

VOICE INTERFACE FOR CONSUMER PRODUCTS

FIELD AND BACKGROUND OF THE INVENTION

The present invention relates to consumer appliances and, more particularly, to a
5 voice interface to improve the human interaction with consumer appliances.

Modern man is inundated with machines and appliances of all kinds during daily life. The user interface in most appliances commonly includes buttons, dials or keypads. However, a simpler, more natural and oftentimes more convenient interface between man and machine is human speech. Thus there is a need for and it would be advantageous to have
10 a voice interface for home appliances and consumer electronic products.

Speech recognition has been developing over the past decades and various methodologies have been introduced for automated speech recognition (ASR), including constrained grammar recognition and natural language recognition.

Speech recognition technology is used for telephone applications like travel booking
15 and information, financial account information, customer service call routing, and directory assistance. Using constrained grammar recognition, such applications can achieve high accuracy. Speech recognition systems optimized for telephone applications can often supply information about the confidence of a particular recognition, and if the confidence is low, the system triggers the application to prompt callers to confirm or repeat their request (for
20 example "I heard you say 'billing', is that right?").

Grammar constrained recognition constrains the possible recognized phrases to a small or medium-sized formal grammar of possible responses, which is typically defined using a grammar specification language. This type of recognition works best when the speaker is providing short responses to specific questions, like yes-no questions; picking an
25 option from a menu; selecting an item from a well-defined list, such as financial securities

like stocks and mutual funds or names of airports; or reading a sequence of numbers or letters, like an account number.

The grammar specifies the most likely words and phrases a person will say in response to a prompt and then maps those words and phrases to a token, or a semantic
5 concept. For example, a yes-no grammar might map "yes", "yeah", "uh-huh", "sure", and "okay" to the token "yes" and "no", "nope", "nuh-uh", and "no way dude!" to the token "no". A grammar for entering a 10-digit account number would have ten slots each of which contain one digit which could be zero through nine, and result from the grammar would be the 10-digit number that was spoken. If the speaker says something that doesn't match an
10 entry in the grammar, recognition will fail. Typically, if recognition fails, the application will re-prompt users to repeat what they said, and recognition will be tried again. If a telephone answering system using grammar constrained recognition is well designed and is repeatedly unable to understand the user (typically due to the caller misunderstanding the question, having a thick accent, mumbling, or speaking over a large amount of background noise or
15 interference), the telephone answering system should be backed up by another input method or transfer the call to an operator. Callers who are asked to repeat themselves over and over quickly become frustrated and agitated.

Natural language recognition allows the speaker to provide natural, sentence-length responses to specific questions. Natural language recognition uses statistical models. The
20 general procedure is to store a large number of typical responses, with each response matched up to a token or concept. For example for the concept "forward my call to the billing department", you would want to recognize sentences like "I have a problem with my bill", "I was charged incorrectly", "How much do I owe this month", etc. It is difficult to create large, rich grammars that consider the context in which the words are said. In addition, as a
25 grammar gets very large, the chances of having similar sounding words in the grammar greatly increases.

Some systems use a hybrid of constrained grammar and natural language recognition that permits sentence-length responses to specific questions, but ignores the irrelevant part of the sentence using a natural language "garbage model". Combining this approach with prompts that encourage short answers can be effective at maximizing the accuracy and
5 correctness of recognition.

Speech recognition is performed by inputting a speech signal, typically using a microphone and digitizing the signal. The speech signal is input into a circuit including a processor which performs a Fast Fourier transform (FFT) using any of the known FFT algorithms. Practically, the input digitized voiced signal in the time domain is placed in an
10 input data buffer. The FFT algorithm and the processing is simplified if performed "out-of-place" i.e. if an output buffer is distinct from the input buffer. For example, the Stockham auto-sort algorithm (Stockham, 1966) performs every stage of the FFT out-of-place, typically writing back and forth between two arrays, transposing one "digit" of the indices with each stage.

15 An "in-place" FFT algorithm uses the same data buffer for the input data and the output (frequency domain) data. A typical strategy for "in-place" algorithms without auxiliary storage and without separate digit-reversal passes involves small matrix transpositions (which swap individual pairs of digits) at intermediate stages, which can be combined with the radix butterflies to reduce the number of passes over the data (Johnson &
20 Burrus, 1984; Temperton, 1991; Qian et al., 1994; Hegland, 1994).

After performing FFT, the frequency domain data is generally filtered *e.g.* Mel filtering to correspond to the way human speech is perceived. A sequence of coefficients are used to generate voice prints of words or phonemes based on Hidden Markov Models (HMMs). A hidden Markov model (HMM) is a statistical model where the system being
25 modeled is assumed to be a Markov process with unknown parameters, and the challenge is to determine the hidden parameters, from the observable parameters, based on this

assumption. The extracted model parameters can then be used to perform speech recognition. Having a model which gives the probability of an observed sequence of acoustic data given a word phoneme or word sequence enables working out the most likely word sequence.

5 References

Rabiner, Lawrence, Biing-Hwang Juang Fundamentals of Speech Recognition, Prentice-Hall
James W. Cooley and John W. Tukey, "An algorithm for the machine calculation of complex Fourier series," Math. Comput. 19, 297–301 (1965).

.0

T. G. Stockham, "High speed convolution and correlation", Spring Joint Computer Conference, Proc. AFIPS 28, 229–233 (1966).

H. W. Johnson and C. S. Burrus, "An in-place in-order radix-2 FFT," Proc. ICASSP,
15 28A.2.1–28A.2.4 (1984).

C. Temperton, "Self-sorting in-place fast Fourier transform," SIAM J. Sci. Stat. Comput. 12
(4), 808–823 (1991).

20 Qian, C. Lu, M. An, and R. Tolimieri, "Self-sorting in-place FFT algorithm with minimum working space," IEEE Trans. ASSP 52 (10), 2835–2836 (1994).

M. Hegland, "A self-sorting in-place fast Fourier transform algorithm suitable for vector and parallel processing," Numerische Mathematik 68 (4), 507–547 (1994).

25

Matteo Frigo and Steven G. Johnson: FFTW, <http://www.fftw.org/>. A free (GPL) C library for computing discrete Fourier transforms in one or more dimensions, of arbitrary size, using the Cooley-Tukey algorithm. Also M. Frigo and S. G. Johnson, "

5 All references are hereby incorporated as if entirely set forth herein.

Background benefits from:

http://en.wikipedia.org/wiki/Speech_recognition, http://en.wikipedia.org/wiki/Cooley-Tukey_FFT_algorithm

10 SUMMARY OF THE INVENTION

The term "programmable device" as used herein refers to a microprocessor or a dedicated device manufactured using any technology such as ASIC, FPGA or CPLD. The terms "microprocessor" and "programmable device" are used herein interchangeably.

The term "open words" as used herein refers to a set of words recognizable during a
15 specific stage during a speech recognition scenario.

The term "in-place FFT algorithm" as used herein refers to any mixed radix or real mixed radix algorithm.

The term "generic" as used herein means that any voice interface application can be applied to multiple programmable devices (or device family) typically by integrating
20 appropriate libraries available in the application development kit of the present invention.

The terms "manufacturer" and "developer" are used herein interchangeably and refers to the entity that develops an appliance for manufacturing the appliance.

The term "manufacturer independent" refers to a property of the application kit of the present invention, that voice interface applications may be developed for multiple types of
25 appliances and/or multiple manufacturers of the same type of appliance.

According to the present invention there is provided a method for generating voice interface for appliances which may be performed by a manufacturer of the appliance. The manufacturer is provided with a programmable device for controlling the appliance, the programmable device having resources of less than 9 kilobytes of random access memory and capable of less than 41 million instructions per second. The manufacturer is further provided with an application development kit for building an application for the voice interface including a speech recognition module. The manufacturer programs the programmable device with the application. When the application is run, such as by a user of the appliance, the application operates the appliance. Preferably, while programming and running the application, the application includes multiple stages and for each stage a different set of open words are recognizable by the speech recognition module. Preferably, the open words are recognized by the speech recognition module, solely in response to a previously stored question posed to a user of the appliance. Preferably, the speech recognition module uses supervised recognition algorithms. Preferably, while running the application, a speech recognition calculation begins on-the-fly, as soon as speech of a user is detected. Preferably, resources of the programmable device include less than 5 kilobytes of random access memory. Preferably, the speech recognition module includes an in-place algorithm for computing a fast Fourier transform. Preferably, programming is performed using assembly code optimized for speed. Preferably, the programmable device is selected by the manufacturer from multiple different programmable device families. Preferably, code is portable between a plurality of programmable device families.

According to the present invention there is provided a voice interface application development kit provided to a manufacturer of a consumer appliance for integrating a voice interface for the consumer appliance. The development kit includes an application generator which receives as inputs from the manufacturer multiple stages of a voice interface application, and for each stage a question is posed to a user of the appliance and a limited

number of open words are recognizable in response to said question. The kit further includes a data base of words from which the open words are selected by the manufacturer, the data base further includes models for recognizing the words. The manufacturer selects a programmable device from programmable device families and builds a voice interface circuit included in the appliance by programming the programmable device with code which implements an application generated with the application generator. Preferably, the number of open words is less than twenty, limited by resources of the programmable device. Preferably, a portion of the code is generic and supported by all the programmable device families. The kit further includes a voice output module which poses questions to the user by controlling part of the voice interface circuit. Preferably, the kit further includes a speech recognition module which applies an in-place fast Fourier transform algorithm to voice input data received in the appliance. Preferably, the speech recognition module applies supervised recognition algorithms.

According to the present invention there is provided a program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform a method for building a voice interface circuit which controls an appliance. The method is performed by a manufacturer of the appliance. The method includes programming a programmable device in multiple stages of a voice interface application. For each stage a question is posed to a user of the appliance and a limited number of open words are recognizable in response to the question. The programmable device included in the voice interface circuit, is selected from a plurality of programmable device families and a portion of the programming is generic to all programmable device families and a portion of the programming is specific to the family of the programmable device. Preferably, the programmable device includes resources of less than 9 kilobytes of random access memory and capable of less than 41 million instructions per second. wherein

said programmable device includes resources of less than 5 kilobytes of random access memory and capable of less than 21 million instructions per second.

BRIEF DESCRIPTION OF THE DRAWINGS

5 The invention is herein described, by way of example only, with reference to the accompanying drawings, wherein:

FIG. 1 is a prior art drawing of a system and method for providing a voice interface in a consumer appliance, according to an embodiment of the present invention;

FIG. 2 is a drawing of the software modules in an application development kit,
10 according to an embodiment of the present invention; and

FIG. 3 is a flow drawing showing stages of a voice interface application, according to an embodiment of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

15 The present invention is of a system and method of creating a voice interface for home appliances. By way of introduction, consider the ubiquitous clock-radio or alarm clock used to arouse us in the morning. The standard interface to the clock-radio includes several buttons. Typically, both time and alarm are set by cycling through digits by pushing one or more of the buttons. This process is repeated to set hours and minutes. The alarm time is
20 similarly set. Those who do not have a regular schedule, for instance, are required to repeat the process of re-setting the alarm time several times during the week. The standard clock-radio interface is not generally convenient for individuals who share bedrooms when each individual has a different schedule. Improved clock/radio/alarm interfaces can be purchased, however, given a plethora of buttons many people only learn the most basic functions.

25 As a consumer appliance, the clock/radio/alarm is very cost sensitive. Each additional feature requires an additional button adding cost and size to the unit. Alternatively, the

feature may be multiplexed with existing buttons adding additional complexity to the human interface.

A principal intention of the present invention is to provide a voice interface to home appliances, such as a clock-radio. The voice interface, according to an embodiment of the present invention is implemented with minimal additional cost, on the order of a few
5 dollars by requiring minimal computing and data storage resources such as is available using a small 16 bit microprocessor, (for instance TMS320LF2401A with 2 kilobytes of random access memory and 40Mhz (Texas Instruments Inc., 12500 TI Boulevard, Dallas, TX) or equivalent processor, i.e. ASIC capable of 20 million instructions per second (MIPS). It
10 should be noted that the programmable device used in embodiments of the present invention has much less resources than processors used in prior-art speech recognition systems, *e.g.* telephone answering systems.

Another intention of the present invention is to provide software tools to manufacturers of home appliances for building a voice interface for the appliance. Each
15 manufacturer of consumer appliances may build a voice interface with use of the application kit, of the present invention, according to his own requirements.

Another intention of the present invention is to provide performance, typically speed and recognition accuracy sufficient that the voice interface is convenient to use, Performance is measured both by speed and by accuracy of speech recognition. Speed required for a
20 correct recognition of a response is on the order of 1-2 secs, accuracy of recognition is preferably greater than 95%. Preferably, the performance is sufficient so that a conventional interface is not required as a backup, reducing the overall cost of the consumer appliance.

The principles and operation of a system and method of creating a voice interface for consumer appliances, according to the present invention, may be better understood with
25 reference to the drawings and the accompanying description.

Before explaining embodiments of the invention in detail, it is to be understood that the invention is not limited in its application to the details of design and the arrangement of the components set forth in the following description or illustrated in the drawings. The invention is capable of other embodiments or of being practiced or carried out in various
5 ways. Also, it is to be understood that the phraseology and terminology employed herein is for the purpose of description and should not be regarded as limiting.

It should be noted that while the discussion herein is directed to small consumer appliances, the principles of the present invention may be adapted for large appliances, *e.g.* automobiles or use in non-consumer applications as well.

10 Further the speech recognition algorithm, such as the algorithm for performing a fast Fourier transform may be any such algorithm known in the art.

Referring now to the drawings, Figure 1 illustrates a system and method for providing a voice interface in a consumer appliance **101**. Typically, a manufacturer/developer of appliance **101** builds a voice interface circuit **103** as part of consumer appliance **101** under
15 development or an emulation circuit (not shown) which emulates the function of consumer appliance **101** being developed. Voice interface circuit includes a programmable device, *e.g.* microprocessor **115** with a cable **109** to a personal computer **111** which programs the microprocessor with a voice interface application. Microprocessor **115** has a connection **105** through appropriate circuitry to a microphone and a speaker cable **107** to a speaker (not
20 shown) for voice output from appliance **101**. Program storage device, *e.g.* CDROM **113** is used to load a voice interface application development kit into personal computer **111** for the purpose of building the voice interface for appliance **101**.

Figure 2 illustrates a block diagram of software modules included in voice interface application development kit **20**, according to the present invention.

25 Application development kit **20** includes an application generator or scenario creator **201**. Application generator **201** is used to generate a series of questions which will be posed

to users of appliance **101**. Application generator **20** is used to define a set of open words which are valid user responses to the questions posed. When the voice interface is run, models of the open words are stored in random access memory attached to or packaged with microprocessor **115**. The number of open words is limited to about ten or 20, when using
5 small programmable device **115**, depending on the speed or accuracy required for speech recognition. Preferably, an attempt to use too many open words during any stage of the application results in application generator **201** to generate a warning or error message to the development person (manufacturer) operating application generator **201**. Commonly used words are typically provided with voice interface application development kit **20** in a
10 recorded words database **205**. Alternatively or in addition, the manufacturer of the appliance may record his own words and build his own recorded word database **205**. Application generator **201** is preferably written a generic language, typically ANSI C so that many microprocessor families **115** are supported. The manufacturer/developer may choose microprocessor **115** typically one already used by the manufacturer or already integrated into
15 appliance **101**.

Speech recognition module **203** reads the voice data and performs fast Fourier transforms with a butterfly/permutation process and compares the output data, *e.g.* Mel Frequency Cepstrum coefficients, to the models of open words stored in RAM memory.

Since RAM memory is limited, *e.g.* to 2K words (or 4K bytes), fast Fourier transforms (FFT)
20 (and inverse) transforms are preferably performed using an in-place algorithm, storing for instance time domain input voice data and the output frequency domain data in the same array. Since only a real FFT is required, a 256 word data buffer is sufficient if an in-place algorithm is used. The use of an in-place algorithm includes a penalty in calculation time. In order to increase calculation speed, speech recognition module **203** is written in assembler
25 code optimized for speed. Since assembler code is not typically generic and each programmable device **115** family has its own instruction set, assembler code libraries **207** are

included in application development kit **20** to support multiple families of programmable devices **115**. In order to further increase speed of calculation, speed recognition calculations are performed "on-the-fly" triggered by the onset of voice reception and do not wait for the word to be fully spoken and received.

5 Voice interface application development kit **20** further includes a voice output module which records the questions generated by application generator **201** and plays the questions on the speaker of appliance **101** through speaker connection **107**. Voice interface development kit further includes an option to record documentation **211** for the manufacturer and/or a user of appliance **101**.

0 Figure 3 illustrates a voice interface scenario **30** for a DVD recorder remote control unit. Voice interface scenario **30** may be generated by a manufacturer while developing a voice interface for the DVD recorder control unit. Typically, scenario **30** begins with a listening step **301** as a background process. A person is prompted to speak one or more names name which refer to the control unit, for instance to wake-up DVD from sleep mode.
5 He speaks the name "CHARLEY". Speech recognition module **203** builds a model of the received name "CHARLEY" and calculates a model and places the model of "CHARLEY" in FLASH memory, attached to programmable device **115**. Subsequently, on power up the model of "CHARLEY" is loaded in RAM. As scenario **30** proceeds, the person is prompted to enter an opening question and/or response by the control unit. The person chooses the
20 word "HELLO" from recorded words database **205**.

 Scenario **30** continues with listening step **301b**, and enters two open words {RECORD, SET} from recorded words database **205**. The word "RECORD" if received is used to initiate a recording, the word "SET" is used to set a parameter in the control unit or DVD recorder. In response to "RECORD", the control unit is programmed to respond (step
25 **303b**) "WHICH DAY?". In listening step **301c**, open words which are valid spoken responses are:

{SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, TODAY, TOMORROW}

At each stage of scenario **30**, the open words relevant to the stage are loaded from ROM and/or FLASH memory to RAM connected to programmable device **115**.

- 5 Scenario **30** continues straightforwardly (not shown in Figure 3), for instance the control unit asks by playing through the speaker:

“WHAT TIME TO BEGIN?”

Typically, for this question there are many possible responses. For hours, open words are for instance “ONE” through “TWELVE” with “AM” and “PM” Recognizing minutes includes
10 additional possibilities for instance for 16:30 a user may respond “FOUR THIRTY” or “HALF PAST FOUR”.

At each stage of scenario **30**, a limited number of open words is used in order to speed up and facilitate the speech recognition performance without requiring excessive computing resources. According to an embodiment of the present invention, since the
15 manufacturer/developer controls the application he/she can predict the type and order of words the user is expected to say. Alternatively, application generator **201** includes a dedicated module for handling time of day response. For example in scenario **30** when asking: “WHAT TIME TO BEGIN” the the user response is predicted based on the following assumptions:

- 20 The first word is usually a number, so open words including models of “one” to “twelve” are placed in memory. The second word (if there is one) can be “fifteen”, “thirty”, “forty- five”, “AM”, “PM”, “O'clock”. The third word can be AM, PM.

If a number is not, recognized and the user response is perceived as “garbage” and the program loads the word “half” as an open word. If the first word is “Half” then the second
25 word must be “past”. Third word again must be a number from “one” to “twelve”.

By supplying the manufacturer/developer with an application kit, he/she who controls the application can always divide the application into a larger number of stages, maintaining a small number (less than ten or twenty) of open words. Since the number of possibilities is very small, a dedicated recognition algorithm (*e.g.* a supervised Viterbi algorithm) may be
5 used since the recognition algorithm is dedicated to the specific application and scenario.

By limiting the number of open words at every stage of the speech recognition scenario, the manufacturer/developer empowered with an application generator according to the present invention, can integrate a voice interface using a programmable device of minimal resources and maintain a low cost bill of materials for the consumer appliance.
10 Since we can predict the sentence structure, the first word is a number between “one” and “twelve” and the second word is “AM or “PM” etc. then a special recognition algorithm (*e.g.* a supervised recognition algorithm) is dedicated for this structure type. The supervised recognition algorithm allows downsizing the number of possibilities and by that achieve greater accuracy. In addition, building a supervised recognition algorithm creates more
15 accuracy to the recognition process.

While the invention has been described with respect to a limited number of embodiments, it will be appreciated that many variations, modifications and other applications of the invention may be made.

WHAT IS CLAIMED IS

1. A method for enabling an appliance manufacturer to develop a specific voice interface for a specific appliance, the method comprising the steps of:

(a) providing a generic, manufacturer independent application development kit for use on embedded devices, the kit capable of developing different embedded voice interface applications for different appliances manufactured by same or different appliance manufacturers;

(b) providing an embedded programmable device for controlling the specific appliance; and

(c) enabling the appliance manufacturer to program the embedded programmable device with a specific voice interface application developed using the generic application development kit, thereby providing an embedded voice interface which allows the specific appliance to be operated by voice inputs,

2. The method of claim 1, wherein the specific appliance includes a consumer appliance.

3. The method of claim 1, wherein the specific appliance includes any appliance other than large appliances such as automobiles.

4. The method of claim 2, wherein the consumer appliance is an automobile.

5. A method for enabling an appliance manufacturer to develop a specific embedded voice interface for a specific appliance, the method comprising the steps of:

(a) providing an embedded programmable device for controlling a respective specific appliance;

(b) providing a generic, manufacturer independent application development kit to the appliance manufacturer for building a specific application for the specific embedded voice interface, wherein the kit includes assembler code and assembler code libraries;

(c) programming the programmable device with the specific application; and

(d) running the specific application, thereby operating the specific appliance.

6. The method of claim 5, wherein the step of programming is performed by the appliance manufacturer.

7. The method of claim 5, wherein the appliance manufacturer manufactures a plurality of different appliances requiring different embedded voice interfaces, wherein the step of providing a generic manufacturer independent application development kit includes providing a single application kit for building different applications for respective different types of appliances and wherein the step of programming includes programming a respective embedded programmable device of each specific appliance with a respective application.

8. The method of claim 5, wherein the specific appliance includes a consumer appliance.

9. The method of claim 5, wherein the specific appliance includes any appliance other than large appliances such as automobiles.

10. The method of claim 8, wherein the consumer appliance is an automobile.

11. A system for enabling an appliance manufacturer to develop a specific voice interface for a respective specific appliance, comprising:

(a) a generic, manufacturer independent application development kit for use on embedded devices, the kit including assembler code and assembler code libraries and capable of developing different embedded voice interface applications for different appliances; and

(b) an embedded programmable device capable of being programmed with a specific embedded voice interface application developed using the generic application development kit;

whereby an embedded voice interface developed by the programmable device allows the specific appliance to be operated by voice inputs.

12. The system of claim 11, wherein the generic, manufacturer independent application development kit includes:

i) an application generator used to generate a series of questions to be posed to the specific appliance and to define a set of open words which are valid user responses to the series of questions,

ii) a recorded database having commonly used words stored therein,

iii) a speech recognition module for recognizing the user responses, and

iv) a voice output module for recording the questions generated by the application generator and for playing the generated questions on a speaker.

13. The system of claim 11, wherein the embedded programmable device is selectable from a plurality of programmable device families.

14. The system of claim 11, wherein the specific appliance is selected from the group consisting of a consumer appliance and any appliance other than large appliances such as automobiles.

15. The method of claim 14, wherein the consumer appliance is an automobile.

16. The method of claim 1 or 5, wherein said programmable device having resources of less than 9 kilobytes of random access memory and capable of less than 41 million instructions per second.

17. The system of claim 11, wherein said programmable device having resources of less than 9 kilobytes of random access memory and capable of less than 41 million instructions per second.