. Vol. 11, no. 2. pp. 134-175 (160-3).
25. Szczepaniak, John (2014). *The Untold History of Japanese Game Developers*. Vol. 1. SMG Szczepaniak. pp. 544–615. ISBN 978-0992926007.
26. CadeMetz (2020-08-20). "Ann Syrdal, Who Helped Give Computers a Female Voice, Dies at 74" (https://www.nytimes.com/2020/08/20/technology/ann-syrdal-who-helped-give-computers-a-female-voice-dies-at-74.html). *The New York Times*. Retrieved 2020-08-23.
27. Kurzweil, Raymond (2005). *The Singularity is Near*. Penguin Books. ISBN 978-0-14-303788-0.
28. Taylor, Paul (2009). *Text-to-speech synthesis* (https://archive.org/details/texttospeechsynt00tayl_030). Cambridge, UK: Cambridge University Press. p. 3 (https://archive.org/details/texttospeechsynt00tayl_030/page/n26). ISBN 9780521899277.
29. Alan W. Black, Perfect synthesis for all of the people all of the time. (https://www.cs.cmu.edu/~awb/papers/IEEE2002/allthetime/allthetime.html) IEEE TTS Workshop 2002.

30. John Kominek and Alan W. Black. (2003). CMU ARCTIC databases for speech synthesis. CMU-LTI-03-177. Language Technologies Institute, School of Computer Science, Carnegie Mellon University.
31. Julia Zhang. Language Generation and Speech Synthesis in Dialogues for Language Learning (http://groups.csail.mit.edu/sls/publications/2004/zhang_thesis.pdf), masters thesis, Section 5.6 on page 54.
32. William Yang Wang and Kallirroi Georgila. (2011). Automatic Detection of Unnatural Word-Level Segments in Unit-Selection Speech Synthesis (https://www.cs.cmu.edu/~yww/papers/asru2011.pdf), IEEE ASRU 2011.
33. "Pitch-Synchronous Overlap and Add (PSOLA) Synthesis" (https://web.archive.org/web/20070222180903/http://www.fon.hum.uva.nl/praat/manual/PSOLA.html). Archived from the original (http://www.fon.hum.uva.nl/praat/manual/PSOLA.html) on February 22, 2007. Retrieved 2008-05-28.
34. T. Dutoit, V. Pagel, N. Pierret, F. Bataille, O. van der Vrecken. The MBROLA Project: Towards a set of high quality speech synthesizers of use for non commercial purposes (http://ai2-s2-pdfs.s3.amazonaws.com/7b1f/dadf05b8f968a5b361f6f82852ade62c8010.pdf). *ICSLP Proceedings*, 1996.
35. Muralishankar, R; Ramakrishnan, A.G.; Prathibha, P (2004). "Modification of Pitch using DCT in the Source Domain". *Speech Communication*. **42** (2): 143–154. doi:10.1016/j.specom.2003.05.001 (https://doi.org/10.1016%2Fj.specom.2003.05.001).
36. "Education: Marvel of The Bronx" (http://content.time.com/time/magazine/article/0,9171,904056,00.html). *Time*. 1974-04-01. ISSN 0040-781X (https://www.worldcat.org/issn/0040-781X). Retrieved 2019-05-28.
37. "1960 - Rudy the Robot - Michael Freeman (American)" (http://cyberneticzoo.com/robots/1960-rudy-the-robot-michael-freeman-american/). *cyberneticzoo.com*. 2010-09-13. Retrieved 2019-05-23.
38. LLC, New York Media (1979-07-30). *New York Magazine* (https://books.google.com/books?id=bNECAAAMBAJ&q=Leachim+Michael+Freeman&pg=PA40). New York Media, LLC.
39. *The Futurist* (https://books.google.com/books?id=_QJmAAAAMAAJ&q=leachim). World Future Society. 1978. pp. 359, 360, 361.
40. L.F. Lamel, J.L. Gauvain, B. Prouts, C. Bouhier, R. Boesch. Generation and Synthesis of Broadcast Messages (http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.53.6101&rep=rep1&type=pdf), *Proceedings ESCA-NATO Workshop and Applications of Speech Technology*, September 1993.

41. Dartmouth College: *Music and Computers* (http://digitalmusics.dartmouth.edu/~book/MATCpages/chap.4/4.4.formant_synth.html) Archived (https://web.archive.org/web/20110608035309/http://digitalmusics.dartmouth.edu/~book/MATCpages/chap.4/4.4.formant_synth.html) 2011-06-08 at the Wayback Machine, 1993.
42. Examples include Astro Blaster, Space Fury, and Star Trek: Strategic Operations Simulator
43. Examples include Star Wars, Firefox, Return of the Jedi, Road Runner, The Empire Strikes Back, Indiana Jones and the Temple of Doom, 720°, Gauntlet, Gauntlet II, A.P.B., Paperboy, RoadBlasters, Vindicators Part II (http://www.arcade-museum.com/game_detail.php?game_id=10319), Escape from the Planet of the Robot Monsters.
44. John Holmes and Wendy Holmes (2001). *Speech Synthesis and Recognition* (2nd ed.). CRC. ISBN 978-0-7484-0856-6.
45. Lucero, J. C.; Schoentgen, J.; Behlau, M. (2013). "Physics-based synthesis of disordered voices" (http://www.cic.unb.br/~lucero/papers/768_Paper.pdf) (PDF). *Interspeech 2013*. Lyon, France: International Speech Communication Association: 587–591. doi:10.21437/Interspeech.2013-161 (https://doi.org/10.21437%2FInterspeech.2013-161). Retrieved Aug 27, 2015.
46. Englert, Marina; Madazio, Glaucya; Gielow, Ingrid; Lucero, Jorge; Behlau, Mara (2016). "Perceptual error identification of human and synthesized voices". *Journal of Voice*. **30** (5): 639.e17–639.e23. doi:10.1016/j.jvoice.2015.07.017 (https://doi.org/10.1016%2Fj.jvoice.2015.07.017). PMID 26337775 (https://pubmed.ncbi.nlm.nih.gov/26337775).
47. "The HMM-based Speech Synthesis System" (http://hts.sp.nitech.ac.jp/). Hts.sp.nitech.ac.j. Retrieved 2012-02-22.
48. Remez, R.; Rubin, P.; Pisoni, D.; Carrell, T. (22 May 1981). "Speech perception without traditional speech cues" (https://web.archive.org/web/20111216113028/http://www.bsos.umd.edu/hesp/mwinn/Remez_et_al_1981.pdf) (PDF). *Science*. **212** (4497): 947–949. Bibcode:1981Sci...212..947R (https://ui.adsabs.harvard.edu/abs/1981Sci...212..947R). doi:10.1126/science.7233191 (https://doi.org/10.1126%2Fscience.7233191). PMID 7233191 (https://pubmed.ncbi.nlm.nih.gov/7233191). Archived from the original (http://www.bsos.umd.edu/hesp/mwinn/Remez_et_al_1981.pdf) (PDF) on 2011-12-16. Retrieved 2011-12-14.

49. Lyu, Siwei (2020). "Deepfake Detection: Current Challenges and Next Steps" (https://ieeexplore.ieee.org/document/9105991). *2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. pp. 1–6. arXiv:2003.09234 (https://arxiv.org/abs/2003.09234). doi:10.1109/icmew46912.2020.9105991 (https://doi.org/10.1109%2Ficmew46912.2020.9105991). ISBN 978-1-7281-1485-9. S2CID 214605906 (https://api.semanticscholar.org/CorpusID:214605906). Retrieved 2022-06-29.
50. Diakopoulos, Nicholas; Johnson, Deborah (June 2020). "Anticipating and addressing the ethical implications of deepfakes in the context of elections" (http://journals.sagepub.com/doi/10.1177/1461444820925811). *New Media & Society* (published 2020-06-05). **23** (7): 2072–2098. doi:10.1177/1461444820925811 (https://doi.org/10.1177%2F1461444820925811). ISSN 1461-4448 (https://www.worldcat.org/issn/1461-4448). S2CID 226196422 (https://api.semanticscholar.org/CorpusID:226196422).
51. Chadha, Anupama; Kumar, Vaibhav; Kashyap, Sonu; Gupta, Mayank (2021), Singh, Pradeep Kumar; Wierzchoń, Sławomir T.; Tanwar, Sudeep; Ganzha, Maria (eds.), "Deepfake: An Overview" (https://link.springer.com/10.1007/978-981-16-0733-2_39), *Proceedings of Second International Conference on Computing, Communications, and Cyber-Security*, Singapore: Springer Singapore, vol. 203, pp. 557–566, doi:10.1007/978-981-16-0733-2_39 (https://doi.org/10.1007%2F978-981-16-0733-2_39), ISBN 978-981-16-0732-5, S2CID 236666289 (https://api.semanticscholar.org/CorpusID:236666289), retrieved 2022-06-29
52. "AI gave Val Kilmer his voice back. But critics worry the technology could be misused" (https://www.washingtonpost.com/technology/2021/08/18/val-kilmer-ai-voice-cloning/). *Washington Post*. ISSN 0190-8286 (https://www.worldcat.org/issn/0190-8286). Retrieved 2022-06-29.
53. Etienne, Vanessa (August 19, 2021). "Val Kilmer Gets His Voice Back After Throat Cancer Battle Using AI Technology: Hear the Results" (https://people.com/movies/val-kilmer-gets-his-voice-back-after-throat-cancer-battle-using-ai-technology-hear-the-results/). *PEOPLE.com*. Retrieved 2022-07-01.
54. Almutairi, Zaynab; Elgibreen, Hebah (2022-05-04). "A Review of Modern Audio Deepfake Detection Methods: Challenges and Future Directions" (https://doi.org/10.3390%2Fa15050155). *Algorithms*. **15** (5): 155. doi:10.3390/a15050155 (https://doi.org/10.3390%2Fa15050155). ISSN 1999-4893 (https://www.worldcat.org/issn/1999-4893).

55. Chen, Tianxiang; Kumar, Avrosh; Nagarsheth, Parav; Sivaraman, Ganesh; Khoury, Elie (2020-11-01). "Generalization of Audio Deepfake Detection" (https://www.isca-speech.org/archive/odyssey_2020/chen20_odyssey.html). *The Speaker and Language Recognition Workshop (Odyssey 2020)*. ISCA: 132–137. doi:10.21437/Odyssey.2020-19 (https://doi.org/10.21437%2FOdyssey.2020-19). S2CID 219492826 (https://api.semanticscholar.org/CorpusID:219492826).
56. Ballesteros, Dora M.; Rodriguez-Ortega, Yohanna; Renza, Diego; Arce, Gonzalo (2021-12-01). "Deep4SNet: deep learning for fake speech classification" (https://www.sciencedirect.com/science/article/pii/S0957417421008770). *Expert Systems with Applications*. **184**: 115465. doi:10.1016/j.eswa.2021.115465 (https://doi.org/10.1016%2Fj.eswa.2021.115465). ISSN 0957-4174 (https://www.worldcat.org/issn/0957-4174). S2CID 237659479 (https://api.semanticscholar.org/CorpusID:237659479).
57. Suwajanakorn, Supasorn; Seitz, Steven M.; Kemelmacher-Shlizerman, Ira (2017-07-20). "Synthesizing Obama: learning lip sync from audio" (https://doi.org/10.1145/3072959.3073640). *ACM Transactions on Graphics*. **36** (4): 95:1–95:13. doi:10.1145/3072959.3073640 (https://doi.org/10.1145%2F3072959.3073640). ISSN 0730-0301 (https://www.worldcat.org/issn/0730-0301). S2CID 207586187 (https://api.semanticscholar.org/CorpusID:207586187).
58. Brewster, Thomas. "Fraudsters Cloned Company Director's Voice In $35 Million Bank Heist, Police Find" (https://www.forbes.com/sites/thomasbrewster/2021/10/14/huge-bank-fraud-uses-deep-fake-voice-tech-to-steal-millions/). *Forbes*. Retrieved 2022-06-29.
59. "Speech synthesis" (http://www.w3.org/TR/speech-synthesis/#S3.1.8). World Wide Web Organization.
60. "Blizzard Challenge" (http://festvox.org/blizzard). Festvox.org. Retrieved 2012-02-22.
61. "Smile -and the world can hear you" (https://web.archive.org/web/20080517102201/http://www.port.ac.uk/aboutus/newsandevents/news/title%2C74220%2Cen.html). University of Portsmouth. January 9, 2008. Archived from the original (http://www.port.ac.uk/aboutus/newsandevents/news/title,74220,en.html) on May 17, 2008.
62. "Smile – And The World Can Hear You, Even If You Hide" (https://www.sciencedaily.com/releases/2008/01/080111224745.htm). *Science Daily*. January 2008.

63. Drahota, A. (2008). "The vocal communication of different kinds of smile" (https://web.archive.org/web/20130703062330/https://peer.ccsd.cnrs.fr/docs/00/49/91/97/PDF/PEER_stage2_10.1016/j.specom.2007.10.001.pdf) (PDF). *Speech Communication*. **50** (4): 278–287. doi:10.1016/j.specom.2007.10.001 (https://doi.org/10.1016%2Fj.specom.2007.10.001). S2CID 46693018 (https://api.semanticscholar.org/CorpusID:46693018). Archived from the original (http://peer.ccsd.cnrs.fr/docs/00/49/91/97/PDF/PEER_stage2_10.1016%252Fj.specom.2007.10.001.pdf) (PDF) on 2013-07-03.
64. Muralishankar, R.; Ramakrishnan, A. G.; Prathibha, P. (February 2004). "Modification of pitch using DCT in the source domain". *Speech Communication*. **42** (2): 143–154. doi:10.1016/j.specom.2003.05.001 (https://doi.org/10.1016%2Fj.specom.2003.05.001).
65. Prathosh, A. P.; Ramakrishnan, A. G.; Ananthapadmanabha, T. V. (December 2013). "Epoch extraction based on integrated linear prediction residual using plosion index". *IEEE Trans. Audio Speech Language Processing*. **21** (12): 2471–2480. doi:10.1109/TASL.2013.2273717 (https://doi.org/10.1109%2FTASL.2013.2273717). S2CID 10491251 (https://api.semanticscholar.org/CorpusID:10491251).
66. EE Times. "TI will exit dedicated speech-synthesis chips, transfer products to Sensory (http://www.eetimes.com/electronics-news/4102385/TI-will-exit-dedicated-speech-synthesis-chips-transfer-products-to-Sensory) Archived (https://web.archive.org/web/20120528014257/http://www.eetimes.com/electronics-news/4102385/TI-will-exit-dedicated-speech-synthesis-chips-transfer-products-to-Sensory) 2012-05-28 at the Wayback Machine." June 14, 2001.
67. "1400XL/1450XL Speech Handler External Reference Specification" (https://web.archive.org/web/20120324014644/http://www.atarimuseum.com/ahs_archives/archives/pdf/computers/8bits/1400xlmodem.pdf) (PDF). Archived from the original (http://www.atarimuseum.com/ahs_archives/archives/pdf/computers/8bits/1400xlmodem.pdf) (PDF) on 2012-03-24. Retrieved 2012-02-22.
68. "It Sure Is Great To Get Out Of That Bag!" (http://www.folklore.org/StoryView.py?story=Intro_Demo.txt). folklore.org. Retrieved 2013-03-24.
69. "Amazon Polly" (https://aws.amazon.com/polly/). *Amazon Web Services, Inc*. Retrieved 2020-04-28.
70. Miner, Jay; et al. (1991). *Amiga Hardware Reference Manual* (3rd ed.). Addison-Wesley Publishing Company, Inc. ISBN 978-0-201-56776-2.

71. Devitt, Francesco (30 June 1995). "Translator Library (Multilingual-speech version)" (https://web.archive.org/web/20120226143859/https://uk.aminet.net/util/libs/translator42.readme). Archived from the original (http://uk.aminet.net/util/libs/translator42.readme) on 26 February 2012. Retrieved 9 April 2013.
72. "Accessibility Tutorials for Windows XP: Using Narrator" (https://web.archive.org/web/20030621002716/http://www.microsoft.com/enable/training/windowsxp/usingnarrator.aspx). Microsoft. 2011-01-29. Archived from the original (http://www.microsoft.com/enable/training/windowsxp/usingnarrator.aspx) on June 21, 2003. Retrieved 2011-01-29.
73. "How to configure and use Text-to-Speech in Windows XP and in Windows Vista" (http://support.microsoft.com/kb/306902). Microsoft. 2007-05-07. Retrieved 2010-02-17.
74. Jean-Michel Trivi (2009-09-23). "An introduction to Text-To-Speech in Android" (http://android-developers.blogspot.com/2009/09/introduction-to-text-to-speech-in.html). Android-developers.blogspot.com. Retrieved 2010-02-17.
75. Andreas Bischoff, The Pediaphon – Speech Interface to the free Wikipedia Encyclopedia for Mobile Phones (http://www.dr-bischoff.de/research/pdf/bischoff_pediaphon_uwsi2007_final.pdf), PDA's and MP3-Players, Proceedings of the 18th International Conference on Database and Expert Systems Applications, Pages: 575–579 ISBN 0-7695-2932-1, 2007
76. "RHVoice.org" (https://rhvoice.org/). *rhvoice.org*. Retrieved 2022-03-27.
77. "Languages | RHVoice.org" (https://rhvoice.org/languages/). *rhvoice.org*. Retrieved 2022-03-27.
78. "gnuspeech" (https://www.gnu.org/software/gnuspeech/). Gnu.org. Retrieved 2010-02-17.
79. "The MARY Text-to-Speech System (MaryTTS)" (http://mary.dfki.de/). *mary.dfki.de*. Retrieved 2021-11-19.
80. "Smithsonian Speech Synthesis History Project (SSSHP) 1986–2002" (https://web.archive.org/web/20131003104852/http://amhistory.si.edu/archives/speechsynthesis/ss_home.htm). Mindspring.com. Archived from the original (http://www.mindspring.com/~ssshp/ssshp_cd/ss_home.htm) on 2013-10-03. Retrieved 2010-02-17.
81. Jia, Ye; Zhang, Yu; Weiss, Ron J. (2018-06-12), "Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis", *Advances in Neural Information Processing Systems*, **31**: 4485–4495, arXiv:1806.04558 (https://arxiv.org/abs/1806.04558)

82. Arık, Sercan Ö.; Chen, Jitong; Peng, Kainan; Ping, Wei; Zhou, Yanqi (2018), "Neural Voice Cloning with a Few Samples" (http://papers.nips.cc/paper/8206-neural-voice-cloning-with-a-few-samples), *Advances in Neural Information Processing Systems*, **31**, arXiv:1802.06006 (https://arxiv.org/abs/1802.06006)
83. "Fake voices 'help cyber-crooks steal cash' " (https://www.bbc.com/news/technology-48908736). *bbc.com*. BBC. 2019-07-08. Retrieved 2019-09-11.
84. Drew, Harwell (2019-09-04). "An artificial-intelligence first: Voice-mimicking software reportedly used in a major theft" (https://www.washingtonpost.com/technology/2019/09/04/an-artificial-intelligence-first-voice-mimicking-software-reportedly-used-major-theft/). *Washington Post*. Retrieved 2019-09-08.
85. Thies, Justus (2016). "Face2Face: Real-time Face Capture and Reenactment of RGB Videos" (http://www.graphics.stanford.edu/~niessner/thies2016face.html). Proc. Computer Vision and Pattern Recognition (CVPR), IEEE. Retrieved 2016-06-18.
86. Suwajanakorn, Supasorn; Seitz, Steven; Kemelmacher-Shlizerman, Ira (2017), *Synthesizing Obama: Learning Lip Sync from Audio* (http://grail.cs.washington.edu/projects/AudioToObama/), University of Washington, retrieved 2018-03-02
87. Ng, Andrew (2020-04-01). "Voice Cloning for the Masses" (https://web.archive.org/web/20200807111844/https://blog.deeplearning.ai/blog/the-batch-ai-against-coronavirus-datasets-voice-cloning-for-the-masses-finding-unexploded-bombs-seeing-see-through-objects-optimizing-training-parameters). *deeplearning.ai*. The Batch. Archived from the original (https://blog.deeplearning.ai/blog/the-batch-ai-against-coronavirus-datasets-voice-cloning-for-the-masses-finding-unexploded-bombs-seeing-see-through-objects-optimizing-training-parameters) on 2020-08-07. Retrieved 2020-04-02.
88. https://www.rehab.research.va.gov/jour/84/21/1/pdf/cooper.pdf
89. "Speech Synthesis Software for Anime Announced" (http://www.animenewsnetwork.com/news/2007-05-02/speech-synthesis-software). *Anime News Network*. 2007-05-02. Retrieved 2010-02-17.
90. "Code Geass Speech Synthesizer Service Offered in Japan" (http://www.animenewsnetwork.com/news/2008-09-09/code-geass-voice-synthesis-service-offered-in-japan). Animenewsnetwork.com. 2008-09-09. Retrieved 2010-02-17.
91. "Usage of text-to-speech in AI video generation" (https://elai.io/). *Elai.io*. Retrieved 10 August 2022.

External links

- Media related to Speech synthesis at Wikimedia Commons
- Speech synthesis (https://curlie.org/Computers/Speech_Technology/Speech_Synthesis/) at Curlie
- Simulated singing with the singing robot Pavarobotti (https://www.youtube.com/watch?v=CE6zy8aUwtQ) or a description from the BBC on how the robot synthesized the singing (https://www.youtube.com/watch?v=SNqNM6Ccck8).
