

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMAZON.COM, INC.,
Petitioner

IPR2025-00625
U.S. Patent No. 8,930,196

**DECLARATION OF CHRISTOPHER M. SCHMANDT,
UNDER 37 C.F.R. § 1.68 IN SUPPORT OF PETITION
FOR *INTER PARTES* REVIEW**

TABLE OF CONTENTS

I.	INTRODUCTION	4
II.	QUALIFICATIONS AND PROFESSIONAL EXPERIENCE	7
III.	LEVEL OF ORDINARY SKILL IN THE ART	12
IV.	RELEVANT LEGAL STANDARDS	13
V.	THE '196 PATENT	15
A.	Technology Overview	15
1.	Speech Recognition.....	15
2.	Feature Vectors	20
3.	Acoustic Models	24
4.	Hidden Markov Models	26
B.	Overview of the '196 Patent.....	36
VI.	CLAIM CONSTRUCTION	39
VII.	IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE....	40
A.	Ground 1: Claims 1-5, 8-9, 16-17, and 19-20 are obvious under §103 over Bou-Ghazale in view of Konopka.....	41
1.	Summary of Bou-Ghazale.....	41
2.	Summary of Konopka	45
3.	Reasons to Combine Bou-Ghazale and Konopka.....	46
4.	Detailed Analysis of the Claims	52
B.	Ground 2: Claims 6-7 are obvious under §103 over Bou-Ghazale in view of Konopka and further in view of Lee.	151
1.	Summary of Lee.....	151
2.	Reasons to Combine Bou-Ghazale and Konopka with Lee ...	151
3.	Detailed Analysis of the Claims	155
C.	Ground 3: Claims 10-13, and 18 are obvious under §103 over Bou- Ghazale in view of Konopka and further in view of Takiguchi.	167

1.	Summary of Takiguchi	167
2.	Reasons to Combine Bou-Ghazale and Konopka with Takiguchi	167
3.	Detailed Analysis of the Claims	172
D.	Ground 4: Claims 14-15 are obvious under §103 over Bou-Ghazale in view of Konopka and Takiguchi and further in view of Lee.....	179
VIII.	CONCLUSION.....	181

I, Christopher Schmandt, do hereby declare as follows:

I. INTRODUCTION

1. I am making this declaration at the request of Amazon.com, Inc. in the matter of the *Inter Partes* Review of U.S. Patent No. 8,930,196 (“the ’196 patent”) to Jung et al.

2. I am being compensated for my work in this matter at my standard hourly rate. I am also being reimbursed for reasonable and customary expenses associated with my work and testimony in this proceeding. My compensation is not contingent on the outcome of this matter or the specifics of my testimony.

3. I have been asked to provide my opinions regarding whether the subject matter of claims 1-20 (“the Challenged Claims”) of the ’196 patent would have been obvious to a person having ordinary skill in the art (“POSITA”) at the time of the alleged invention, in light of the prior art. It is my opinion that the Challenged Claims would have been obvious to a POSITA.

4. In the preparation of this declaration, I have studied:

- a. the ’196 patent, Ex.1001;
- b. the prosecution history of the ’196 patent (“’196 File History”), Ex.1002;
- c. S. E. Bou-Ghazale et al., "Hands-free voice activation of personal communication devices," *2000 IEEE International Conference on*

Acoustics, Speech, and Signal Processing. Proceedings (Cat.

No.00CH37100), Istanbul, Turkey, 2000, pp. 1735-1738 vol.3, doi:

10.1109/ICASSP.2000.862087. (“Bou-Ghazale”), Ex.1005;

d. U.S. Patent Application Pub. 2008/0059188 to Konopka et al.
 (“Konopka”), Ex.1006;

e. Lee et al., “Competing models-based text-prompted speech
independent verification algorithm,” *Speech communication*, 48(1), 28-44.
 (“Lee”), Ex.1007;

f. U.S. Patent Application Pub. 2006/0122832 to Takiguchi et al.
 (“Takiguchi”), Ex.1008;

g. Definition of “MIPS (million instructions per second),”
Waterfields Guide to Computer Terms, Waterfields Press, 1995, Ex.1012;

h. Definition of “Million Instructions Per Second (MIPS),”
Specialist Computing’s Computer Jargon Dictionary and Thesaurus, 2nd
Edition, Specialist Computing Limited, 2004, Ex.1013;

i. Definition of “MIPS,” The Facts on File Dictionary of
Computer Science, Revised Edition, Facts on File, 2006, Ex.1014;

j. U.K. Patent Application Pub. 2409750 to Hogenhout, et al.,
Ex.1015; and

k. S.J. Young, N.H. Russell, and J.H.S. Thorton, “Token Passing:
a Simple Conceptual Model for Connected Speech Recognition Systems,”
Cambridge University Engineering Department, 1989, Ex.1016.

5. In forming the opinions expressed below, I have considered: the documents listed above; the relevant legal standards, including the standard for obviousness; and my own knowledge and experience based upon my work in the field of speech recognition as described below, as well as portions of the following additional materials:

- a. C.H. Lee et al., “Acoustic modeling for large vocabulary speech recognition,” *Computer and Speech Language*, (1990) 4, 127-165, Ex.1017;
- b. C. Schmandt, *Voice Communication with Computers*, Van Nostrand Reinhold, 1994, Ex.1018;
- c. John Holmes and Wendy Holmes, *Speech Synthesis and Recognition*, 2nd Edition, Taylor & Francis, 2001, Ex.1019;
- d. L.R. Rabiner, “A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition,” *Proceedings of the IEEE*, Vol. 77, No. 2, February 1989 (“Rabiner 89”), Ex.1021;
- e. L. Rabiner et al., *Fundamentals of Speech Recognition*, Prentice Hall PTR, 1993 (“Rabiner”), Ex.1022;

- f. U.S. Patent No. 4,926,488 to Nadas et al. (“Nadas”), Ex.1023;
- g. U.S. Patent Application Pub. 2003/0220796 to Aoyama et al.
 (“Aoyama”), Ex.1024;
- h. U.S. Patent No. 6,092,045 to Stubley et al. (“Stubley”), Ex.1025;
- i. U.S. Patent No. 6,151,574 to Lee et al. (“Lee ’574”), Ex.1026;
- j. U.S. Patent No. 6,452,348 to Toyoda (“Toyoda”), Ex.1027; and
- k. any other document cited below.

6. Unless otherwise noted, all **emphasis** in any quoted material has been added. Claim terms are *italicized*. Citations to non-patent document exhibits are to original page numbering in the exhibits unless the Bates-stamped page numbers are specifically referenced in the format “Page X of X.”

II. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE

7. My complete qualifications and professional experience are described in my *Curriculum Vitae*, a copy of which can be found in Exhibit 1004. The following is a brief summary of my relevant qualifications and professional experience.

8. I retired six years ago after a 40-year career at the Massachusetts Institute of Technology (“MIT”); for most of that time I was employed as a Principal Research Scientist at the Media Laboratory. In that role, I also served as

faculty for the MIT Media Arts and Sciences academic program. I was a founder of the Media Laboratory, a research lab which now spans two buildings.

9. I received my B.S. degree in Electrical Engineering and Computer Science from MIT in 1978, and my M.S. in Visual Studies (Computer Graphics) from MIT the following year. I was employed at MIT since 1980, initially at the Architecture Machine Group which was an early computer graphics and interactive systems research lab. In 1985, I helped found the Media Laboratory and continued to work there until retirement. I was director of a research group titled “Living Mobile.” My research spanned distributed communication and collaborative systems, with an emphasis on multi-media and user interfaces, with a strong focus on speech-based systems and portable devices. I have over 70 published conference and journal papers and one book in the field of speech technology—including HMM-based speech recognition—and user interaction.

10. For 40 years, my research has heavily focused on speech recognition and speech-based user interfaces. I was a pioneer in developing voice-based user interfaces, including “Put That There” in 1980 and “Intelligent Ear” in 1981. Put That There was one of the first speech recognition systems employing multi-modal input (speech and pointing) and had conversational ability (hearing speech input and responding with synthetic speech output). Put That There used a “wake word” or “voice trigger” to enable speech commands in environments with other

extraneous speech and non-speech. Intelligent Ear was one of the first graphical user interfaces for editing documents with audio utilizing keyword recognition. I also built one of the earliest unified messaging systems (“Phone Slave”, 1983) which combined text messages (e.g., email) and telephone-based voice messages, and used speech recognition over the phone for remote access and a graphical user interface for local access. I also applied speech recognition to extend information retrieval techniques to recorded radio newscasts (“Synthetic News Radio”, 1999). With my students, we built speech recognition-based systems for various mobile, wearable, or portable systems, including: “Nomadic Radio” which utilized the wake word “Nomadic” to activate a neck-worn radio (1997); “ComMotion” (1999) for a wearable system that utilized headphones with speech recognition capability; and “Impromptu” (2001) for a pocket-sized mobile computer. We also built several other speech-based interfaces for taking notes, managing answering machines, and more. (“Voicenotes” 1993, “Chatter” 1994, “TalkBack” 2003).

11. In addition to employing speech recognition as an interaction technique, my work explored novel uses of digitized audio in speech recognition, including voice messages, “live” environmental audio, and “always on” recording techniques. For example, “NewsComm” (1996) used automatic speaker detection and segmentation techniques to enable rapid skimming of what we refer to today as podcasts. “The Audio Notebook” (1997) allowed for selective retrieval and

skimming of recorded audio lectures using pitch and time-based prosodic cues to detect portions of the recording that the talker emphasized and topic transitions.

12. We also built several audio recognition systems that detect contextual events in environmental or ambient audio and selectively present them to users. “Listen In” (2003) performed near real time processing of audio from a distributed network of microphones in a domestic environment to detect activities of daily life. A similar concept extended this work to “Monkey Business” (2006) which used audio cues to detect activity in an office environment. This networked system would listen for novel auditory events and then transmit relevant segments between members of a distributed work group. Another version of this system captured and transmitted auditory activity between a home and office environment. “Impromptu” (2001) used two sound awareness applications: a baby monitor application listened for acoustic signatures of a crying baby and would then play a notification sound and open a two-way voice over IP channel to a wireless device for streaming playback; a burglar alarm application listened for and recognized a triggering sound, played a notification sound (e.g., a barking dog) in response, transmitted the triggering sound for playback, then switched to two-way audio streaming.

13. In 2007, I was awarded the prestigious Association for Computing Machinery (“ACM”) Computer Human Interface (“CHI”) Academy membership specifically for those years of work pioneering speech user interfaces.

14. My work also explored speech-recognition distributed systems. Because of my early work building distributed speech systems, I served for several years in the mid-1990s with a working group on the impact of multimedia systems on the Internet reporting to the Internet Engineering Task Force (“IETF”) and later the Internet Activities Board (“IAB”). This work impacted emerging standards such as Session Initiation Protocol (“SIP”).

15. In my faculty position I taught graduate level courses in speech technology and user interaction design, and directly supervised student research and theses at the Bachelors, Masters, and PhD level. I oversaw the Masters and PhD thesis programs for the entire Media Arts and Sciences academic program during my more senior years. I also served on the Media Laboratory intellectual property committee for many years, including as the chair.

16. I have reviewed the ’196 patent, and relevant excerpts of the prosecution history of the ’196 patent. Based on my experience and education, I believe that I am qualified to offer opinions as to the knowledge and level of skill of one of ordinary skill in the art at the time of the invention of the ’196 patent (which I further describe below).

III. LEVEL OF ORDINARY SKILL IN THE ART

17. I understand there are multiple factors relevant to determining the level of ordinary skill in the pertinent art, including (1) the levels of education and experience of persons working in the field at the time of the invention; (2) the sophistication of the technology; (3) the types of problems encountered in the field; and (4) the prior art solutions to those problems.

18. I have been informed that the earliest possible priority date for the '196 patent is May 28, 2008.

19. A POSITA in the field of the '196 patent, as of its earliest possible priority date in May 2008, would have been someone knowledgeable and familiar with speech recognition arts that are pertinent to the '196 patent. That person would have at least a bachelor's degree in computer engineering, computer science, or equivalent training, and approximately two years of experience working in the field of speech recognition. Lack of work experience can be remedied by additional education, and vice versa.

20. For purposes of this Declaration, in general, and unless otherwise noted, my statements and opinions, such as those regarding my experience and what a POSITA would have understood or known generally (and specifically related to the references I consulted herein), reflect the knowledge that existed in the relevant field as of the earliest possible priority date of the '196 patent (i.e.,

May 28, 2008). Unless otherwise stated, when I provide my understanding and analysis below, it is consistent with the level of a POSITA prior to the earliest possible priority date of the '196 patent.

IV. RELEVANT LEGAL STANDARDS

21. I am not an attorney. In preparing and expressing my opinions and considering the subject matter of the '196 patent, I am relying on certain basic legal principles that Amazon's counsel has explained to me. These principles are discussed below.

22. I understand that prior art to the '196 patent includes patents and printed publications in the relevant art that predate the priority date of the '196 patent.

23. I have been informed by Amazon's counsel that a claimed invention is unpatentable under 35 U.S.C. § 103 if the differences between the claimed invention and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a POSITA. I have also been informed by Amazon's counsel that the obviousness analysis considers factual inquiries, including the level of ordinary skill in the art, the scope and content of the prior art, and the differences between the prior art and the claimed subject matter.

24. I have been further informed by Amazon's counsel that there are several recognized rationales for combining references or modifying a reference to show obviousness of claimed subject matter. These rationales include: (a) combining prior art elements according to known methods to yield predictable results; (b) simple substitution of one known element for another to obtain predictable results; (c) use of a known technique to improve a similar device (method, or product) in the same way; (d) applying a known technique to a known device (method, or product) ready for improvement to yield predictable results; (e) choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success; and (f) some teaching, suggestion, or motivation in the prior art that would have led a POSITA to modify the prior art or to combine prior art teachings to arrive at the claimed invention.

25. I further understand that certain factors may support or rebut the obviousness of a claim. I understand that such secondary considerations include, among other things, commercial success of the patented invention, skepticism of those having ordinary skill in the art at the time of invention, unexpected results of the invention, any long-felt but unsolved need in the art that was satisfied by the alleged invention, the failure of others to make the alleged invention, praise of the alleged invention by those having ordinary skill in the art, and copying of the alleged invention by others in the field. I understand that there must be a nexus—a

connection—between any such secondary considerations and the alleged invention.

I also understand that contemporaneous and independent invention by others is a secondary consideration tending to show obviousness.

26. I am not aware of any allegations by the named inventor of the '196 patent or any assignee of the '196 patent that any secondary considerations tend to rebut the obviousness of any Challenged Claim of the '196 patent.

V. THE '196 PATENT

A. Technology Overview

27. Before discussing the '196 patent, I will provide a brief overview of the relevant technology. The prior art discussed in the grounds below and the '196 patent relate to speech recognition using acoustic models, for example, hidden Markov models (HMM). The use of hidden Markov models in speech recognition was a widely known technique well before the earliest possible priority date of the '196 patent.

1. Speech Recognition

28. Speech recognition using computer systems was known and widely used well before the priority date of the '196 patent. Some of the earliest speech recognition systems were developed in the 1950s and 1960s. Ex.1022 (“Rabiner”), Pages 37-38 of 333. By the 1980s, speech recognition systems using Hidden Markov Models were also well known and “widely applied in virtually every speech-recognition research laboratory in the world.” Ex.1022, Page 39 of 333.

29. In general, speech recognition refers to “the technologies used to allow computers to recognize the words in human speech.” Ex.1018 (“Schmandt,” a textbook I published in 1994), Page 156 of 343. Indeed, speech was known to be used in “a variety of computing environments based on appreciating its role in human communication. Speech can be used as a method of **interacting** with a computer to place requests or receive warnings and notices. Voice can also be used as the underlying **data** itself[.]” Ex.1018, Page 26 of 343. Further, at least as early as 1994, “[d]esktop workstations [could] already support both these speech functions.” *Id.*

30. In general, it was known that speech recognition systems included three basic components:

(1) “speech **representation** that is computationally efficient for pattern matching” such as “Hidden Markov Models” that “are a statistical representation based on the various ways words or phonemes may be pronounced.” Ex.1018, Pages 156-157 of 343. I discuss Hidden Markov Models in more detail at Section V.A.4 below.

(2) A “set of **templates** or **models**, which are descriptions of each word to be recognized, in the representation of speech used by the recognizer. The templates describe the words in the recognizer’s vocabulary, i.e., those words that the recognizer can identify. They are reference models against

which an input can be compared to determine what was spoken.” Ex.1018, Page 157 of 343.

(3) A “**pattern matching** algorithm to determine which template is most similar to a specimen of speech input. This element of the speech recognizer must determine word boundaries, locate the most similar template, and decide whether the difference between the input and the selected template is minor enough to accept the word.” *Id.*

In general, a speech recognizer captures incoming speech and converts it “to the chosen internal representation (see Figure 7.1). The pattern matcher then selects the template that **most closely matches the input** or rejects the utterance if no template is a close enough match.” *Id.* The functional elements of an exemplary speech recognizer are depicted below in Figure 7.1.

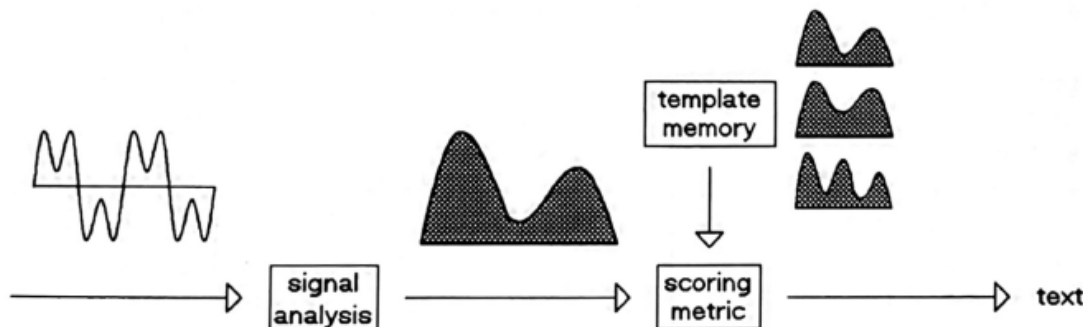


Figure 7.1. The functional elements of a speech recognizer. The user’s speech is digitized and converted to the recognizer’s internal representation. The captured speech is then compared with the words stored in the recognizer’s template memory. The pattern matching algorithm then determines which template, if any, is the closest match.

Ex.1018, Figure 7.1, Page 157 of 343.

31. When discussing speech recognition, one important topic is phonemes, which are speech sounds that form the words in a language:

The **phonemic** layer is concerned with the string of phonemes of which the words are composed. **Phonemes are the speech sounds that form the words of any language.** Phonemes include all the sounds associated with vowels and consonants. A grunt, growl, hiss, or gargling sound is not a phoneme in English, so it cannot be part of a word; such sounds are not referred to as speech. At the phoneme layer, while talking we are either continuously producing speech sounds or are silent. We are not silent at word boundaries; the phonemes all run together.

Ex.1018, Page 34 of 343. “Phonemes are important small units, several of which make up most syllables.” Ex.1018, Page 38 of 343. Further, a “**phoneme** is a unit of speech, the set of which defines all the sounds from which words can be constructed in a particular language. There is at least one pair of words in a language for which replacing one phoneme with another will change what is spoken into a different word. Of course, not every combination of phonemes results in a word; many combinations are nonsense.” Ex.1018, Pages 38-39 of 343.

32. One speech recognition technique known well before the priority date of the '196 patent was the statistical pattern-recognition approach to speech recognition. This well-known statistical pattern-recognition technique includes

four main steps: (1) feature measurement, (2) pattern training, (3) pattern classification, and (3) decision logic. Ex.1022, Page 82 of 333. The first step is “[f]eature measurement, in which a sequence of measurements is made on the input signal to define the ‘test pattern.’ For speech signals the feature measurements are usually the output of some type of spectral analysis technique, such as a filter bank analyzer, a linear predictive coding [LPC] analysis, or a discrete Fourier transform (DFT) analysis.” *Id.* The second step, pattern training, involves creating reference patterns or models that characterize “the statistics of the features of the reference pattern.” *Id.* During the pattern classification step, an “unknown test pattern is **compared** with each (sound) class reference pattern and a measure of similarity (distance) between the test pattern and each reference pattern is computed.” *Id.* Further, to “**compare** speech patterns (which consist of a sequence of spectral vectors), we require” “a local distance measure, in which local distance is defined as the spectral ‘distance’ between two well-defined spectral vectors[.]” *Id.* The fourth step is decision logic, “in which **the reference pattern similarity scores are used to decide which reference pattern (or possibly which sequence of reference patterns) best matches the unknown test pattern.**” *Id.*

33. For example, in Hidden Markov Model (HMM)-based speech recognition systems, “classification consists of determining which HMM has the highest probability of producing the observed acoustic sequence,” where “[t]he

Viterbi algorithm is a commonly used method[.]” Ex.1018, Page 171 of 343; *see also* Ex.1022, Page 190 of 333 (“A formal technique for finding this single best state sequence exists, based on dynamic programming methods, and is called the **Viterbi algorithm**.”). “As is well understood in the art, [] received utterances are analyzed with the models, e.g., **using a Viterbi algorithm**, and **scores are assigned representing how well the utterance fits the given models**[.]” i.e., Viterbi scores. Ex.1006, [0041].

2. Feature Vectors

34. Using feature vectors in speech recognition systems was well known before the priority date of the '196 patent. For example, it was known that in “calculating a distance between two words it is usual to derive a short-term distance that is local to corresponding parts of the words, and to integrate this distance over the entire word duration. Parameters representing the acoustic signal must be derived over some span of time, during which the properties are assumed not to change much.” Ex.1019, Page 13 of 102. Moreover, in “one such span of time the measurements can be stored as a set of numbers, or **feature vector**, which may be regarded as representing a point in multi-dimensional space. The properties of a whole word can then be described as a succession of feature vectors (often referred to as **frames**), each representing a time slice[.]” *Id.*

35. In general, feature vectors are representations of speech where each dimension of the vector corresponds to “some prescribed acoustic feature”:

In speech processing, speech is generally represented by an n-dimensional space in which each dimension corresponds to some prescribed acoustic feature. For example, each component may represent a amplitude of energy in a respective frequency band. For a given time interval of speech, each component will have a respective amplitude. **Taken together, the n amplitudes for the given time interval represent an n-component vector in the n-dimensional space.**

Ex.1023 (“Nadas”), 1:15-23. Additionally, “[w]hen an unknown speech input is uttered, for each time interval, a value is measured or computed for each of the n components, where each component is referred to as a feature. The values of all of the features are consolidated to form an n-component feature vector for a time interval.” Ex.1023, 1:32-37. “The components of the feature vectors (and prototype vectors)” may “correspond to **well-known (1) Cepstral coefficients, (2) linear predictive coding coefficients, or (3) frequency band-related characteristics.**”

Ex.1023, 10:15-19. Moreover, a “**speech type of event, such as a word or a phoneme, is characterized by a sequence of feature vectors in the time period over which the speech event was produced.**” Ex.1023, 1:50-52.

36. In one example, Konopka illustrates using well-known feature extraction techniques “to extract parameters or feature vectors representing acoustic data”:

The acoustic data is then passed to the **feature extraction module 202**, which is used to extract parameters or **feature vectors representing information related to the incoming acoustic data**.

The feature extraction module 202 performs edge-detection, signal conditioning and feature extraction. According to one embodiment, speech edge detection is accomplished using noise estimation and energy detection based on the 0th Cepstral coefficient and zero-crossing statistics. Feature extraction and signal conditioning consist of extracting Mel-frequency cepstral coefficients (MFCC), delta information and acceleration information. **It is a 38 dimensional feature vector based on 12.8 ms sample buffers overlapped by 50%.** Such feature extraction modules 202 and functionality are well understood in the art, and that one skilled in the art may implement the feature extraction module in a variety of ways. **Thus, the output of the feature extraction module 202 is a sequence of feature vectors.**

Ex.1006 (“Konopka”), [0037]-[0038].

37. Aoyama illustrates another example of using well-known feature extraction techniques in which a feature extraction unit analyzes audio input by

performing a Mel Frequency¹ Cepstrum Coefficient (MFCC) analysis and extracts information to generate a feature vector utilizing a predictive coefficient (LPC) or Cepstrum coefficient. Ex.1024 (“Aoyama”), [0138]-[0139], [0150]-[0152], [0156]-[0160].

38. It was well-known that converting speech samples into feature vectors is advantageous because feature vectors are more compact and provide better speech recognition performance than the speech samples themselves:

The purpose of the signal processing component is to convert the utterance, which is a series of samples of speech, into a more compact series of feature vectors. Besides being more compact, the feature vectors provide better speech recognition performance since they tend to remove many speaker-dependent effects and improve the robustness of the system to noise.

Ex.1025 (“Stubley”), 11:30-36.

39. In the context of speech recognition systems using Hidden Markov Models (HMMs), it was also known to analyze input speech and represent it as feature vectors that correspond to phonemes that are characterized by HMMs:

¹ I note that the Mel scale is a frequency representation which approximates the frequency response of the human auditory system, i.e., how humans hear frequency.

[I]nput speech is analyzed in signal frames, represented by feature vectors corresponding to phonemes making up individual words. The phonemes are characterized by hidden Markov models (HMMs), and a Viterbi algorithm is used to identify a sequence of HMMs which best matches, in a maximum likelihood sense, a respective concatenation of phonemes corresponding to an unknown, spoken utterance.

Ex.1026, 1:23-31.

3. Acoustic Models

40. It was also known before the priority date of the '196 patent to use acoustic models in speech recognition systems. For example, it was known to use a set of templates or models as a representation of speech used by the speech recognizer:

A set of **templates** or **models**, which are descriptions of each word to be recognized, in the representation of speech used by the recognizer. The templates describe the words in the recognizer's vocabulary, i.e., those words that the recognizer can identify. They are reference models against which an input can be compared to determine what was spoken.

Ex.1018, Page 157 of 343.

41. Further, it was known to use acoustic models representing known speech to compare unknown speech to, for performing speech recognition. For

example, Stubbley illustrates that it was well-known to “compar[e] a series of observations representing unknown speech, to stored models representing known speech” where “the models comprise representations of elements of speech[.]”

Ex.1025, 1:5-10, 1:60-62, 2:51-52. More specifically, “the **models [could] comprise groups of representations of phonemes**. This enables the models to be stored and used more efficiently.” Ex.1025, 2:40-42.

42. It was also known to use a hidden Markov model as an acoustic model in speech recognition. For example, speech signals can be characterized in terms of signal models—a “signal model can provide the basis for a theoretical description of a signal processing system which can be used to process the signal so as to provide a desired output.” Ex.1021 (“Rabiner 89”), Page 1 of 30. Further, such signal models include “statistical models in which one tries to characterize only the statistical properties of the signal[.]” with one example being a “hidden Markov model (HMM)” *Id.* Thus, one particular type of acoustic model is a Hidden Markov Model, which I discuss in detail below. Moreover, “[t]he basic principle behind **HMM-based speech recognition systems** is that an utterance is compared with a number of speech models (**from the acoustic models** module 220) in order to find the model that best matches the utterance.” Ex.1006, [0043].

4. Hidden Markov Models

43. Before the priority date of the '196 patent, Hidden Markov Models were used as a technique to “make pattern matching more powerful for dealing with variations in speech patterns.” Ex.1018, Page 165 of 343. In general, a Hidden Markov Model (HMM) is “a two-stage probabilistic process that can be used as a powerful representation of speech. A Hidden Markov Model is a well-behaved mathematical construct[.]” Ex.1018, Page 168 of 343. Further, it was known that HMMs consisted of states. “An HMM consists of a number of internal states; the model passes from an initial state to the final state as a step-by-step process generating an observable output at each step (state transition). For example, the states may correspond to **phonemes** in a word with the observable output corresponding to the presence or absence of a number of acoustic features.” Ex.1018, Pages 168-169 of 343. Additionally, “[a]t each step, the model can either move to a new state or stay in the current one. The model is ‘hidden’ in that we cannot observe the state directly but only its output. From the series of observable outputs, we can attempt to guess when the model was in each state.” Ex.1018, Page 169 of 343. “Alternatively, we can say whether some series of outputs was likely to have been generated by a particular HMM.” *Id.*

44. Schmandt illustrates a “pedagogic example” of a “box-and-bowl setup” to explain Hidden Markov Models:

For example, consider the arrangement shown in Figure 7.10, in which a box and a bowl are each full of black balls and white balls. The box has many more black balls than white while white balls dominate the bowl. Starting with the box, we remove one ball from it at random and pass it to an observer in another room who cannot see what is being done. Then a normal six-sided die is tossed. If the result is less than four, then the source of the next ball is the bowl. If the die is greater than three, then the next ball will be selected from the box. The cycle is then repeated. Whenever we select from the box, a die throw of less than four shifts attention to the bowl. Once we start selecting from the bowl, we continue selecting from it unless the die shows a one, at which point we are finished. This model is likely to spend a majority of its cycles selecting from the bowl since a state transition from the box has a probability of $1/2$ of shifting to the bowl, but a bowl transition has a probability of only $1/6$ of terminating and $5/6$ of continuing with the bowl. We should expect this arrangement to initially present more black balls than white to the observer (while it is in the box state) and then to produce more white balls. We should also expect more white balls overall. But keep in mind that this is a probabilistic process: Not all black balls come from the box and not all selections come from the bowl.

Id. In this example, the “box-and-bowl setup is a Hidden Markov Model; it is

characterized by a number of **states** and associated probabilities for each **transition** from any state. The box and bowl are each a state; the rules about throwing the die define the transition properties.” Ex.1018, Pages 169-170 of 343. The “[p]robability is expressed as a number between zero and one. If an event occurs with probability $1/2$, we would expect it to happen once in two trials. A probability of $1/6$ implies that we can expect the event once out of six trials. A smaller probability indicates that an event is less likely to occur.” Ex.1018, Page 169 of 343.

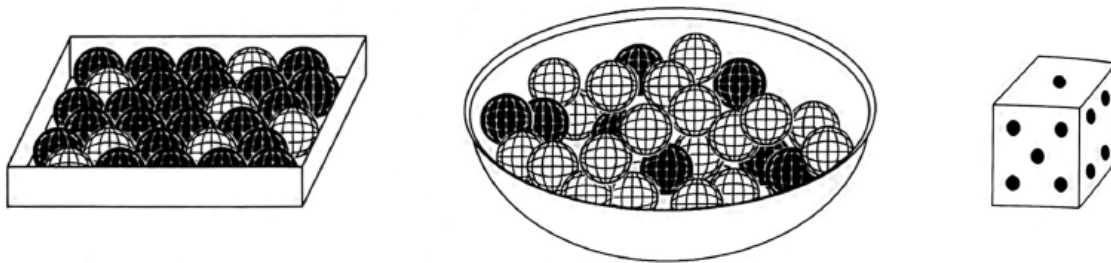


Figure 7.10. A box and a bowl full of colored balls illustrate a Hidden Markov Model.

Ex.1018, Figure 7.10, Page 170 of 343.

45. Further, a “more general HMM” is depicted in Figure 7.11 “A Markov Model” below. In the general HMM, “in each state $S(n)$ there is some probability $P(n,n)$ of remaining in that state and some probability $P(n, n+1)$ of transitioning to the next state.” Ex.1018, Page 170 of 343. Additionally, “[i]n some models it may be possible to bypass a state or a series of states as indicated by the dashed arc labeled P_{13} in the figure. The sum of the probabilities of all the arcs

leaving a state is one; the model accounts for every possible path through its states.” *Id.* .

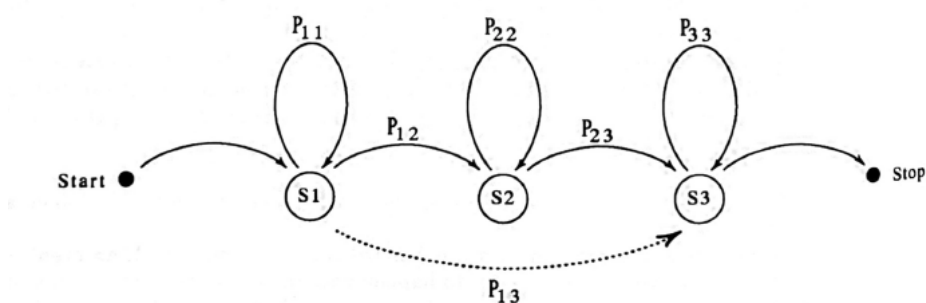


Figure 7.11. A Markov Model.

Ex.1018, Figure 7.11, Page 170 of 343.

46. The example of the box-and-bowl setup and the more general HMM model apply to speech recognition because an HMM’s internal states can represent acoustic segments:

An HMM can be used to represent a word with internal states representing characteristic acoustic segments, possibly phonemes or allophones. The output of a state is a frame or vector of acoustic parameters or features; this output is probabilistic to allow for variability in pronunciation and hence differences in the acoustic representation. The duration of an acoustic segment is a function of the number of steps in which the model is in the state corresponding to the segment. Staying in the same state, i.e. lengthening a phone, depends on the probability associated with the transition from that state to itself (P_{11} in Figure 7.11). Arcs

such as P_{13} may be included to indicate that an intermediate state **S2** is optional during pronunciation, such as the second syllable in ‘chocolate.’

In terms of the basic recognizer components, the ‘templates’ consist of a set of HMMs with one HMM associated with every word. The acoustic representation of an input specimen of speech is *not* a collection of states but rather a set of acoustic parameters corresponding to the possible observed sequences from the template HMMs. For the HMM-based recognizer, classification consists of determining which HMM has the highest probability of producing the observed acoustic sequence. The Viterbi algorithm is a commonly used method of solving this classification problem[.]

Ex.1018, Pages 170-171 of 343.

47. “Hidden Markov Models are a powerful representation of speech and consequently are used in many speech recognition systems currently under development.” Ex.1018, Page 171 of 343. HMMs can be used to “represent not only the phones of each word but also the probabilities of transitioning from one word to another.” *Id.*

48. Further, an observation vector in an HMM can be “LPC-derived weighted cepstral coefficients and weighted cepstral derivatives[.]” Ex.1022, Page 231 of 333. For example, an observation vector in the HMM can be a “38-

component **vector** consisting of 12 **cepstral coefficients**, 12 delta cepstrum coefficients, 12 delta-delta cepstral coefficients, delta log energy, and delta-delta log energy.” Ex.1022, Page 280 of 333; *see also* Ex.1022, Pages 301-302 of 333 (teaching spectral vectors with 38 components).

49. The use of HMMs in speech recognition is a widely known technique that POSITAs have been using since well before the earliest possible priority date of the '196 patent. For example, Nadas (issued in 1990) illustrates an acoustic coprocessor for speech recognition and “contemplates a probabilistic embodiment in which each prototype—rather than identifying a vector—corresponds to a probabilistic finite state machine PFSM (or **Markov model**).” Ex.1023, 9:68-10:4.

50. In another example, Aoyama uses an HMM as the acoustic model for speech recognition, storing it stored in an acoustic model memory unit:

The matching unit 92, referring to the acoustic model memory unit 93, the dictionary memory unit 94 and the grammar memory unit 95 in utilizing the feature vector from the feature extraction unit 91 as occasion demands, speech recognizes the voice (input speech) entered into the microphone 51 based on such as the Hidden Markov model (HMM) law.

More specifically, **the acoustic model memory unit 93 memorizes acoustic model** (e.g., including the standard pattern to be used in DP (Dynamic Programming)

matching, other than HMM) showing the acoustic feature on the sub-words such as phoneme, syllable, and phoneme series in the audio language for identifying the speech. **Here, since the speech recognition is conducted based on the Hidden Markov Model law, the HMM will be used as the acoustic model.**

Ex.1024, [0139]-[0140]; *see also* Ex.1024, [0150]-[0151].

51. In yet another example, Stubbley uses HMMs as acoustic models:

Advantageously, the models are represented as finite state machines with probability distribution functions attached. Continuously varying signals can be modeled as a series of stationary states, with a probability function at each state, indicating whether the signal moves to the next state or stays at the given state, after a given time interval. An example is the **Hidden Markov Model (HMM)**. This enables the models to contain a representation of patterns with varying rates of change, yet a common underlying pattern.

Ex.1025, 2:29-37; *see also* Ex.1025, 7:23-35, 12:29-44 (“**The modelling paradigm used here is to construct models using hidden Markov models (HMMs)[.]**”).

52. In another example, Lee describes analyzing input speech represented by feature vectors that correspond to phonemes of words, where phonemes are characterized by Hidden Markov Models:

[I]nput speech is analyzed in signal frames, represented by feature vectors corresponding to phonemes making up individual words. The phonemes are characterized by hidden Markov models (HMMs), and a Viterbi algorithm is used to identify a sequence of HMMs which best matches, in a maximum likelihood sense, a respective concatenation of phonemes corresponding to an unknown, spoken utterance.

It is well known that each HMM comprises model parameters, e.g., mixture components which are characterized by Gaussian distributions. **In a learning phase in a speech recognition system, the HMMs are adapted to input speech by a user to adjust to the particular speech characteristics of the user, thereby increasing accuracy of the speech recognition.**

Ex.1026, 1:23-38 .

53. In yet another example, Toyoda illustrates using a continuous HMM to recognize voice input and referencing acoustic models representing phonemes:

The voice recognition module recognizes a voice input to the microphone 9 using a continuous HMM (Hidden Markov Model), and outputs voice recognition results.

Ex.1027 (“Toyoda”), 7:48-50.

Using the feature parameters from the feature parameter extractor 41, the **matching unit 42 recognizes the voice**

input to the microphone 9 based on the continuous HMM model while referencing an acoustic model memory 43, a dictionary memory 44, and a grammar memory 45 as necessary.

The acoustic model memory 43 stores an acoustic model that represents an acoustic feature such as phonemes and syllables in a voice to be recognized. Since voice recognition is here carried out using the continuous HMM, an HMM is employed.

Ex.1027, 7:61-8:5.

54. Bou-Ghazale illustrates another example of the well-known technique of using HMMs for speech recognition:

We propose a real-time speaker-independent voice activation system composed of an **HMM-based continuous speech recognition algorithm**.

Ex.1005 (“Bou-Ghazale”), Sec.1, 1735.

55. Moreover, Konopka illustrates yet another example of using HMMs for speech recognition:

[T]he speech recognition module 204 functions as a **Hidden-Markov Model (HMM)-based continuous speech recognizer** that has the ability to reject “unmodeled events”, e.g. out-of vocabulary events, disfluencies, environmental noise, etc.... The feature vectors from the feature extraction module 202 are input

to the speech recognition module 204, i.e., input to the speech decoder 216 of the speech recognition module (SRM) 204. Thus, the speech recognition module (SRM) 204 is responsible for requesting speech feature vectors from the feature extraction module (FEM) 202 and **finding the most likely match of the corresponding utterance with a set of speech models, while rejecting non-speech events, using an approach based on Hidden Markov Models (HMMs).**

Ex.1006, [0039]-[0040].

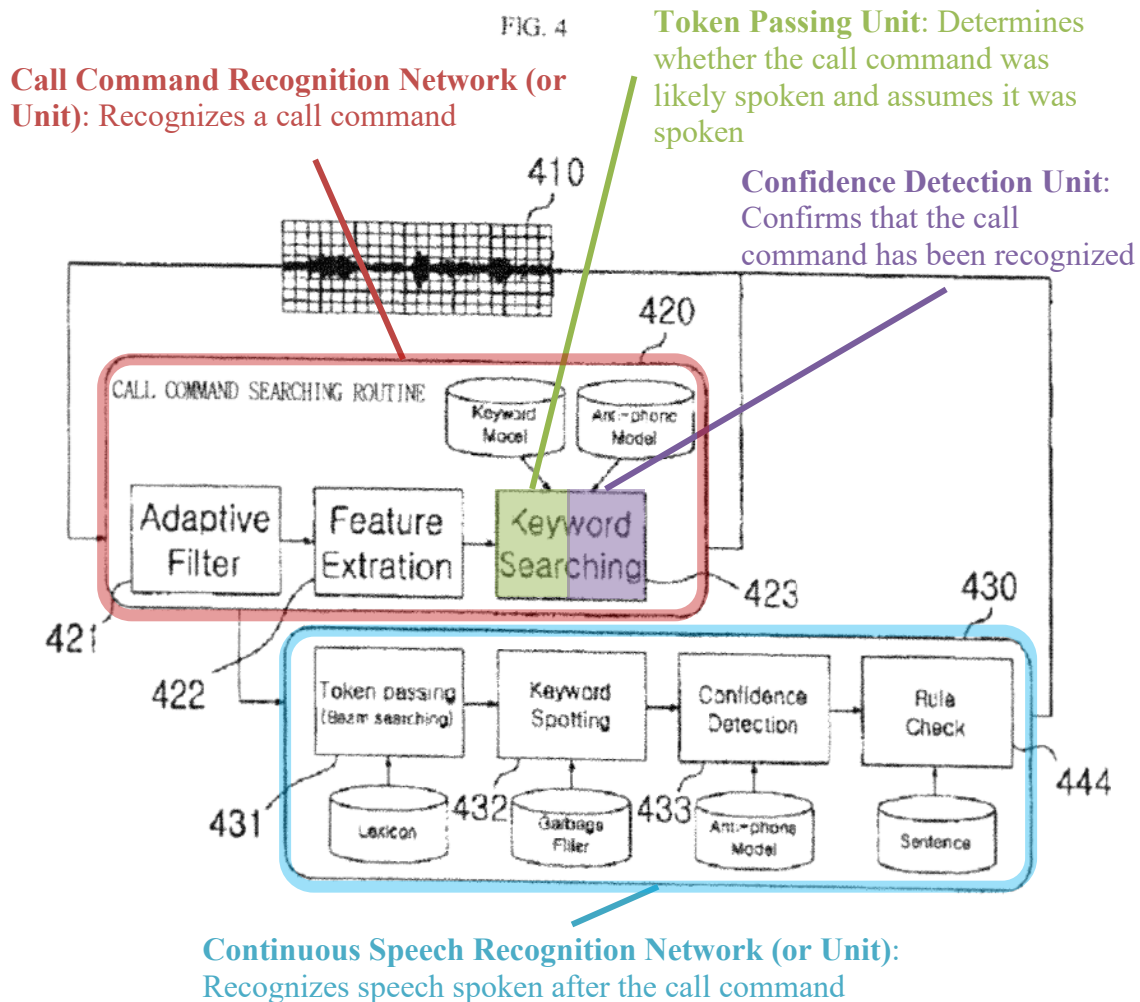
56. Further, Konopka illustrates an example of the well-known technique of token passing—a technique for tracking the best word (or state) sequence in a model (e.g., HMM) that matches unknown speech—in the context of HMMs and speech recognition:

Token Passing is a well-known approach to tracking the best word hypotheses through an HMM....[T]his token passing technique provides a highly memory efficient solution to an otherwise well-known problem in HMM-based speech recognition systems; the storage of back-pointers that allows for finding the best word sequence hypothesis once all speech data has been processed.

Ex.1006, [0052].

B. Overview of the '196 Patent

57. The '196 patent relates to a two-step “continuous speech recognition” method and system that continually listens for a “call command” (i.e., a wake word), and, if recognized, listens for a subsequent command (i.e., the “actual command”). Ex.1001, 1:9-10, 6:24-28. The '196 patent's system is depicted in Fig. 4 below.



Ex.1001, Fig. 4 (annotated).

58. As shown above, the system includes a “call command recognition network 420” for “recognizing call commands and detecting the confidence thereof.” Ex.1001, Fig. 4, 5:7-12. The system also includes a “continuous speech recognition network 430” for “recognizing speech which is an actual command.” Ex.1001, Fig. 4, 6:24-28. The continuous speech recognition unit is “activat[ed]” if a call command is recognized by the call command recognition unit. Ex.1001, claim 1.

59. With respect to the call command recognition network, it “includes...a keyword searching unit 423.” Ex.1001, Fig. 4, 5:7-12. “The keyword searching unit 423 includes a token passing unit...and a confidence detection unit.” Ex.1001, Fig. 4, 5:40-52. To determine whether the call command was spoken, “based on probability, every input speech frame is compared with a recognition network for only one word in real time.” Ex.1001, 6:52-7:3, 3:64-4:2, 7:34-39, claim 1. Specifically, the call command recognition network “us[es] a left-to-right model” for one word to perform this comparison, such as a “keyword model” as shown in Fig. 4. Ex.1001, 3:64-4:2, 6:52-7:3, Fig. 4. The call command recognition network also accounts for “noise” and “silence” in input speech frames: “the keyword searching unit 423 includes a token passing unit for configuring a speech frame, which includes a silence interval accompanied by noise and call commands from the speaker.” Ex.1001, 5:40-52, 5:13-19.

60. If it is determined that the call command was probably spoken—based on the “tokens” that have the highest “accumulated probability”—the token passing unit “assume[s] that an actual call command has been spoken”:

When the **accumulated probability of a relevant token**, obtained at the moment at which the transition to a silence model is made while real-time computation is performed on the frames which are continuously input, **corresponds to the top 10% of the 20 to 30 tokens in the call command recognition network**, it is assumed that an **actual call command has been spoken**, and information about a relevant frame is transferred to the confidence detection routine 421, 422 and 423.

Ex.1001, 6:62-7:3.

However, in the call command recognition network according to the present invention, the accumulated probabilities and model transition states for only 20 to 30 tokens are stored. Among these 20 to 30 tokens, only tokens for which the uppermost probability is stored are required in FIG. 6, **when a relevant token is a token for which the accumulated probability at the moment at which the transition to silence is made in as is stored as the uppermost probability, it is recognized that the call command has been spoken**, and the detection of confidence thereof is performed.

Ex.1001, 7:29-39; *see also* Ex.1001, 7:18-24, claim 1.

61. Then, the confidence detection unit confirms that the call command is recognized based on whether the confidence rate of the input speech (represented by a “log likelihood ratio” (LLR)) meets a threshold. Ex.1001, 8:28-42, 7:63-8:15. “When LLR is determined...and ‘LLR>threshold’ is satisfied, the continuous speech recognition network is executed” and “speech spoken subsequent to the call command is transferred to the continuous speech recognition engine to perform continuous speech recognition.” Ex.1001, 7:63-8:15, 8:32-42, 6:24-28.

62. As I will explain below, however, each of the above speech recognition concepts were previously known in the art. *See, e.g.*, Ex.1006, [0052] (“token passing” is “a well-known approach to tracking the best word hypotheses”).

VI. CLAIM CONSTRUCTION

63. It is my understanding that in order to properly evaluate the ’196 patent, the terms of the claims must first be interpreted. It is my understanding that for the purposes of this *inter partes* review, the claims are to be given their ordinary and customary meaning as would have been understood by a POSITA in light of the specification and prosecution history, unless the inventor has set forth a special meaning for a term. I have also been informed that claim terms only need to be construed to the extent necessary to resolve the obviousness inquiry. I have reviewed the entirety of the ’196 patent, as well as its prosecution history. It is my

opinion that for purposes of applying the prior art presented herein to evaluate the patentability of the Challenged Claims, no claim terms require express construction.

64. I have also been informed that in the district court proceeding concerning the '196 patent, the Patent Owner's position in its Infringement Contentions is that no terms invoke §112, ¶6. In the event the Board finds any of the "*unit*" terms in the Challenged Claims invoke §112, ¶6, I have been instructed to adopt a "combination of hardware, processing, and software functions" as the corresponding structure, which I have been informed mimics the language Patent Owner uses to describe Petitioner's allegedly infringing products in the Infringement Contentions.

VII. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE

65. I have been asked to provide my opinion as to whether the Challenged Claims of the '196 patent would have been obvious in view of the prior art. The discussion in this Declaration provides a detailed analysis of how the asserted prior art references teach each limitation of the Challenged Claims.

66. As part of my analysis, I have considered, and discuss in detail, the scope and content of the prior art and any differences between the alleged invention and the prior art.

67. As described in detail below, it is my opinion that the alleged invention recited in the Challenged Claims would have been obvious in view of the teachings of the asserted prior art and the knowledge of a POSITA before the time of the alleged invention of the '196 patent. I apply the prior art as follows:

Grounds	Claim(s)	Basis
#1	1-5, 8-9, 16-17, 19-20	§103 Bou-Ghazale and Konopka
#2	6-7	§ 103 Bou-Ghazale , Konopka , and Lee
#3	10-13, 18	§ 103 Bou-Ghazale , Konopka , and Takiguchi
#4	14-15	§ 103 Bou-Ghazale , Konopka , Takiguchi , and Lee

A. Ground 1: Claims 1-5, 8-9, 16-17, and 19-20 are obvious under §103 over Bou-Ghazale in view of Konopka.

1. Summary of Bou-Ghazale

68. Like the '196 patent, Bou-Ghazale describes a two-step continuous speech recognition system that “continuously” listens for a “pre-defined activation command” while rejecting extraneous speech and noise, and, if recognized, listens for a subsequent, actual command. Ex.1005, Abstract (“We propose a real-time speaker-independent voice activation system composed of an HMM-based continuous speech recognition algorithm.”), Sec.1, 1735. More specifically, Bou-Ghazale relates to a “real-time speaker-independent voice activation system” that is “designed to continuously listen to incoming unconstrained speech and to

recognize a pre-defined activation command while rejecting any other extraneous speech or noise.” Ex.1005, Abstract. “Upon recognizing that the activation command has been spoken, the voice activation system turns [a] device on or switches to [a] main speech recognition system to receive further commands from the user.” Ex.1005, Sec.1, 1735. The overall system provides a convenient, hands-free way to activate and control a device with one’s voice in noisy “busy-hands busy-eyes environments.” *Id.* For instance, when “operating a cellular phone in a moving car” a user does not have to “manually activate the device of the speech recognition system by pushing a button before...issu[ing] voice commands to control the device.” *Id.*

69. In more detail, with reference to Figure 2 below, Bou-Ghazale’s voice activation system includes a hidden Markov model (HMM)-based speech recognition component that continually “processes input speech on a frame-by-frame basis and makes a decision at every frame whether the activation word has been spoken.” Ex.1005, Sec.2, 1735; *see also* Ex.1005, Sec.2.2, 1736. If the speech recognition component determines that it is likely that the activation word has been spoken, then a post-processing component next “validates the speech recognition result” using a set of criteria and confidence measurements. Ex.1005, Sec.2, 1735, Sec.2.3, 1736; *see also* Sec.3.1-3.2, 1736-37. Once the post-processing component

confirms that the activation word has been recognized, then the main speech recognition system is activated for recognizing “further commands.” *Id.*

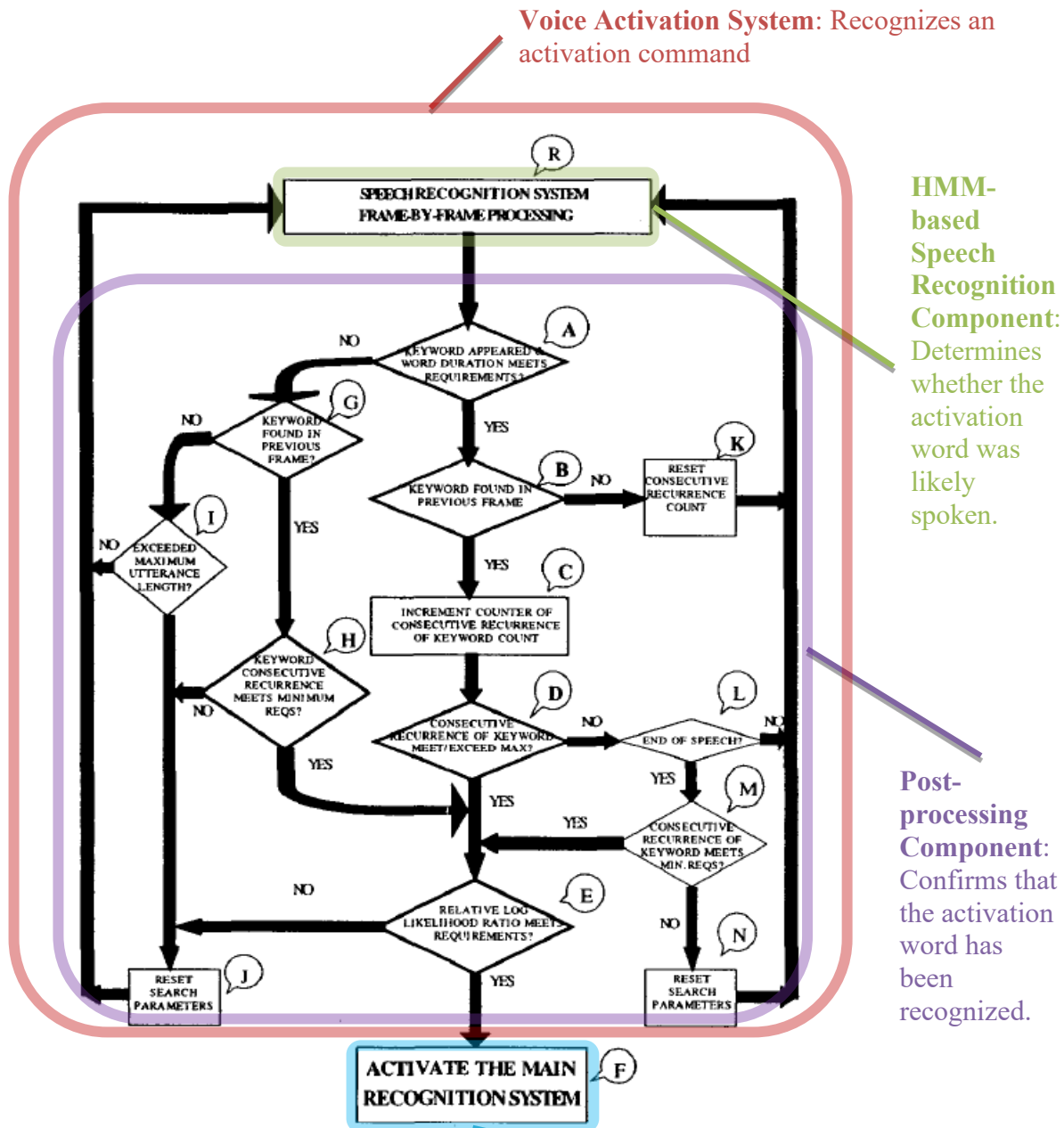


Figure 2. Flow graph of the post-processing algorithm

Ex.1005, Figure 2 (annotated).

2. Summary of Konopka

70. Like the '196 patent and Bou-Ghazale, Konopka relates to HMM-based continuous speech recognition. Ex.1005, Abstract; Ex.1006, [0002] (“The present invention relates to speech recognition, and more specifically to natural language speech recognition.”), [0039] (“the speech recognition module 204 functions as a Hidden-Markov Model (HMM)-based continuous speech recognizer”). Also like the '196 patent and Bou-Ghazale, Konopka describes a two-step speech recognition system that first searches for an attention word (while rejecting other noises) and upon recognition of the attention word, it listens for subsequent speech, such as a user request. Ex.1006, Abstract, [0039], Fig. 4.

71. Konopka describes a method and device for speech recognition that uses a “Hidden-Markov Model (HMM)-based continuous speech recognizer that has the ability to reject ‘unmodeled events,’ e.g. out-of-vocabulary events, disfluencies, environmental noise, etc.” Ex.1006, [0039]. Konopka teaches that it was well-known in speech recognition systems, that in order to determine the best state (or word) sequence, “it is required to perform ‘back tracing’.” Ex.1006, [0052]. However, Konopka explains that additional improvements were necessary to decrease the memory requirements associated with the traditional approach to back-tracing. *See id.* (“Various methods have been proposed to minimize the memory requirements associated with storing such back-pointers[.]”). To this end,

Konopka explains that “token passing” was a “well-known approach” to improving HMM-based systems. *Id.* (“Token Passing is a well-known approach to tracking the best word hypotheses through an HMM.”). Token passing is a back-tracing technique that is “based on the idea of passing ‘tokens’ around to the various states instead of allocating memory on a per-state basis.” *Id.* Accordingly, token passing provides a highly memory efficient solution to the known problem in HMM-based systems—“storage of back-pointers that allows for finding the best word sequence hypothesis once all speech data has been processed.” *Id.*

3. Reasons to Combine Bou-Ghazale and Konopka

72. It is my opinion that a POSITA would have been motivated to combine the teachings of Bou-Ghazale and Konopka. It would have been obvious, beneficial, and predictable to utilize Konopka’s token passing technique in combination with Bou-Ghazale’s voice activation system. This combination would, for example, provide a more memory efficient way of matching incoming speech to known words.

73. As an initial matter, a POSITA, when considering the teachings of Bou-Ghazale would have also considered the teachings of Konopka. Both Bou-Ghazale and Konopka relate to HMM-based continuous speech recognition. Ex.1005, Abstract (“We propose a real-time speaker-independent voice activation system composed of an HMM-based continuous speech recognition algorithm.”);

Ex.1006, [0002] (“The present invention relates to speech recognition, and more specifically to natural language speech recognition.”), [0039] (“the speech recognition module 204 functions as a Hidden-Markov Model (HMM)-based continuous speech recognizer”). As such, both Bou-Ghazale and Konopka are analogous art to the ’196 patent which intends to provide a “continuous speech recognition system to recognize continuous speech smoothly in a noisy environment.” *See*, e.g., Ex.1001, Abstract. Further, the system described in the ’196 patent is based on hidden Markov models considering the diagram of the call command recognition network shown in Figure 6 which is a graphical representation of a hidden Markov model. Ex.1001, Fig. 6; *See* Ex.1015, Figure 2, Pages 2, 26, 30 of 589, 13 (illustrating “a basic HMM as a collection of states”); Ex.1016, Figure 1, Page 3 of 23 (showing a “structure of a word model” that uses HMMs); Ex.1017, Figure 2, Page 6 of 40 (showing an “HMM representation” of “typical model structures” for word models); Ex.1018, Figure 7.11, Page 170 of 343 (showing a “general HMM” that can be used to represent a word).

74. As explained in detail below, a POSITA would have been specifically motivated to incorporate Konopka’s token passing technique into Bou-Ghazale because doing so would address a well-known problem with HMM-based speech recognition systems, such as Bou-Ghazale’s system—“high memory requirements.” Ex.1006, [0052]. A POSITA would understand that applying

Konopka's token passing technique would enhance Bou-Ghazale's system, and thus a POSITA would seek to combine them.

75. In more detail, Bou-Ghazale teaches a voice activation system that includes an HMM-based continuous speech recognition component that **“performs a trace-back at every frame, and passes the result string with the highest likelihood score** to the post-processing component.” Ex.1005, Sec.2.2, 1736. However, Bou-Ghazale does not explicitly describe how it performs back-tracing to determine the best result string (i.e., the one with the highest likelihood score), including whether it uses the traditional approach or token passing.

76. Konopka explains that a well-known problem with the traditional approach to performing back-tracing is that it requires “each state contain a pointer back to the previously best state for each frame” which can lead to “high memory requirements” for storing the back-pointers. Ex.1006, [0052]. As a result, Konopka explains that “various methods have been proposed to minimize the memory requirements associated with storing such back-pointers.” *Id.* For example, Konopka provides a “well-known” and predictable solution to this problem “based on the idea of passing ‘tokens’ around to the various states instead of allocating memory on a per-state basis”—this back-tracing technique is called “token passing.” *Id.* (“Token passing is a well-known approach to tracking the best word hypotheses through an HMM.”). In Konopka's token passing technique, the speech

decoder “looks back” to the previous state in the word model state sequence and determines whether it is similar (i.e., a “member of the same acoustic model”) to the current state; the token pointer associated with the best state is kept:

[I]nstead of storing one token pointer in each state, the speech decoder 216 uses two arrays S₁ and S₂ to hold the token pointers for each state. Array S₁ keeps the token pointers for each state and the previous frame, and S₂ keeps the token pointers for each state and the current frame. **When each state i “looks back” to find the previously best state j, two things can happen.** If the previous best state j is a member of the same acoustic model as i, the token pointer for state j in S₁ is copied into position i in S₂. **If this is not the case, a new token is created and stored in position i in S₂.** The new token gets the same contents as token i in S₁, and in the token history, a reference to model m, i.e. m, is added. Once all states have been processed for the current frame, the pointers to structures S₁ and S₂, are swapped, and the process is repeated for the following frame.

Id. Accordingly, using Konopka’s well-known token passing technique, token pointers corresponding to the sequence (or string) of states that are less likely to correspond to the unknown input speech are discarded at each state during the back tracing process. Ex.1006, [0052]. As a result, “a highly memory efficient solution to an otherwise well-known problem in HMM-based speech recognition systems;

the storage of back-pointers that allows for finding the best word sequence hypothesis once all speech data has been processed” is provided. *Id.*

77. A POSITA would have been motivated to implement Konopka’s token passing technique for HMM-based recognition systems into the HMM-based speech recognition component of Bou-Ghazale’s voice activation system to provide a more memory efficient solution to the well-known problem of storing back-pointers using traditional back-tracing techniques. Specifically, the benefits of Konopka’s technique would have motivated a POSITA to use token passing for Bou-Ghazale’s back-tracing technique. Accordingly, when Bou-Ghazale’s speech recognition component (as modified by Konopka) “performs a trace-back at every frame” it would discard the strings that are less likely to match unknown speech and keep those that are a more likely match, until it determines the result string that most likely matches the unknown speech (“the result string with the highest likelihood score”) and passes it to the post-processing component. Ex.1005, Sec.2.2, 1736 As explained by Konopka, this solution would “minimize the memory requirements associated with storing” back-pointers and make Bou-Ghazale’s system more efficient. Ex.1006, [0052].

78. A POSITA would have had a reasonable expectation of success applying Konopka’s technique to Bou-Ghazale. For example, the combination does not change the intended functionality of either Bou-Ghazale’s voice activation

system or Konopka's token passing technique. Konopka's token passing was specifically intended to improve HMM-based speech recognition systems, such as Bou-Ghazale's voice activation system. *See* Ex.1006 [0052] (“[T]his token passing technique provides a highly memory efficient solution to an otherwise well-known problem in HMM-based speech recognition systems”). Moreover, POSITAs routinely utilized token passing in the context of HMM-based systems. Ex.1015, Pages 20-23 of 58 (describing a decoder for an automatic speech recognition system that uses a “**token passing algorithm**”), 29 of 58 (“The Acoustic Model (AM) is a mathematical model of a speech unit, for example word, phone or syllable. The preferred model unit is the phone model, and **the preferred acoustic model is the Hidden Markov Model (HMM)**”); Ex.1016, 2-5 (describing a “token passing model” for recognition of a word in unknown speech that is used with Hidden Markov Models (HMM's) but “is not restricted to HMM's” and explaining that “the correspondence between the token passing model and hidden Markov models is obvious and direct”). Thus, it would have been within the background knowledge and standard capabilities of a POSITA to modify Bou-Ghazale's speech recognition algorithm to use token passing as its back-tracing technique.

79. Accordingly, a POSITA would have found it obvious to apply Konopka's teachings to Bou-Ghazale's voice activation system and methods

because the combination merely amounts to applying a known token passing technique to a system ready for improvement.

4. Detailed Analysis of the Claims

a) Claim 1

[1.0] *A method for recognizing a speech of a speaker by a device, the method comprising:*

80. First, Bou-Ghazale describes a method for recognizing a pre-defined activation command (“*speech*”) spoken by a user (“*speaker*”) using a voice activation system. Bou-Ghazale explains that its voice activation system is “designed to continuously listen to incoming unconstrained **speech** and to **recognize a pre-defined activation command** while rejecting any other extraneous speech or noise.” Ex.1005, Abstract, 1735. “Upon **recognizing that the activation command has been spoken**, the voice activation system turns the **device** on or switches to the main speech recognition system to receive further commands from the **user**.” Ex.1005, Sec.1, 1735; *see also* Ex.1005, Abstract, 1735 (“The proposed system is intended for turning a device on or for activating a main speech recognition application upon **recognizing the activation utterance**.”).²

² In Bou-Ghazale, the terms “pre-defined activation command,” “pre-selected activation utterance,” “activation word,” “desired word,” and “keyword” are used interchangeably.

Specifically, the method of recognizing the activation command using the components of the voice activation system involves “a speech recognition component process[ing] the input speech on a frame-by-frame basis and mak[ing] a decision at every frame whether the **activation word** has been spoken[.]” as illustrated by Block R in the flow diagram of Figure 2 below. Ex.1005, Sec.2, 1735; *see also* Ex.1005, Sec.2.2, 1736 (“The **recognition algorithm** performs a trace-back at every frame and passes the result string with the highest likelihood score to the post-processing component.”). Then, a “postprocessing component validates the **speech recognition result**” according to the steps of the “post-processing **algorithm**” illustrated in Figure 2 below. Ex.1005, Sec.2, 1735; *see also* Ex.1005, Sec.2.3-3.2, 1736-1737 (detailing the steps of the “post-processing

algorithm”).

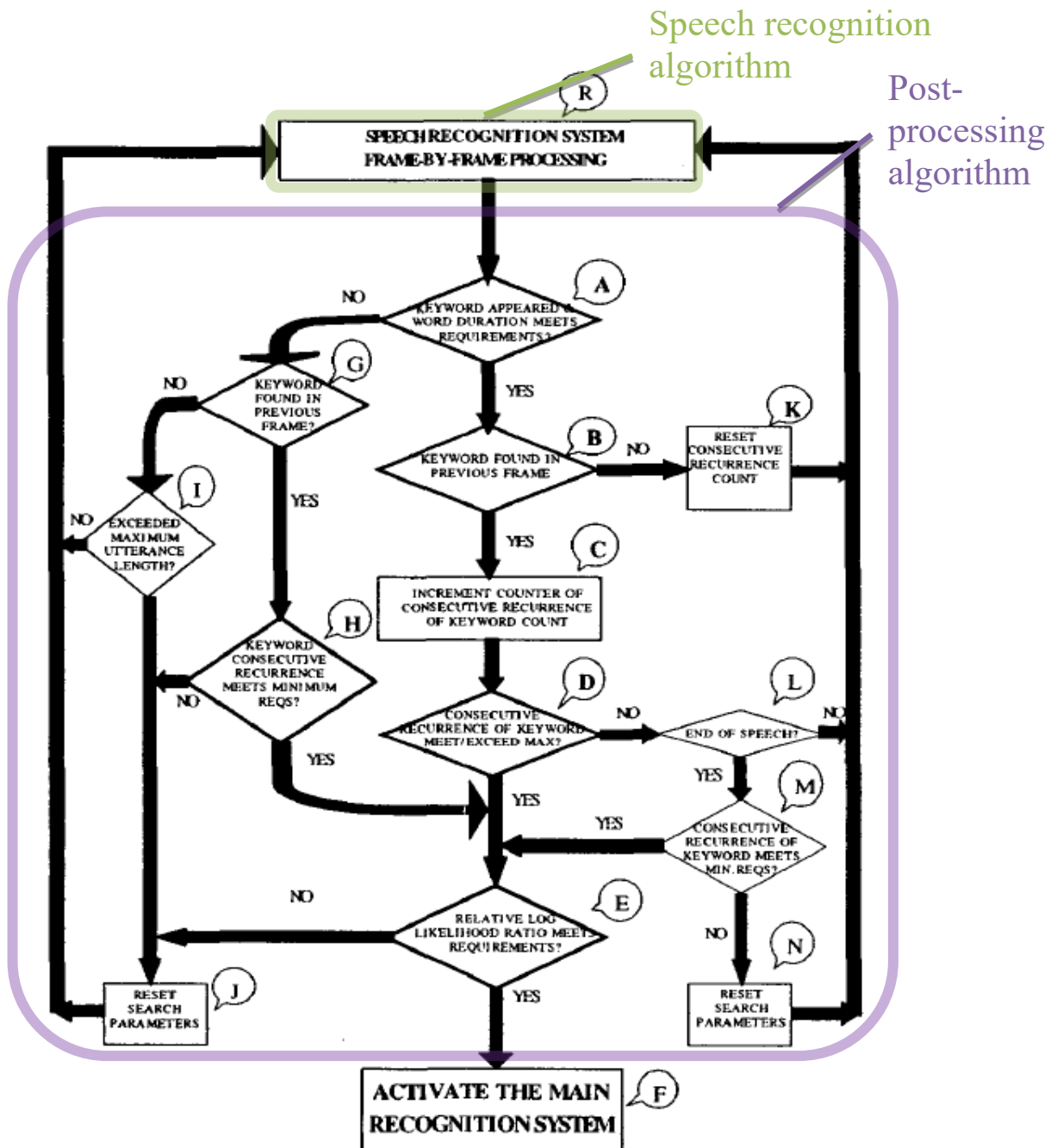


Figure 2. Flow graph of the post-processing algorithm

Ex.1005, Figure 2 (annotated).

81. Second, Bou-Ghazale’s voice activation system recognizes the activation command using “*a device*” because Bou-Ghazale teaches that its voice activation system “can be implemented in a **personal communication device**.” Ex.1005, Sec.5, 1738; *see also* Ex.1005, Title (“HANDS-FREE VOICE ACTIVATION OF PERSONAL COMMUNICATION **DEVICES**”), Abstract (“The proposed system is intended for turning a **device** on...upon recognizing the activation utterance”), Sec.1, 1735 (“[W]e propose using a voice activation system for turning the **device** on simply by speaking a pre-selected activation utterance.”) (“Upon recognizing that the activation command has been spoken, the voice activation system turns **the device** on”).

82. Thus, Bou-Ghazale’s teaching of a method for recognizing a pre-defined activation command spoken by a user using a voice activation system that can be implemented on a personal communication device renders obvious a “*method for recognizing a speech of a speaker by a device*.”

[1.1] *receiving, by a preset call command recognition unit, an input speech spoken by the speaker;*

83. First, Bou-Ghazale teaches that “[t]he voice activation system continuously **listens to and processes incoming speech**[,]” including speech spoken by “the **user**.” Ex.1005, Sec.1, 1735; *see also* Ex.1005, Abstract (“a system designed to continuously **listen to incoming unconstrained speech** and to recognize a pre-defined activation command”), Sec.1, 1735 (“The voice activation

system...continuously **processes incoming speech** as opposed to processing limited duration utterances.”), Sec.1, 1735 (“since **the user** is not required to **speak** within a specified period of time, the system ought to make a decision on a continuous basis whether the **desired word was spoken** or not”). Bou-Ghazale further explains that when there is speech activity, the input speech is sampled, filtered, and processed:

Whenever speech activity is detected, the speech recognition component processes the input speech on a frame-by-frame basis and makes a decision at every frame whether the activation word has been spoken.

Ex.1005, Sec.2, 1735.

The **input speech is sampled** at 8kHz, **then filtered** by a band-pass filter with a pass-band from 300 to 3300 Hz. **The filtered speech is processed** at 20 ms frame rate with a Hamming window of 30 ms. A total of 10 LPC-based cepstral parameters, 10 delta-cepstra and 1 delta-energy are computed for each frame.

Ex.1005, Sec.2.2, 1736. Thus, Bou-Ghazale’s teaching of the voice activation system listening to, sampling, and processing incoming speech renders obvious “*receiving...an input speech spoken by the speaker.*”

84. Second, Bou-Ghazale’s “voice activation system” that “continuously listens to and processes incoming speech” is the claimed *preset call command*

recognition unit because it is “designed to...**recognize a pre-defined activation command.**” Ex.1005, Abstract. As explained in more detail hereafter for [1.3], a “pre-defined activation command” renders obvious a “*preset call command*” because it is a command that is defined before speech is spoken and is recognized in speech in order to activate a device or subsequent speech recognition system. Further, Bou-Ghazale explains that the “voice activation system” is a component of a two-component system that includes the “voice activation system” and the “main speech recognition system.” Ex.1005, 1735 (“Upon recognizing that the activation command has been spoken, **the voice activation system** turns the device on or switches to the **main speech recognition system** to receive further commands from the user.”). A component is a “*unit*” according to the ’196 patent. Ex.1005, Sec.2-2.3, 1735-36; Ex.1001, 5:59-6:3 (“token passing **unit 431 is a component**”).

85. Thus, Bou-Ghazale’s voice activation system, which is a component for recognizing a pre-defined activation command (“*preset call command recognition unit*”), that continuously listens to, samples, and processes incoming speech spoken by a user, renders obvious “*receiving, by a preset call command recognition unit, an input speech spoken by the speaker.*”

86. In the event this limitation (or any other “*unit*” limitation) is construed as a means-plus-function limitation, a combination of hardware and software (Bou-

Ghazale’s personal communication device with processor executing speech recognition software—*see* [1.0] and [9.2]) carries out the functions recited in association with the “*preset call command recognition unit*” in [1.1]-[1.4]. This same alternative means-plus-function mapping applies to every “*unit*” term in Challenged Claims 1-20.

[1.2] *receiving, by the preset call command recognition unit, a subsequent input speech spoken, by the speaker, subsequently to the input speech;*

87. As explained with respect to limitation [1.1], Bou-Ghazale’s voice activation system renders obvious “*a preset call command recognition unit.*” The voice activation system also receives “*subsequent input speech*” because it “**continuously**” listens to and processes incoming speech. Ex.1005, Sec.1, 1735.

88. Bou-Ghazale teaches that **all** incoming speech is *received* by the voice activation system because it teaches that “[t]he voice activation system **continuously listens to and processes incoming speech.**” Ex.1005, Sec.1, 1735; *see also* Ex.1005, Abstract (“a system designed to continuously **listen to incoming unconstrained speech** and to recognize a pre-defined activation command”), Sec.1, 1735 (“The voice activation system...**continuously processes incoming speech** as opposed to processing limited duration utterances.”). Bou-Ghazale’s voice activation system receives speech that occurs before the activation word; it receives speech containing the activation word; and—relevant to this limitation—it

receives speech after the activation word (“*subsequent input speech*”) because it “continuously listens to and processes incoming speech.” Ex.1005, Sec.1, 1735. Further, Bou-Ghazale teaches that it captures subsequent speech any time after the activation word is spoken—regardless of whether there is silence following the activation word—, and within a 10-15 second interval before the main speech recognition system is deactivated. Ex.1005, Sec.3.2, Figure 2, 1736-37, Sec.5, 1738 (“If **no valid word is spoken in the following 10-15 seconds**, then the system switches back to the stand-by mode and waits to be activated again.”). For example, the subsequent input speech could occur immediately after the activation word without any intervening silence, as exemplified by path A, B, C, D, E, F in Figure 2 of Bou-Ghazale, in which the activation word has been recognized and the main speech recognition system has been activated, but the end of speech has not been reached. Ex.1005, Sec.3.2, Figure 2, 1736-37, Sec.5, 1738. Or, the subsequent input speech could occur after a period of silence, as exemplified by path A, B, C, D, L, M, E, F in which the “end of speech” has been reached and the main speech recognition system has been activated, but subsequent speech was not immediately received. Ex.1005, Sec.3.2, Figure 2, 1736-37, Sec.5, 1738. In either case, regardless of whether there is a delay after the activation word is spoken and before the subsequent input speech is spoken, it would have been obvious to a POSITA that one way in which Bou-Ghazale’s main speech recognition system

would receive the subsequent input speech would be through the voice activation system. Bou-Ghazale's voice activation system would first receive the subsequent input speech to, for example, utilize the feature extraction features of the speech recognition component to extract the "cepstral parameters" for the subsequent input speech before processing it using the main speech recognition system.

Ex.1005, Sec.3.2, Figure 2, 1736-37.

89. In more detail, Bou-Ghazale explains that the speech recognition component of its voice activation system performs an initial acoustic analysis on the input speech signal to derive a representation of the input speech in the form of a series of "cepstral parameters" (also called cepstral coefficients) for every 20 millisecond (ms) frame:

The input speech is sampled at 8kHz, then filtered by a band-pass filter with a pass-band from 300 to 3300 Hz. The filtered speech is processed at 20 ms frame rate with a Hamming window of 30 ms. A total of 10 LPC-based cepstral parameters, 10 delta-cepstra and 1 delta-energy are computed for each frame.

Ex.1005, Sec.2.2, 1736.

90. The cepstral parameters quantify the frequencies at which the energy in the speech signal is distributed—i.e., they are numerical representations that reflect how the speech "sounds." The "deltas" indicate how the speech signal

parameters change from one frame to the next—more specifically, they indicate how the distribution of energy in various frequencies is occurring in the speech signal. A representation of an input speech signal based on cepstral parameters is particularly well-suited for speech recognition using hidden Markov models (HMMs), which Bou-Ghazale utilizes: “The recognition system is frame-synchronous and is based on continuous density hidden Markov models (CD-HMM).” Ex.1005, Sec.2.2, 1736. Such a representation was commonplace for HMM-based speech recognition systems at the time of the ’196 patent and was not limited to systems for recognizing a single keyword. For example, Konopka and Lee both disclose performing feature extraction to derive a representation of input speech based on “cepstral coefficients”:

Feature extraction and signal conditioning consists of **extracting Mel-frequency cepstral coefficients (MFCC)**, delta information, and acceleration information.

Ex.1006, [0038].

The **recognizer feature vector** consists of 39 parameters: **12 Mel-warped frequency cepstra coefficients (MFCC)**, **12 delta cepstral coefficients**, 12 delta-delta cepstral coefficients, energy, and the delta and delta-delta of the energy.

Ex.1007, [0038]. This is consistent with the ’196 patent which uses a cepstral representation of speech based on “Mel Frequency Cepstrum Coefficients (MFC)

parameters,” where “Mel Frequency” refers to a method of biasing cepstral coefficients to more closely reflect how the ear perceives frequency. Ex.1001, 7:43-44.

91. As discussed below with respect to limitations [1.3]-[1.5], after Bou-Ghazale’s voice activation system determines that an initial “*input speech*” contains the activation word, a main speech recognition system (“*continuous speech recognition system*”) is activated for receiving and processing speech that occurs after the activation word—“further commands” (“*subsequent input speech*”). Ex.1005, Sec.1, 1735, Sec.5, 1738. Before the main speech recognition system processes such speech, a representation of the subsequent speech signal based on cepstral parameters (or coefficients) would be derived, as was well-known according to the examples described above and Bou-Ghazale’s disclosure. Since Bou-Ghazale’s voice activation system already includes feature extraction features for deriving such a representation, it would have been obvious to a POSITA to utilize these already-existing features of the voice activation system on all incoming speech. Accordingly, it would have been obvious for Bou-Ghazale’s voice activation system, specifically its speech recognition component to receive speech that occurs after the activation word to, for example, extract its cepstral parameters before the main speech recognition system processes the subsequent input speech using its vocabulary models, grammar, and evaluation criteria.

92. This is consistent with the '196 patent, which demonstrates speech 410 arriving at the call command recognition network 420, which includes an adaptive filter 420 and feature extraction 422. Ex.1001, Fig. 4. The output of 422 is used by the keyword searching unit 423. Fig. 4 does not show a separate speech input into the continuous speech recognition network 430, which includes its own keyword spotting unit 432. Instead, the extracted features derived in the call command recognition 420 using the adaptive filter 421 and feature extraction 422 would also be utilized in the continuous speech recognition network 430, as it does not have its own feature extraction component.

93. Accordingly, Bou-Ghazale's voice activation system "continuously" listens to and processes all incoming speech and it would have been obvious that it does this at least because it includes feature extraction features for deriving a representation of input speech (like the '196 patent's call command recognition network 420 which includes feature extraction features). It would have therefore been obvious to a POSITA that one way in which Bou-Ghazale's main speech recognition system receives "further commands" is through the voice activation system.

94. Because Bou-Ghazale's system "continuously" listens to and processes all incoming speech spoken by a user including speech spoken subsequent the activation command ("*subsequent input speech*"), Bou-Ghazale

renders obvious “*receiving, by the preset call command recognition unit, a subsequent input speech spoken, by the speaker, subsequently to the input speech.*”

[1.3] comparing, by the preset call command recognition unit, the input speech with a preset call command;

95. First, with respect to the “*preset call command*,” Bou-Ghazale describes a “**pre-defined activation command**” to be recognized by the voice activation system (“*preset call command recognition unit*”):

Voice activation in this context refers to a system designed to continuously listen to incoming unconstrained speech and to recognize a **pre-defined activation command** while rejecting any other extraneous speech or noise.

Ex.1005, Abstract.

96. An object of Bou-Ghazale’s system is to activate a device or speech recognition system “simply by speaking a **pre-selected utterance.**” Ex.1005, Sec.1, 1735 (“[W]e propose using a voice activation system for turning the device on simply by speaking a pre-selected activation utterance.”). With respect to prior communication systems, Bou-Ghazale explains:

Current personal communication systems such as a cellular phone with integrated speech recognition capabilities require the user to manually activate the device or the speech recognition system by pushing a button before the user can issue voice commands to control the device.

Ex.1005, Sec.1, 1735. Use of a “pre-defined activation command” (also called a “pre-selected utterance”) as taught by Bou-Ghazale prevents a user from having to “manually” activate a device. *Id.* Instead, Bou-Ghazale’s system uses a specific command that is defined before speech is spoken to activate the device or speech recognition system:

Upon recognizing that the **activation command** has been spoken, the voice activation system turns the device on or switches to the main speech recognition system to receive further commands from the user.

Id. For example, Bou-Ghazale uses the activation word “wake-up.” Ex.1005, Sec.4.1, 1737 (“the specific activation utterance is chosen as the word ‘**wake-up**’”), Sec.4.3, 1737 (“The test data containing the **activation word ‘wake-up’** consisted of 293 isolated utterances and 47 sentences.”), Sec.4.4, 1737-38 (discussing a set of experiments for determining statistical parameters for the activation word “wake-up”). Thus, Bou-Ghazale’s teaching of a “pre-defined activation command”—also called the “pre-selected activation utterance,” “activation word,” “desired word,” and “keyword”³—renders obvious “a *preset call command*” because it is a command that is defined before speech is spoken

³ See *supra* note 3.

and is recognized in speech in order to activate a device or subsequent main speech recognition system.

97. Second, Bou-Ghazale teaches that its voice activation system (“*preset call command recognition unit*”) compares incoming speech to “a keyword model for the **desired activation word**” (“*preset call command*”) to determine whether the activation word has been spoken. Ex.1005, Sec.1, 1735. Bou-Ghazale explains that its voice activation system comprises a speech recognition component that determines whether the activation word has been spoken using frame-by-frame processing:

Whenever speech activity is detected, the speech recognition component **processes the input speech on a frame-by-frame basis** and makes a decision at every frame **whether the activation word has been spoken.**”⁴

Ex.1005, Sec.2, 1735.

The recognition algorithm **performs a trace-back at every frame**, and passes the result string with the highest likelihood score to the post-processing component.

Ex.1005, Sec.2.2, 1736.

⁴ In Bou-Ghazale, the terms “real-time speech recognition system” and “real-time speech recognition component” are used interchangeably.

98. To make a decision “whether the activation word has been spoken” at every speech frame, a POSITA would understand that the speech recognition component is “*comparing*” the input speech to the activation word. More specifically, Bou-Ghazale’s speech recognition component is comparing the input speech frames to a keyword model for the activation word. As explained with respect to [1.1], Bou-Ghazale’s speech recognition component receives input speech. Upon receiving input speech, Bou-Ghazale teaches that it extracts the acoustic “parameters” of the input speech on a frame-by-frame basis:

The **input speech is sampled** at 8kHz, then filtered by a band-pass filter with a pass-band from 300 to 3300 Hz. **The filtered speech is processed** at 20 ms frame rate with a Hamming window of 30 ms. **A total of 10 LPC-based cepstral parameters, 10 delta-cepstra and 1 delta-energy are computed for each frame.**

Ex.1005, Sec.2.2, 1736. Then, the acoustic parameters of the input speech are processed, on a frame-by-frame basis, using Bou-Ghazale’s finite state network, as shown in Figure 1 below. Ex.1005, Sec.2.2, 1736. Bou-Ghazale explains that its speech recognition component which “is based on continuous density hidden Markov models (CD-HMM),” uses a “finite state network [that] consists of optional silence and OOV models at the beginning and end, and a parallel network of the **activation keyword** and one OOV **model**.” Ex.1005, Sec.2.2, 1736. As is

well-known in the art of hidden Markov model (HMM)-based speech recognition, words in a speech recognizer's vocabulary are represented as a series of acoustic states, where each state represents an acoustic feature of the word (e.g., vowel, consonant, phoneme). During speech recognition, the acoustic parameters of the input speech at each frame are *compared* to the acoustic features in the states of a given word model to determine how similar the input speech is to the word associated with that model. Accordingly, when Bou-Ghazale's HMM-based speech recognition component processes the input speech, it is "*comparing*" on a frame-by-frame basis the acoustic parameters of the input speech to the acoustic features in the states of the keyword, silence, and OOV models shown in Figure 1 below and determining whether the frame is most similar to a feature of the activation word, silence, or an OOV word. Further, Bou-Ghazale teaches that the input speech frames are conveyed as a "sequence" or "string" of keyword, silence, and OOV frames indicating the most likely match from the keyword, silence, or OOV models to a given frame in the unknown input speech. Ex.1005, Sec.2.2, 1736.

99. Moreover, Bou-Ghazale teaches that as a result of comparing the input speech to the models in the network, the speech recognition component determines "**the result string with the highest likelihood score**". Ex.1005, Sec.2.2, 1736. The "result string with the highest likelihood score" is the string that most likely matches the unknown input speech, and therefore indicates that the

input speech likely matches (i.e., includes) the activation word. *See* Ex.1006, [0052] (explaining that it was known that the highest likelihood (Viterbi) score may correspond to the “best word sequence”).

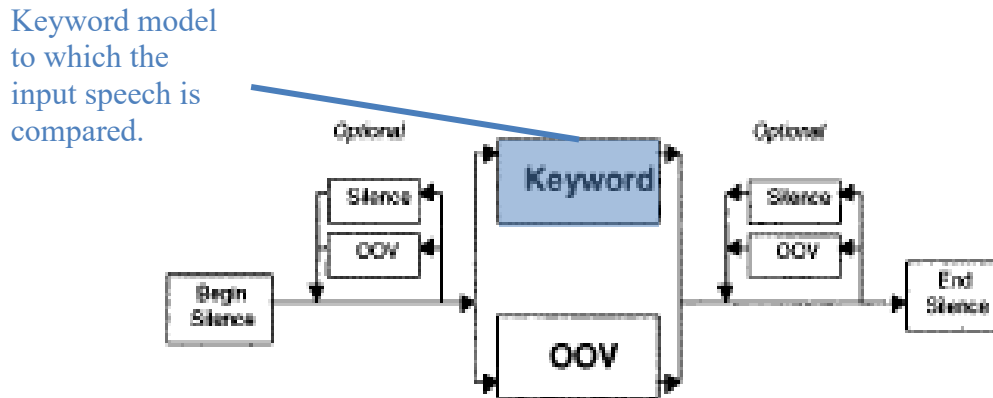


Figure 1. A finite state network representing the allowable word sequences.

Ex.1005, Figure 1 (annotated).

100. Accordingly, during the frame-by-frame processing illustrated in Figure 2 below, Bou-Ghazale’s hidden Markov model (HMM)-based speech recognition component *compares* the input speech to the activation word (“*preset call command*”), specifically a “keyword model for the desired activation word” shown in Figure 1 above. Ex.1005, Sec.1, 1735.

101. Konopka confirms that Bou-Ghazale’s HMM-based speech recognition component *compares* the input speech to the activation word’s keyword model:

The basic principle behind HMM-based speech recognition systems is that **an utterance is compared**

with a number of speech models...in order to find the model that best matches the utterance. This implies that the output of the speech recognition module 204 will be a reference to the model (e.g. word) with the best match.

Ex.1006, [0043].

Compares the input speech to the
keyword model for the activation
word

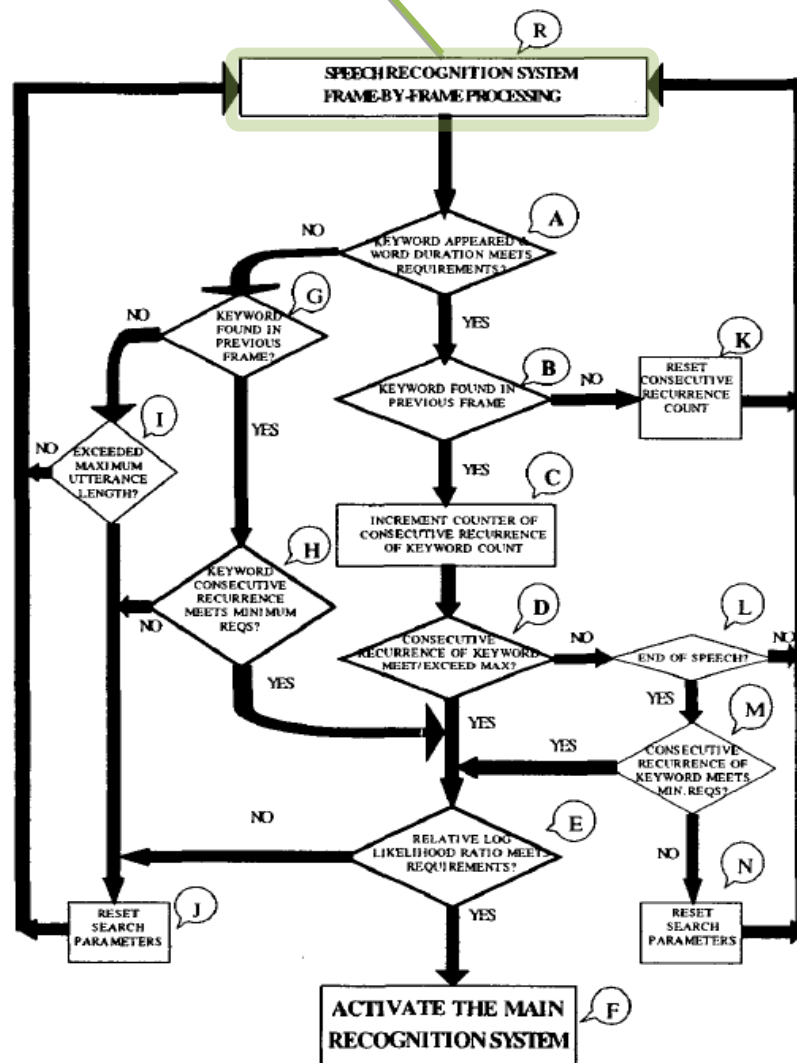


Figure 2. Flow graph of the post-processing algorithm

Ex.1005, Figure 2 (annotated).

102. Thus, Bou-Ghazale's teaching of an HMM-based speech recognition component performing frame-by-frame processing in which it compares incoming speech to a keyword model for a pre-defined activation word, such as the word

“wake-up,” renders obvious “*comparing, by the preset call command recognition unit, the input speech with a preset call command.*”

[1.4] *determining, by the preset call command recognition unit, whether the input speech is recognized as the preset call command according to a result of the comparison; and*

103. As explained in association with limitation [1.3], Bou-Ghazale’s speech recognition component compares input speech to an activation word’s keyword model. If the comparison indicates that it is likely that the activation word has been spoken (“*a result of the comparison*”), the result string (which is a representation of the input speech as described in [1.3]) is passed to a post-processing component to confirm the activation word has been recognized, as shown in Figure 2 below and described in more detail below.⁵ Ex.1005, Sec.2, 1735, Sec.2.3, 1736, Figure 2.

⁵ This is consistent with the ’96 patent which describes recognizing the input speech as the call command in two steps utilizing its token passing unit and its confidence detection unit. *See infra* notes 7 and 8.

Voice Activation System: Determines whether the input speech is recognized as the activation word.

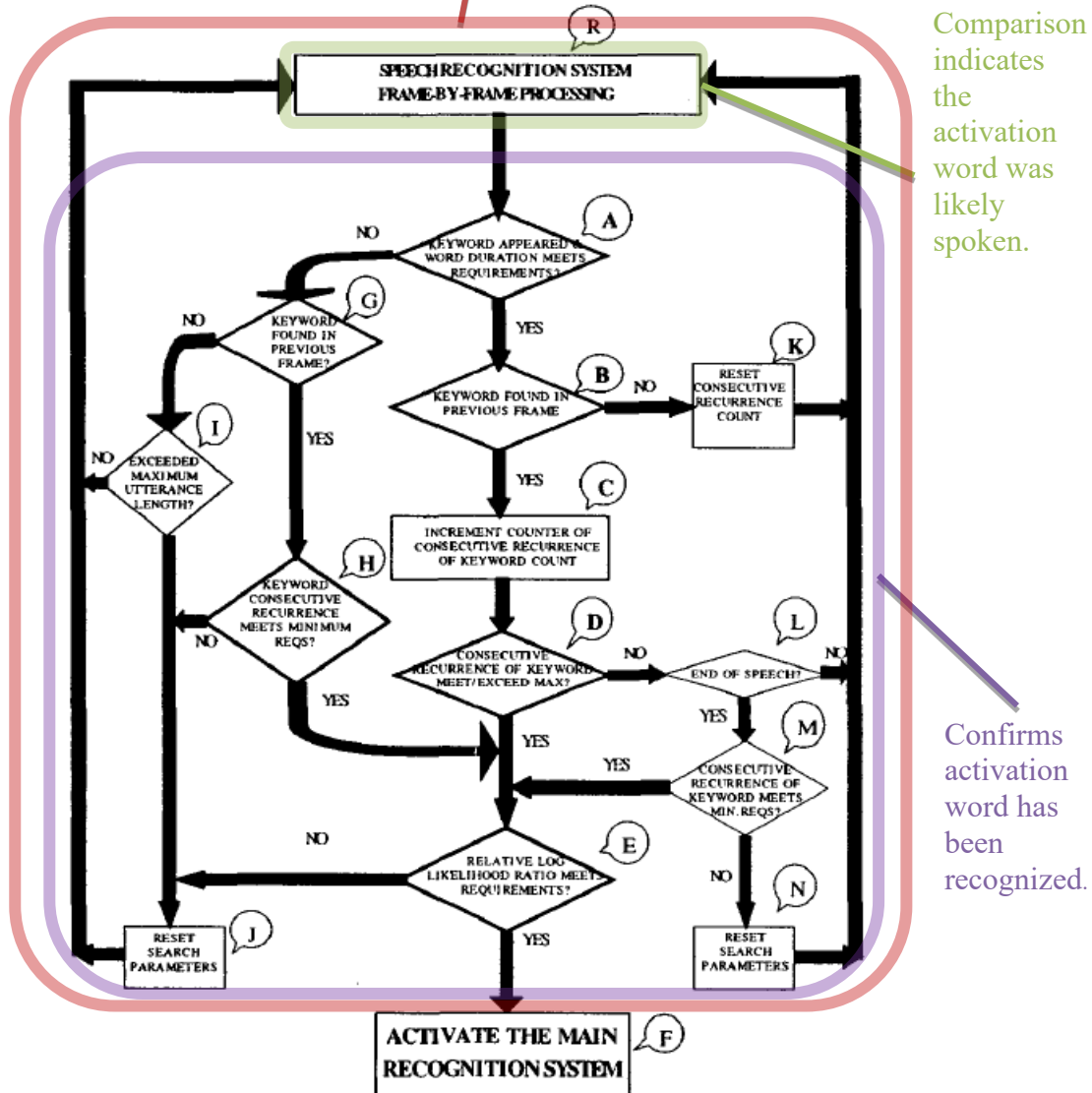


Figure 2. Flow graph of the post-processing algorithm

Ex.1005, Figure 2 (annotated).

104. First, as indicated by the green annotations in Figure 2 above, Bou-Ghazale teaches that “the **speech recognition component** processes the input speech on a frame-by-frame basis and **makes a decision** at every frame **whether the activation word has been spoken.**” Ex.1005, Sec.2, 1735. In particular, Bou-Ghazale teaches that its speech recognition component performs frame-by-frame processing during which it compares incoming speech to a keyword model for a pre-defined activation word, as explained above with respect to limitation [1.3]. Ex.1005, Sec.1-2, 2.2, 1735-36. As a “*result of the comparison*,” Bou-Ghazale’s speech recognition component determines “the **result string** with the highest likelihood score” and “passes [it] to the post-processing component.” Ex.1005, Sec.2.2, 1736; *see also* Ex.1006, [0043] (explaining that in HMM-based recognition systems “an utterance is compared with a number of speech models...in order to find the model that matches the utterance” and the output from this comparison “will be a reference to the model (e.g., word) with the best match”).⁶ The “result string with the highest likelihood score” reflects the string

⁶ This is consistent with the ’196 patent which explains that “every input speech frame is compared with a recognition network for only one word in time” (compare to Bou-Ghazale’s “keyword model”) and based on the tokens that have the highest accumulated probability (compare to Bou-Ghazale’s “result string with

that most likely matches the unknown input speech—it indicates whether and which frames of the input speech likely match (i.e., includes) the activation word. *See* [1.3], Ex.1006, [0052].

105. Moreover, as indicated by the purple annotations in Figure 2 above, Bou-Ghazale teaches that the post-processing component verifies (“validates”) whether the input speech matches the activation word utilizing the result from the speech recognition component:

The **post-processing component validates the recognition results** by applying a set of criteria and confidence measures.

Ex.1005, Abstract.

Whenever speech activity is detected, the speech recognition component processes the input speech on a frame-by-frame basis and makes a decision at every frame whether the activation word has been spoken. **The postprocessing component validates the speech recognition result.**

Ex.1005, Sec.2, 1735. More specifically, Bou-Ghazale teaches that the post-processing component uses the “result string with the highest likelihood score”

the highest likelihood score”) the token passing unit “assume[s] that an actual call command has been spoken.” Ex.1001, 6:55-57, 6:62-7:3, 7:29-39.

determined during the comparison in [1.3] (“*result of the comparison*”) and applies a set of criteria and measurements to verify whether the activation word was recognized in the input speech:

The **post-processing component** decodes the **result string** on a frame-synchronous basis and applies a set of criteria and confidence measures **to decide whether the activation word was a part of the utterance.**

Ex.1005, Sec.2.3, 1736.

106. If the input speech—which is represented by the result string as discussed in [1.3]—meets all the criteria associated with the pre-defined activation word, then “**the activation word is declared to be found**” (i.e., *recognized*) in the input speech by the post-processing component⁷:

The activation word is declared to be found only after it meets a set of statistics derived from training data such as the overall duration of the keyword, the number of times

⁷ This is consistent with the ’196 patent which describes a confidence detection unit that determines that the call command is recognized based on whether the confidence rate of the input speech frames meets a threshold criteria (compare to Bou-Ghazale’s “criteria and confidence measures”): “When the confidence of each of the portions is greater than 0...the call command ‘Nepiya’ is recognized.”

Ex.1001, 8:28-38.

the keyword appears in the result string of consecutive frames, and the confidence level associated with the keyword.”).

Ex.1005, Sec.2.3, 1736. In particular, with reference to Figure 2 below, Bou-Ghazale explains:

In the path traversing A, B, C, D, E, F, the keyword has met all the necessary requirements, and hence **the post-processor declares that the activation word has been found.**

In the path A, B, C, D, L, M, E, F, end of speech has been reached and the keyword has met all the necessary requirements. Hence, **the post-processor declares that the activation word has been found.”**

Ex.1005, Sec.3.2, 1737.

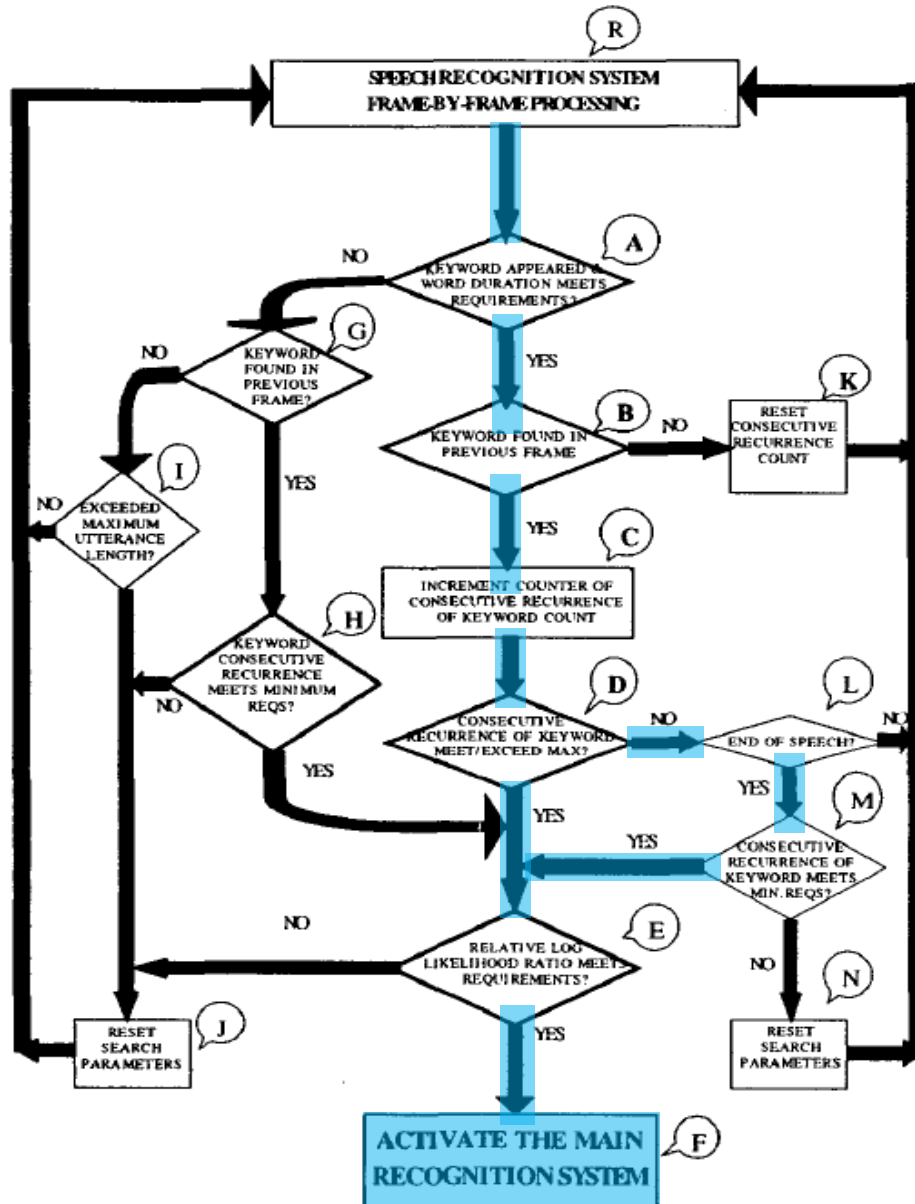


Figure 2. Flow graph of the post-processing algorithm
Ex.1005, Figure 2 (annotated).

107. Thus, Bou-Ghazale's voice activation system determining whether the activation word has been recognized in the input speech based on (i) the result string that represents the most likely keyword match to the unknown input speech as determined during the comparison step of limitation [1.3] (*"result of the*

comparison”) and (ii) whether the input speech meets the criteria and confidence measurements associated with the activation word renders obvious “*determining, by the preset call command recognition unit, whether the input speech is recognized as the preset call command according to a result of the comparison.*”

[1.5] *based on determining that the input speech is recognized as the preset call command, activating a continuous speech recognition unit configured to determine whether the subsequent input speech is recognized as an actual command, in order to reduce computational load on the device; and*

108. First, Bou-Ghazale teaches that once the activation word is declared as found in the input speech (“*based on determining that the input speech is recognized as the preset call command*”), a main speech recognition system is activated (“*activating a continuous speech recognition unit*”), as shown by Block F in Figure 2 below.

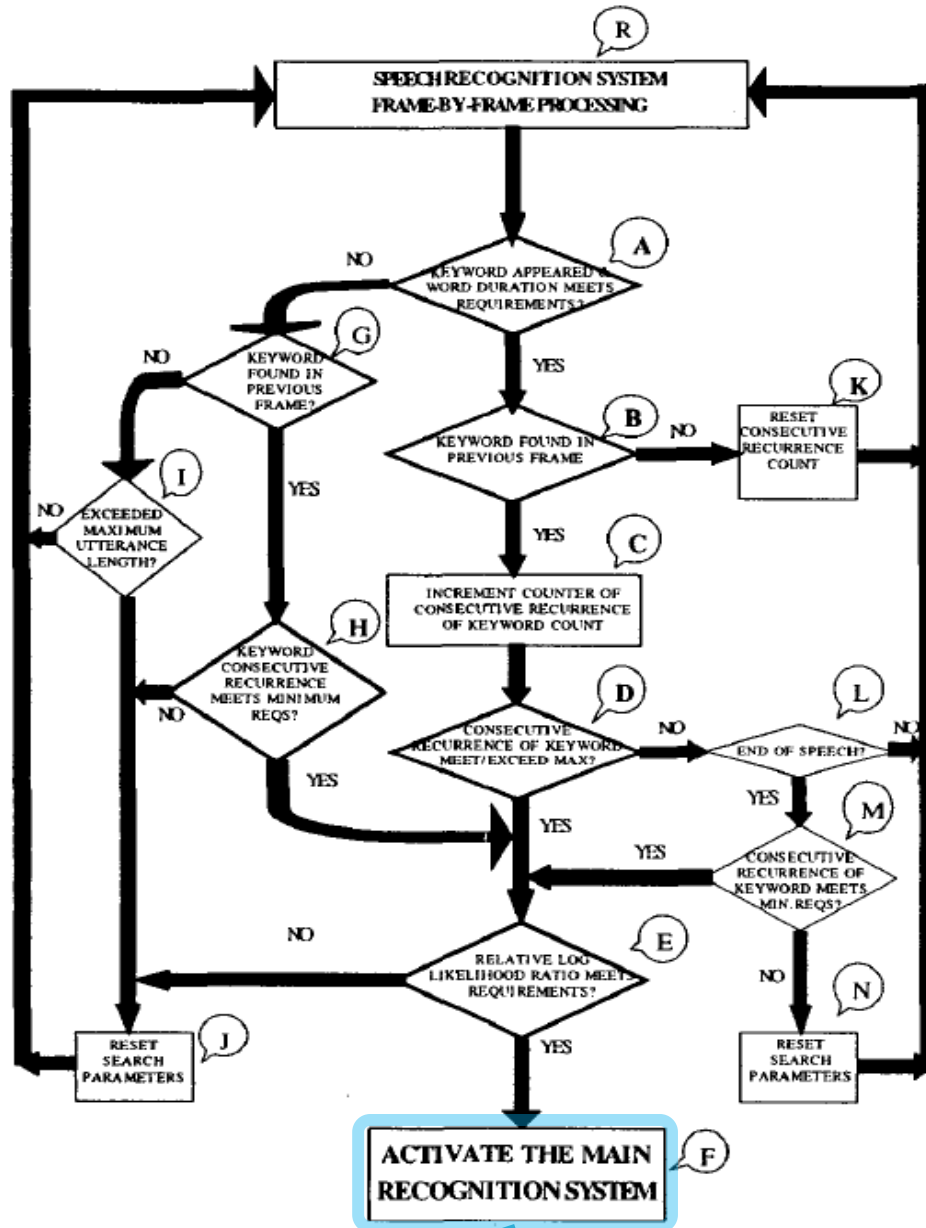


Figure 2. Flow graph of the post-processing algorithm

After the activation word is declared found, the main speech recognition system is activated.

Ex.1005, Figure 2 (annotated).

109. As explained with respect to limitation [1.4], Bou-Ghazale's post-processing component determines whether the activation word has been found in

the input speech based on whether the input speech (represented by the result string) meets the criteria and confidence measurements associated with the activation word. Ex.1005, Sec.2.3, 1736.

The **post-processing component** decodes the **result string** on a frame-synchronous basis and applies a set of criteria and confidence measures **to decide whether the activation word was a part of the utterance.**

Ex.1005, Sec.2.3, 1736. Bou-Ghazale explains that if the input speech meets the criteria and confidence measurements, then the activation word is declared to be found (i.e., recognized) (“*determining that the input speech is recognized as the preset call command*”):

The activation word is declared to be found only after it meets a set of statistics derived from training data such as the overall duration of the keyword, the number of times the keyword appears in the result string of consecutive frames, and the confidence level associated with the keyword.

Ex.1005, Sec.2.3, 1736. Bou-Ghazale further explains that once the activation word is declared found (or “recognized”), then a main speech recognition system is activated (“*activating a continuous speech recognition unit*”):

[T]he voice activation system can be designed in such a way that **once the activation word is recognized, the device** turns on or **switches to the main recognition**

system and waits to be prompted by a word from the main menu of the speech recognition system.

Ex.1005, Sec.5, 1738.

Upon **recognizing that the activation command has been spoken**, the voice activation system turns the device on or **switches to the main speech recognition system** to receive further commands from the user.

Ex.1005, Sec.1, 1735.

110. More specifically, in Figure 2 above, Bou-Ghazale shows that “[t]here are three paths...that meet all the requirements and hence result in **activating the main speech recognition system.**” Ex.1005, Sec.3.2, 1737. As one example, in the path A, B, C, D, E, F, the input speech has satisfied the criteria for the activation word, the activation word is declared found (i.e., recognized), and the main speech recognition system is activated at Block F. Ex.1005, Sec.3.2, 1737 (“[i]n the path traversing A, B, C, D, E, F, the keyword has met all the necessary requirements, and hence **the post-processor declares that the activation word has been found**”).

111. Thus, Bou-Ghazale’s teaching of activating a main speech recognition system, based on the determination that the activation word has been found in the input speech when it meets the statistical criteria for the activation word, renders

obvious “*based on determining that the input speech is recognized as the preset call command, activating a continuous speech recognition unit.*”

112. Second, Bou-Ghazale’s main speech recognition system is the claimed *continuous speech recognition unit* because, when activated, it continuously listens and “waits” to receive valid command words for a period of time before deactivating:

[O]nce the activation word is recognized, the device turns on or **switches to the main recognition system and waits to be prompted by a word from the main menu of the speech recognition system. If no valid word is spoken in the following 10-15 seconds,** then the system switches back to the stand-by mode and waits to be activated again.

Ex.1005, Sec.5, 1738. Bou-Ghazale’s main speech recognition system is “*continuous*” in the same manner as the ’196 patent’s “*continuous speech recognition system*”—both continuously recognize user commands only when activated. See Ex.1001, 8:32-41 (explaining that when “the call command ‘Nepiya’ is recognized...speech spoken subsequent to the call command is transferred to the continuous speech recognition engine to perform continuous speech recognition” and when “it is determined that the call command is not spoken...the process returns to the initial state.”).

113. Further, Bou-Ghazale explains that the “main speech recognition system” is a component of a two-component system that includes the “voice activation system” and the “main speech recognition system.” Ex.1005, 1735 (“Upon recognizing that the activation command has been spoken, **the voice activation system** turns the device on or switches to the **main speech recognition system** to receive further commands from the user.”). A component is a “*unit*” according to the ’196 patent. Ex.1005, Sec.2-2.3, 1735-36; Ex.1001, 5:59-6:3 (“token passing **unit 431 is a component**”).

114. Third, Bou-Ghazale’s main speech recognition system (“*continuous speech recognition unit*”) is configured to determine whether speech received after the activation word (“*subsequent input speech*”) is recognized as a word from the main menu of the speech recognition system (“*actual command*”). Bou-Ghazale explains that after the activation command is recognized as spoken, the device switches to the main speech recognition system to receive speech subsequent the activation command:

Upon recognizing that the activation command has been spoken, the voice activation system turns the device on or switches to the main speech recognition system to **receive further commands** from the user.

Ex.1005, Sec.1, 1735. Once the device has switched to (i.e., activated) the main speech recognition system, it waits for a word from the main menu of the speech recognition system:

[T]he voice activation system can be designed in such a way that once the activation word is recognized, the device turns on or switches to the main recognition system and **waits to be prompted by a word from the main menu of the speech recognition system.**

Ex.1005, Sec.5, 1738. The main speech recognition system either (i) determines that the further command is one of the words from the main menu of the speech recognition system (“*actual command*”), or (ii) determines that a valid command was not spoken (i.e., due to silence or speech that doesn’t match a word in its main menu):

If no valid word is spoken in the following 10-15 seconds, then the system switches back to the stand-by mode and waits to be activated again.

Ex.1005, Sec.5, 1738. Thus, Bou-Ghazale’s teaching of a main speech recognition system that receives further commands and determines whether they are a word from the main menu of the speech recognition system renders obvious “*a continuous speech recognition unit configured to determine whether the subsequent input speech is recognized as an actual command.*”

115. Fourth, Bou-Ghazale teaches that the main speech recognition system (“*continuous speech recognition unit*”) only activates when the activation word is found “*in order to reduce the computational load on the device.*” As discussed further above in this section, Bou-Ghazale explains that “the activation word is declared to be found **only after** it meets a set of statistics.” Ex.1005, Sec.2.3, 1736. If the input speech meets the criteria and confidence measurements (“statistics”), then the activation word is declared to be found and the main speech recognition system is activated, but if the input speech does not meet some of the criteria and confidence measurements, then the post-processing component will not activate the main speech recognition system and the search for an activation word at will start again (as shown by the highlighted paths in Figure 2 below):

There are three paths in the flow diagram of Figure 2 that meet all the requirements and hence result in activating the main speech recognition system. **The remaining side paths will either terminate the search and start over again,** or will continue the search since a theory is developing.

Ex.1005, Sec.3.2, Figure 2, 1736-37.

116. A POSITA would understand that postponing activation of the main speech recognition system unless and until the statistical criteria is met and the activation word is recognized would result in less computational load because the main speech recognition system requires a greater computational load compared to

the voice activation system. As Bou-Ghazale teaches, the main speech recognition system has a larger vocabulary (“menu”) of words that it is programmed to recognize:

In addition, the voice activation system can be designed in such a way that once the activation word is recognized, the device turns on or switches to the main recognition system and waits to be prompted by a word from **the main menu of the speech recognition system**.

Ex.1005, Sec.5, 1738.

117. Therefore, the main speech recognition system must compare incoming speech to several in-vocabulary word models for each word in its main menu. Meanwhile, the voice activation system need only compare incoming speech to one, in-vocabulary model (i.e., the “keyword model for the desired activation word”). Ex.1005, Sec.1, 1735, Sec.2.2 (“The finite state network consists of optional silence and OOV models at the beginning and end, and a parallel network of **the activation keyword** and one **OOV model**.”), Figure 1, 1736. A POSITA would have understood that comparing incoming speech against a single, keyword model requires less computational load than comparing incoming speech against several word models for the words in the main speech recognition system’s “menu”. Accordingly, compared to systems that have a subsequent speech recognition component activated at all times or that use a single,

speech recognition system with a large vocabulary to process all incoming speech, Bou-Ghazale's system "*reduce[s] the computational load on the device*" because it uses a smaller vocabulary voice activation system and only activates its larger vocabulary main speech recognition system when the activation word is found.

If the statistical criteria is not met, then the voice activation system will not activate the main speech recognition system.

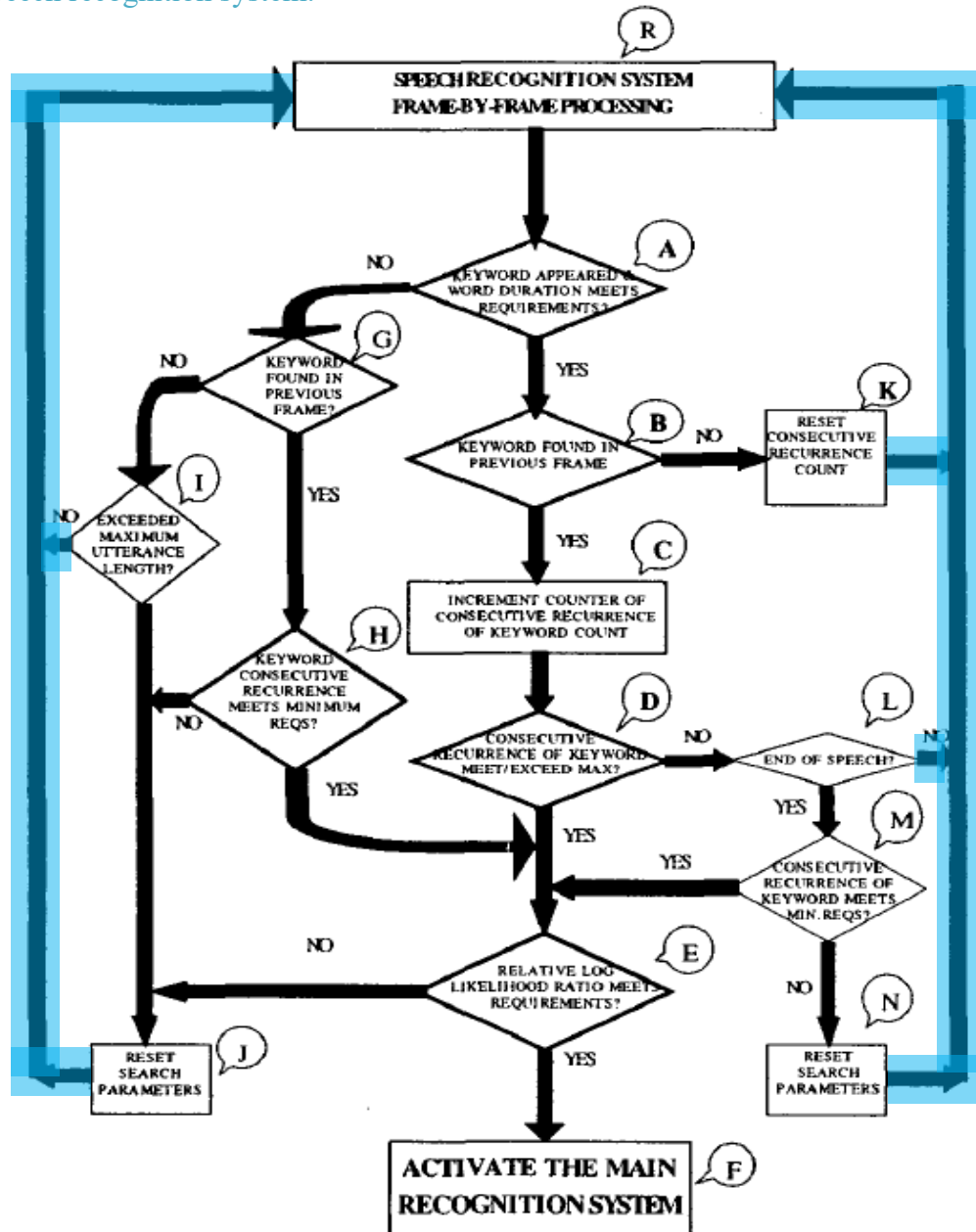


Figure 2. Flow graph of the post-processing algorithm
Ex.1005, Figure 2 (annotated).

118. Moreover, Bou-Ghazale explains that its voice activation system
“requires relatively low power and can be implemented in a personal

communication device due to its **low memory and MIPS requirements (< 1 MIPS).**” Ex.1005, Sec.5, 1738. It was well-known in the art at the time of the ’196 patent that “MIPS” is an acronym for “Million Instructions Per Second” and it refers to a measurement of a processor’s speed. Ex.1012, 63 (defining “MIPS (million instructions per second)” as “a measure of the speed of CPUs” where a CPU is a “central processing unit”); Ex.1013, 129 (defining “Million Instructions Per Second (MIPS)” as “a rate of speed by which data is **processed** or by which a computer performs”); Ex.1014, 140 (defining “MIPS” as “Millions of instructions per second; a measure of **processor speed**”). Accordingly, Bou-Ghazale’s voice activation system that requires a lower processing speed, renders obvious that it is designed to have a lower computational load to process.

119. Thus, because Bou-Ghazale teaches that (i) once the activation word is determined to have been found in the input speech, a main speech recognition system is activated for determining whether subsequent speech matches a word in the main speech recognition system, and (2) Bou-Ghazale’s system requires less power and processing speed because it uses a smaller vocabulary voice activation system and the larger vocabulary main speech recognition system only activates when the statistical criteria is met and the activation word is recognized, Bou-Ghazale renders obvious “*based on determining that the input speech is recognized as the preset call command, activating a continuous speech recognition unit*”

configured to determine whether the subsequent input speech is recognized as an actual command, in order to reduce computational load on the device.”

120. Moreover, the same alternative means-plus-function mapping discussed in [1.1] applies to the “*continuous speech recognition unit*.” In particular, a combination of hardware and software (Bou-Ghazale’s personal communication device with processor executing speech recognition software—*see* [1.0] and [9.3]) carries out the functions recited in association with the “*continuous speech recognition unit*” in [1.5].

[1.6] wherein the preset call command recognition unit comprises a token passing unit configured to assume the input speech to be the preset call command according to the result of the comparison using a minimum recognition network composed of a silence interval accompanied by noise and the preset call command.

121. Bou-Ghazale in view of Konopka renders obvious this limitation.

122. **First**, with respect to the “*token passing unit*,” Bou-Ghazale teaches that its voice activation system (“*preset call command recognition unit*”) comprises a speech recognition component.⁸ Ex.1005, Sec.2, 1735 (also explaining that “**the voice activation system is comprised of... a speech recognition algorithm**”). Bou-Ghazale’s “speech recognition component” is a “*unit*” as

⁸ See *supra* note 5.

claimed—a component is a “unit” according to the ’196 patent. Ex.1001, 5:59-6:3 (“token passing **unit 431 is a component**”).

123. Bou-Ghazale in view of Konopka renders obvious that Bou-Ghazale’s speech recognition component is the “*token passing unit*” as claimed. Bou-Ghazale explains that its speech recognition component (also called the “real-time speech recognition system,” or “recognition system” for short) “is based on continuous density hidden Markov models (CD-HMM).” Ex.1005, Sec.2.2, 1736; *see also* Ex.1005, Abstract (explaining that the proposed voice activation system is “composed of an **HMM-based continuous speech recognition algorithm**”). Further, Bou-Ghazale explains that its speech recognition component performs back-tracing to determine the best result string that most likely matches the unknown speech (i.e., the one with the highest likelihood score):

The recognition algorithm “performs a trace-back at every frame, and **passes the result string with the highest likelihood score** to the post-processing component.

Ex.1005, Sec.2.2, 1736.

124. Back tracing is a well-known speech recognition technique for determining the best state (or word) sequence that closely matches unknown speech, where there are multiple state (or word) sequences that the unknown speech is being compared to in a given word model. Ex.1006, [0052] (“To find the best state sequence, it is required to perform ‘back tracing.’”). As explained with

respect to limitation [1.3] above, words in an HMM-based speech recognizer's vocabulary are represented as a series of acoustic states, where each state represents an acoustic feature of the word (e.g., vowel, consonant, phoneme). During speech recognition, the acoustic parameters of the input speech at each frame are compared to the acoustic features in the states of a given word model and the best sequence of states—that is most similar to the unknown speech—is determined. However, in order to determine which acoustic word feature most likely corresponds to a given state, the state sequence must be traced backwards until the word sequence that reflects the most likely word match is determined. To facilitate back-tracing through the appropriate states, the “traditional” back-tracing approach requires storing back pointers in all states:

To find the best state sequence, it is required to perform “back tracing”. The traditional way of doing this is to let each state contain a pointer back to the previously best state for each frame. Back tracing can then be performed by following the pointers back, starting with the last model state for the state sequence with the highest Viterbi score. This means that if a system uses N states over T discrete time steps, the number of back pointers required is typically NT . This quickly becomes a high number and therefore leads to high memory requirements.

Ex.1006, [0052]; *see also* Ex.1021, Page 8 of 30 (explaining that “to actually retrieve the state sequence, we need to keep track of the argument which maximized (31) [(i.e., back-pointers)] for each t and j ” and to find the best state sequence we need to perform “path (state sequence) backtracking”).

125. A POSITA would have found it obvious for Bou-Ghazale’s HMM-based speech recognition component to use “token passing,” as taught by Konopka, to determine the result string that most likely matches the unknown speech (“the result string with the highest likelihood score”). Ex.1005, Sec.2.2, 1736. Konopka describes an approach to back-tracing that is different from the “traditional” approach: token passing is “based on the idea of passing ‘tokens’ around to the various states instead of allocating memory on a per-state basis.” Ex.1006, [0052]. Konopka teaches that **“token passing is a well-known approach to tracking the best word hypotheses through an HMM.”** Ex.1006, [0052]. Konopka explains that this **“token passing technique provides a highly memory efficient solution to an otherwise well-known problem in HMM-based speech recognition systems;** the storage of back-pointers that allows for finding the best word sequence hypothesis once all speech data has been processed.” Ex.1006, [0052]. In Konopka’s token passing technique, the speech decoder “looks back” to the previous state in the word model state sequence and determines whether it is similar (i.e., a “member of the same acoustic model”) to the current state:

[I]nstead of storing one token pointer in each state, the speech decoder 216 uses two arrays S₁ and S₂ to hold the token pointers for each state. Array S₁ keeps the token pointers for each state and the previous frame, and S₂ keeps the token pointers for each state and the current frame. **When each state i “looks back” to find the previously best state j, two things can happen.** If the previous best state j is a member of the same acoustic model as i, the token pointer for state j in S₁ is copied into position i in S₂. **If this is not the case, a new token is created and stored in position i in S₂.** The new token gets the same contents as token i in S₁, and in the token history, a reference to model m, i.e. m, is added. Once all states have been processed for the current frame, the pointers to structures S₁ and S₂, are swapped, and the process is repeated for the following frame.

Id. Accordingly, during the back tracing process using Konopka’s well-known token passing technique, token pointers corresponding to the sequence (or string) of states that are less likely to correspond to the unknown input speech are discarded at each state and the token pointer corresponding to the best state is kept and passed. Ex.1006, [0052] (If the previous best state j is not a member of the same acoustic model as the current state i “a new token is created and stored in position i in S₂.”). This token passing technique saves memory space compared to

the “traditional” back-tracing approach which requires storing back pointers in all states, as explained above. *See* Ex.1006, [0052].

126. For at least the reasons explained above in Section VII.A.3., a POSITA would have been motivated to implement Konopka’s token passing technique into Bou-Ghazale’s HMM-based speech recognition component. In particular, a POSITA would have been motivated to discard the strings that are less likely to match the unknown speech during the recognition process in Bou-Ghazale, in order to more efficiently and effectively determine the result string that most likely matches the unknown speech (“the result string with the highest likelihood score”) and pass it to the post-processing component.

127. Thus, Bou-Ghazale’s teaching of a voice activation system comprising an HMM-based speech recognition component in view of Konopka’s teaching of a token passing technique for use with HMM-based speech recognition systems renders obvious “*wherein the preset call command recognition unit comprises a token passing unit.*”

128. Second, Bou-Ghazale teaches that its speech recognition component (“*token passing unit*”) assumes the activation word is in the input speech according to the result string with the highest likelihood score (“*configured to assume the input speech to be the preset call command according to the result of the comparison*”). As explained above with respect to limitation [1.3], Bou-Ghazale

teaches that its speech recognition system compares incoming speech to “a keyword model for the **desired activation word**” (“*preset call command*”) to determine whether the activation word has been spoken. Ex.1005, Sec.1, 1735; *see also* Ex.1005, Sec.2.2, 1736. As explained with respect to limitation [1.4], as a “*result of the comparison*” Bou-Ghazale’s speech recognition component determines “the **result string with the highest likelihood score**” and “passes [it] to the post-processing component”. Ex.1005, Sec.2.2, 1736; *see also* Ex.1006, [0043] (explaining that in HMM-based recognition systems “an utterance is compared with a number of speech models...in order to find the model that matches the utterance” and the output from this comparison “will be a reference to the model (e.g., word) with the best match”).

129. According to the token passing technique taught by Konopka, as explained further above in this section, Bou-Ghazale’s “result string with the **highest likelihood score**” corresponds to the “**best** word sequence” that is determined to be closest to the unknown speech. Ex.1006, [0052], [0043]. Other POSITAs have confirmed that it is well-known in the art that a likelihood score corresponds to how similar uttered speech is to sequences in an acoustic model (e.g., phoneme model) for a target word (e.g., a voice code word or an activation word):

When an uttered speech vectors are entered into decoding stage, **likelihood scores are computed at all sequences of words** until the end-point of speech is found. Thus, **a likelihood score of phoneme models similar with uttered speech features become increased**, or else a likelihood score has low value. When a person speaks a word, spoken word can be a completely different word or **a similar word with a voice code word**.

Ex.1007, 30.

130. Therefore, in Bou-Ghazale, when the speech recognition component compares the input speech to the keyword model for the activation word (as shown in Figure 1), the likelihood score associated with the result string for the keyword model will be higher if the input speech is similar to the activation word, and thus, the speech recognition component will “*assume*” that the input speech is the activation word. Ex.1005, Sec.2.2, Figure 1, 1736. Accordingly, the “result string with the highest likelihood score” that is passed to the post-processing component would also convey that the input speech is “*assume[d]*” to include the activation word, as shown by the green annotations in Figure 2 below.

131. After the speech recognition component assumes the input speech to include the activation word, Bou-Ghazale teaches that the post-processing component verifies the assumption using criteria and confidence measurements: “The post-processing component decodes the **result string** on a frame-

Sec.2.3, 1736.

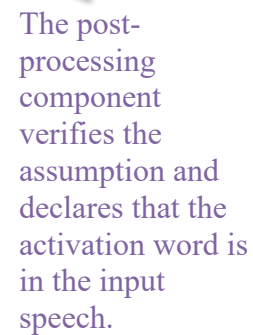


Figure 2. Flow graph of the post-processing algorithm

Once the activation word is declared found, the main recognition system is activated

Ex.1005, Figure 2 (annotated).

132. Thus, Bou-Ghazale’s teaching of a speech recognition component that assumes the activation word is in the input speech according to the result string with the highest likelihood score, as determined using the token passing technique of Konopka, renders obvious a “*token passing unit configured to assume the input speech to be the preset call command according to the result of the comparison.*”

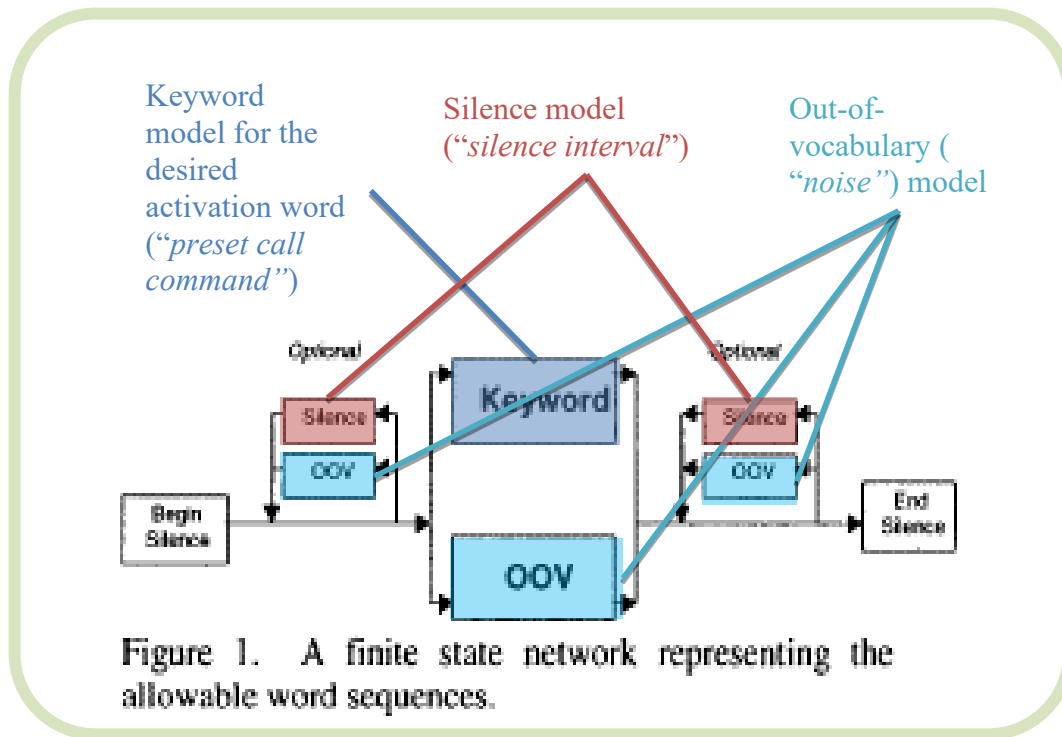
133. Third, Bou-Ghazale teaches that its speech recognition component (“*token passing unit*”) uses “*a minimum recognition network composed of a silence interval accompanied by noise and the preset call command.*” Specifically, Bou-Ghazale teaches that its speech recognition system uses a “finite state network” that includes a keyword model for the desired activation word (“*preset call command*”), a model for out-of-vocabulary words (“*noise*”), and a model for silence (“*silence interval*”):

The recognition system uses a finite state network, as shown in Figure 1, to represent the allowable sequences of silence, keyword, and out-of-vocabulary (OOV) words. The finite state network consists of optional silence and OOV models at the beginning and end, and a parallel network of the activation keyword and one OOV model.

Ex.1005, Sec.2.2, 1736; *see also id.* (listing examples of allowable word sequences, including “[Silence Keyword OOV Silence]” and “[Silence OOV Keyword OOV Silence]”). Bou-Ghazale’s finite state network is illustrated in

Figure 1 below.

Finite state network (*“minimum recognition network”*)



Ex.1005, Figure 1 (annotated).

134. A POSITA would recognize that Bou-Ghazale’s finite state network (shown in Figure 1 above) renders obvious a *“minimum recognition network”* because it is a network for recognizing one keyword, rather than a sequence of words. Ex.1005, Sec.2.2, 1736 (“the recognizer can detect **the word** whether it is spoken in isolation or embedded in extraneous speech”) (also explaining that “the finite state network consists of...a parallel network of **the activation keyword** and one OOV **model**”). In Figure 1 above, Bou-Ghazale shows only one keyword model in its finite state network. This is consistent with the example of a

“*minimum recognition network*” provided in the ’196 patent which recognizes “only one word”, or a “minimum” number of tokens, in recognizing a call command. Ex.1001, 6:60-63 (“Since the recognition network is a recognition network for only one word, a minimum number of (20 to 30) tokens are required.”).

135. A POSITA would also recognize that the out-of-vocabulary (OOV) models in Bou-Ghazale’s finite state network renders obvious “*noise*.” Bou-Ghazale explains that its voice activation system “can be realized by training a keyword model for the desired activation word, **a garbage model representing** both extraneous speech and **background noises**, and a model for background silence.” Ex.1005, Sec.1, 1735. Bou-Ghazale also explains that the out-of-vocabulary (OOV) models used in its finite state network “can be repeated many times to represent all **extraneous...speech** before or after the keyword, or the **utterances that do not include the keyword**.” Ex.1005, Sec.2.2, 1736.

Accordingly, Bou-Ghazale’s teaching of “background noises,” “extraneous speech,” and “utterances that do not include the keyword” discloses “*noise*.” This is also consistent with the ’196 patent which discloses that noise may be “spoken voices”:

The adaptive filter 421 is a Finite Impulse Response (FIR) filter for automatically tracking the spectrum characteristics of surrounding noise from **noise-**

containing input speech (all spoken voices including call commands) and eliminating the surrounding noise.

Ex.1001, 5:30-34.

136. Thus, Bou-Ghazale’s finite state network including OOV models (e.g., a garbage model) that represent extraneous speech, utterances that do not include the keyword, or background noises renders obvious a “*minimum recognition network...accompanied by noise.*”

137. In summary, (i) Bou-Ghazale’s teaching of a voice activation system comprising an HMM-based speech recognition component in view of Konopka’s teaching of an HMM-based token passing technique to determine the result string with the highest likelihood renders obvious “*wherein the preset call command recognition unit comprises a token passing unit,*” and (ii) Bou-Ghazale’s teaching that the speech recognition component “assumes” that the activation word is included in the input speech according to the result string with the highest likelihood score and using a finite state network for recognizing one word that consists of models for the keyword, for out-of-vocabulary words (e.g., extraneous speech and background noises), and silence, renders obvious “*a token passing unit configured to assume the input speech to be the preset call command according to the result of the comparison using a minimum recognition network composed of a silence interval accompanied by noise and the preset call command.*”

138. Moreover, the same alternative means-plus-function mapping discussed in [1.1] applies to the “*token passing unit*.” In particular, a combination of hardware and software (Bou-Ghazale’s personal communication device with processor executing speech recognition software—see [1.0] and [9.2]) carries out the functions recited in association with the “*token passing unit*” in [1.6], which is a component of the “*preset call command recognition unit*.”

b) Claim 2

[2.0] *The method of claim 1, wherein determining whether the input speech is recognized as the preset call command further comprises: computing, by a confidence detection unit, a confidence rate of the input speech; and*

139. First, with respect to “*a confidence detection unit*,” Bou-Ghazale teaches that the voice activation system (“*preset call command recognition unit*”) comprises a post-processing component. Ex.1005, Sec.2, 1735. The post-processing component renders obvious the claimed “*confidence detection unit*” because, as will be explained below, it verifies whether the activation word has been found in the input speech by determining (i.e., *detect[ing]*) a confidence rate for the input speech, called the “relative log-likelihood score.” Ex.1005, Abstract, 1735, Sec.2.3-3.1, 1736. Further, Bou-Ghazale’s “post-processing component” is a “*unit*” as claimed—a component is a “unit” according to the ’196 patent. Ex.1001, 5:65-67 (“the confidence detection **unit** 433 is a **component**”).

140. Second, Bou-Ghazale teaches that the post-processing component determines a “relative log-likelihood score” for the input speech (*“computing...a confidence rate for the input speech”*) to determine whether the input speech is recognized as the activation word. Ex.1005, Sec.2.3-3.1, 1736. As explained with respect to limitation [1.4], Bou-Ghazale’s post-processing component verifies whether the activation word has been found in the input speech (*“determining whether the input speech is recognized as the preset call command”*) based on whether the input speech meets the criteria and confidence measurements associated with the activation word. Ex.1005, Sec.2.3, 1736 (**“the post-processing component decodes the result string on a frame-synchronous basis and applies a set of criteria and confidence measures to decide whether the activation word was a part of the utterance”**), (**“The activation word is declared to be found only after it meets a set of statistics derived from training data such as...the confidence level associated with the keyword.”**). For example, for one such criteria illustrated by Block E in Figure 2 below, Bou-Ghazale teaches that its post-processing component determines a “relative log-likelihood score” (which Bou-Ghazale also refers to as a “relative log likelihood ratio” as shown in Block E of

Figure 2 below⁹) for the input speech and compares it to the “confidence level associated with the keyword,” where the confidence level is represented as a threshold value, η :

The activation word is declared to be found only after it meets a set of statistics derived from training data such as the overall duration of the keyword , the number of times the keyword appears in the result string of consecutive frames, and **the confidence level** associated with the keyword.

Ex.1005, Sec.2.3, 1736.

Criterion 6: The **relative log-likelihood score should be equal to or greater than a threshold, η** , given by: $\eta = (\text{Score}_{\text{keyword}} - \text{Score}_{\text{OOV}})/|\text{Score}_{\text{keyword}}|$, where the accumulated log-likelihood score for the keyword model is denoted by $\text{Score}_{\text{keyword}}$, and the log-likelihood score for the OOV model is denoted by $\text{Score}_{\text{OOV}}$.

⁹ In Bou-Ghazale, the terms “log-likelihood score” and “log likelihood ratio” are used interchangeably. For example, in Criterion 6, Bou-Ghazale explains that the “relative log-likelihood **score**” should be equal to or greater than a threshold.

Ex.1005, Sec.3.1, 1736. Meanwhile, in Block E of Figure 2, which is associated with Criterion 6, Bou-Ghazale asks whether the “relative log likelihood **ratio** meets requirements.” Ex.1005, Sec.3.2, Figure 2, 1736.

Ex.1005, Sec.3.1, 1736; *see also* Ex.1005, Sec.3.2, 1736 (explaining that “block E is associated with the criterion 6”). As demonstrated by the equation recited in Criterion 6 above, Bou-Ghazale’s relative log-likelihood score is determined by computing a log-likelihood score that the input speech corresponds to the keyword ($\text{Score}_{\text{keyword}}$), minus the log-likelihood score that the input speech corresponds to the OOV models ($\text{Score}_{\text{OOV}}$), scaled by the log-likelihood score that the input speech corresponds to the keyword. Ex.1005, Sec.3.1, 1736.

141. A POSITA would have understood that Bou-Ghazale’s “relative log-likelihood score” as defined above is a “*confidence rate*” as claimed because it is compared to another confidence—the “**confidence level** associated with the keyword” represented by the threshold value, η . Ex.1005, Sec.2.3-3.1, 1736. This is consistent with the ’196 patent which explains that a “confidence rate” is a “log-likelihood ratio”—determined by computing a log-likelihood that the input speech corresponds to the call command (“probability $p(0|\lambda_k)$ of phoneme models”), minus the log-likelihood that the input speech corresponds to the models for words that are not the call command, such as an out-of-vocabulary (OOV) word (“probability $p(0|\lambda_k)$ of anti-phoneme models, other than the assumed phoneme models”)— that is compared to a “threshold” value:

With respect to input speech (call command) assumed to be a call command ‘Naraeya’ or ‘Nepiya’ according to the recognized auto segmentation information, a ratio of the

probability $p(0|\lambda_k)$ of phoneme models for the call command recognition network to the probability $p(0|\lambda_k)$ of anti-phoneme models, other than the assumed phoneme models, is computed, and thus **the confidence rate of the call command is computed**. This is represented by the following Equation (2),

$$\text{LLR}_k(0, \lambda_k) = \log p(0|\lambda_k) - \log p(0|\lambda_k) \quad (2)$$

where **LLR is log likelihood ratio**, λ_k is a phoneme model, and λ_k is an anti-phoneme model.

When LLR is determined by Equation (2) and '**LLR>threshold**' is satisfied, the continuous speech recognition network is executed, otherwise the initial state is maintained.

Ex.1001, 7:63-8:13. Further, it was known by other POSITAs prior to the '196 that a log-likelihood ratio is used as a confidence measure and is compared to a threshold value to determine whether to reject speech as out-of-vocabulary or to recognize it as a target word (e.g., voice code word). For example, Lee teaches a two-step voice code verification process and explains that in the second step "[i]t computes a confidence measure that determines whether or not to reject the recognized voice code." Ex.1007, 30. Lee further explains that a confidence measure may be "a simple average of log-likelihood of all the syllables" in a sub-word, and a confidence measure is compared to a threshold:

For every **confidence measure**, a specific **threshold** is set up. If its value is below the threshold, the candidate is discarded from the verification task, resulting in a voice code verification failure.

Ex.1007, 36.

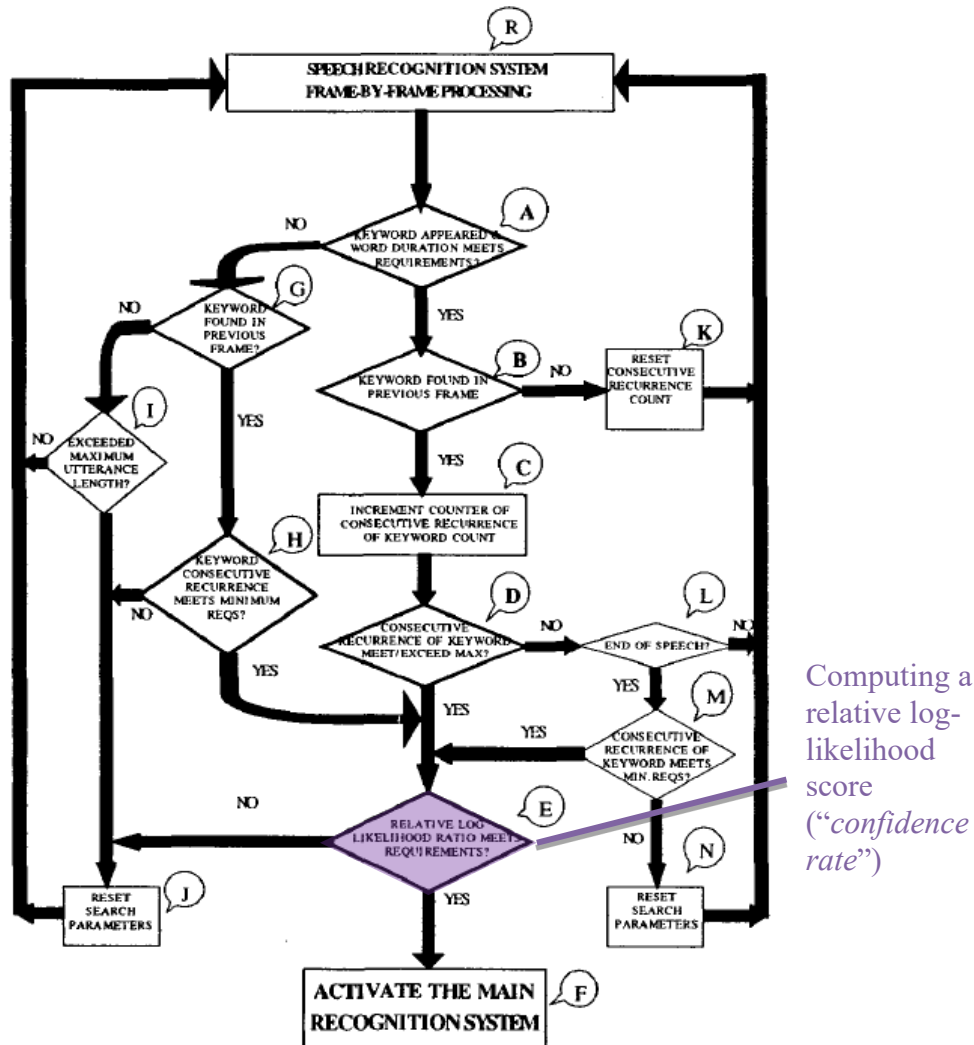


Figure 2. Flow graph of the post-processing algorithm

Ex.1005, Figure 2 (annotated).

142. Thus, Bou-Ghazale's teaching of determining whether the activation word has been found in the input speech by determining, using the post-processing

component of the voice activation system, a relative log-likelihood score—by computing a log-likelihood score that the input speech corresponds to the keyword, minus the log-likelihood score that the input speech corresponds to the OOV models, scaled by the log-likelihood score that the input speech corresponds to the keyword— that is compared to a threshold, η renders obvious “*wherein determining whether the input speech is recognized as the preset call command further comprises: computing, by a confidence detection unit, a confidence rate of the input speech.*”

143. Moreover, the same alternative means-plus-function mapping discussed in [1.1] applies to the “*confidence detection unit.*” In particular, a combination of hardware and software (Bou-Ghazale’s personal communication device with processor executing speech recognition software—*see* [1.0] and [9.2]) carries out the functions recited in association with the “*confidence detection unit*” in [2.0], which is a component of the “*preset call command recognition unit.*”

[2.1] *recognizing, by the confidence detection unit, the input speech as the preset call command based on the computed confidence rate.*

144. Bou-Ghazale teaches that the post-processing component (“*confidence detection unit*”) “declares that the activation word has been found” (i.e., *recogniz[ed]*) based on the relative log-likelihood score of the input speech (“*computed confidence rate*”). Ex.1005, Sec.3.2, Figure 2, 1737. Bou-Ghazale explains:

The activation word is declared to be found only after it meets a set of statistics derived from training data such as the overall duration of the keyword, the number of times the keyword appears in the result string of consecutive frames, and **the confidence level associated with the keyword.**

Ex.1005, Sec.2.3, 1736. More specifically, for one such statistic (or criteria), Bou-Ghazale explains:

Criterion 6: The **relative log-likelihood score should be equal to or greater than a threshold, η** , given by: $\eta = (\text{Score}_{\text{keyword}} - \text{Score}_{\text{OOV}}) / |\text{Score}_{\text{keyword}}|$, where the accumulated log-likelihood score for the keyword model is denoted by $\text{Score}_{\text{keyword}}$, and the log-likelihood score for the OOV model is denoted by $\text{Score}_{\text{OOV}}$.

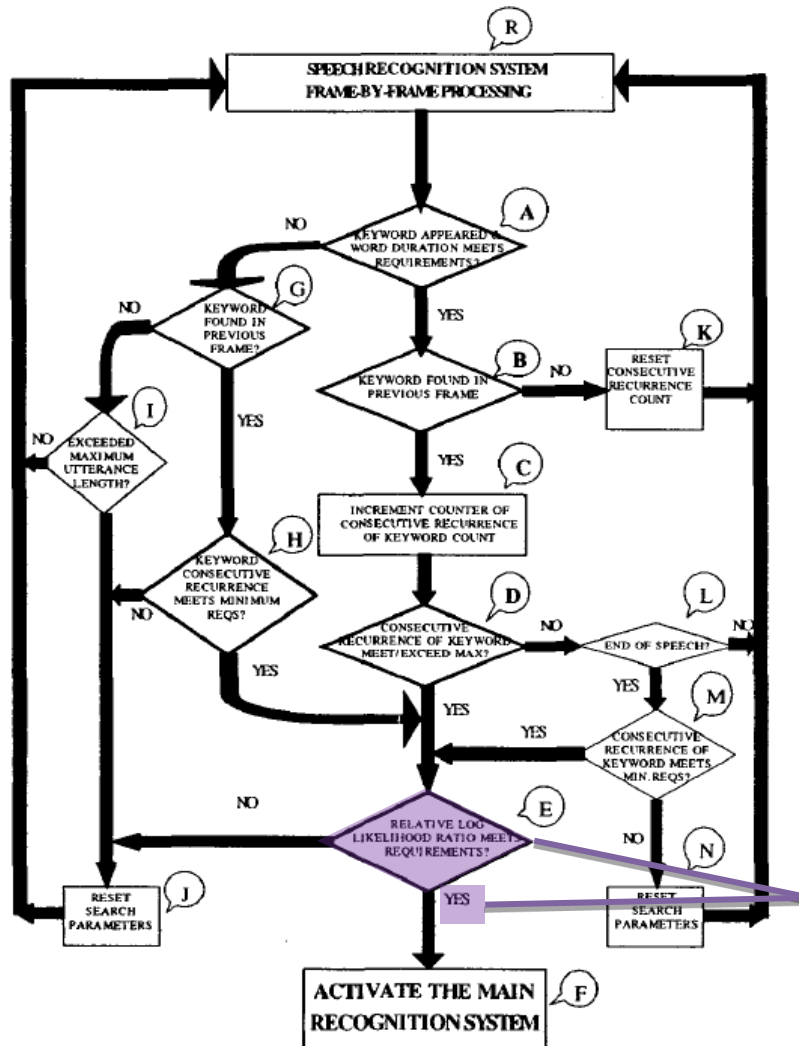
Ex.1005, Sec.2.3-3.1, 1736. The threshold, η , is the confidence level for the activation word and the relative log-likelihood score is the “*computed confidence rate*”, as explained above for limitation [2.0].

145. Based on the relative log-likelihood score (ratio¹⁰) (“*computed confidence rate*”) and whether it meets the threshold, η , the post-processing component will recognize the input speech as the activation word. For example, as shown in Figure 2 below, if the “**relative log likelihood ratio** meets requirements”

¹⁰ See *supra* note 10.

(“yes” path following Block E), then the post-processing component “**declares that the activation word has been found**” (i.e., “recognized”¹¹) and activates the main speech recognition system (Block F). Ex.1005, Sec.3.2, 1736-37 (“[i]n the path traversing A, B, C, D, E, F, the keyword has met all the necessary requirements, and hence **the post-processor declares that the activation word has been found**”), Sec.5, 1738 (“...**once the activation word is recognized**, the device turns on or switches to the main speech recognition system...”). This is consistent with the ’196 patent, which explains that “[w]hen LLR is determined by Equation (2) and ‘**LLR>threshold**’ is satisfied, **the continuous speech recognition network is executed**,” where Equation (2) of the ’196 patent is disclosed by Bou-Ghazale’s equation for determining the relative log-likelihood score (“*computed confidence rate*”) as discussed in [2.0] above and in more detail in [6.0]-[7.0] below. Ex.1001, 7:63-8:13.

¹¹ Bou-Ghazale also uses the term “recognized” to refer to the moment when the post-processing component “declares that the activation word has been found” before activating the main speech recognition system. Ex.1005, Sec.5, 1738 (“...**once the activation word is recognized**, the device turns on or switches to the main speech recognition system...”).



Declaring
("recognizing")
that the
activation word
is found based
on the relative
log-likelihood
score

Figure 2. Flow graph of the post-processing algorithm

Ex.1005, Figure 2 (annotated).

146. Thus, Bou-Ghazale's teaching of the post-processing component declaring that the activation word has been found (i.e., "*recogniz[ed]*") in the input speech based on the relative log-likelihood score ("*computed confidence rate*") renders obvious "*recognizing, by the confidence detection unit, the input speech as the preset call command based on the computed confidence rate.*"

c) Claim 3

[3.0] *The method of claim 2, wherein recognizing the input speech comprises: recognizing the input speech as the preset call command if the computed confidence rate is higher than a threshold value.*

147. As explained in the analysis of limitations [2.0] and [2.1], Bou-Ghazale's post-processing component determines a relative log-likelihood score (ratio¹²) ("*computed confidence rate*") and "declares that the activation word has been found" in the input speech if the relative log-likelihood score ("*computed confidence rate*") is "equal to or **greater than a threshold, η** ." Ex.1005, Sec.3.1-3.2, 1736-37, Figure 2 (showing that if the relative log-likelihood score defined in Criterion 6 is greater than the threshold value, then the "relative log likelihood ratio meets requirements" in the "yes" path following Block E, the activation word is recognized, and the main speech recognition system is activated), 1736-27, Sec.5, 1738.

148. Thus, Bou-Ghazale's teaching of the post-processing component declaring that the activation word has been found in the input speech if the relative log-likelihood score is greater than the threshold value, η renders obvious "*recognizing the input speech as the preset call command if the computed confidence rate is higher than a threshold value.*"

¹² See *supra* note 10.

d) Claim 4

[4.0] *The method of claim 2, wherein recognizing the input speech comprises: determining whether or not the input speech is an authentic call command based on the computed confidence rate; and*

149. As explained in the analysis of limitations [2.0] and [2.1], Bou-Ghazale teaches that its post-processing component (“*confidence detection unit*”) “declares that the activation word has been found” (i.e., recognized) based on whether the relative log-likelihood score of the input speech (“*computed confidence rate*”) is “equal to or greater than a threshold, η .” Ex.1005, Sec.3.1-3.2, 1736-37. Bou-Ghazale teaches that the presence of “false alarm[s]” and the threshold, η are related; specifically, the threshold, η may be adjusted to achieve lower “false alarm rates”—i.e., a lower number of commands that are falsely recognized as the activation word:

As the threshold value is increased, the false rejection rate increases and the false alarm rate decreases.... For achieving very low false alarm rates of 0.26% for isolated utterances and 0% for embedded utterances, η can be set to 0.007.

Ex.1005, Sec.4.4, 1738. A “false alarm” occurs when the system detects speech or noise and falsely recognizes it as an “*authentic call command*” (e.g., the activation word) and switches to the main speech recognition system. For instance, Bou-Ghazale explains that “in a false alarm situation, the user would not notice that the

system got activated since there is no prompt that's being played" rather "the system simply switches to the recognition mode for a short period of time" and then it'll switch back to "stand-by-mode" to wait for the activation word. Ex.1005, Sec.5, 1738; *see also* Ex.1007, 30 (explaining that "mis-recognized or out-of-vocabulary" speech and "noise" can generate false alarms). Bou-Ghazale's system lowers the number of false alarms that are recognized as "*authentic*" and result in the system being activated when it should not be, by adjusting its threshold, η . Ex.1005, Sec.4.4, 1738.

150. Because Bou-Ghazale describes false alarm rates as a "function of η ," a POSITA would have understood that when the post-processing component compares the relative log-likelihood score of the input speech to the threshold, η , it is determining whether or not the input speech is a "false alarm" or false command, such as noise or out-of-vocabulary speech. Ex.1005, Tables 2-3 (providing "false rejection and false alarm rates for the voice activation system as a **function of η** "); *see also* Ex.1007, 30 (explaining that "mis-recognized or out-of-vocabulary" speech and "noise" can generate false alarms). For example, this is illustrated by the inquiry shown in Block E in Figure 2 of Bou-Ghazale below:

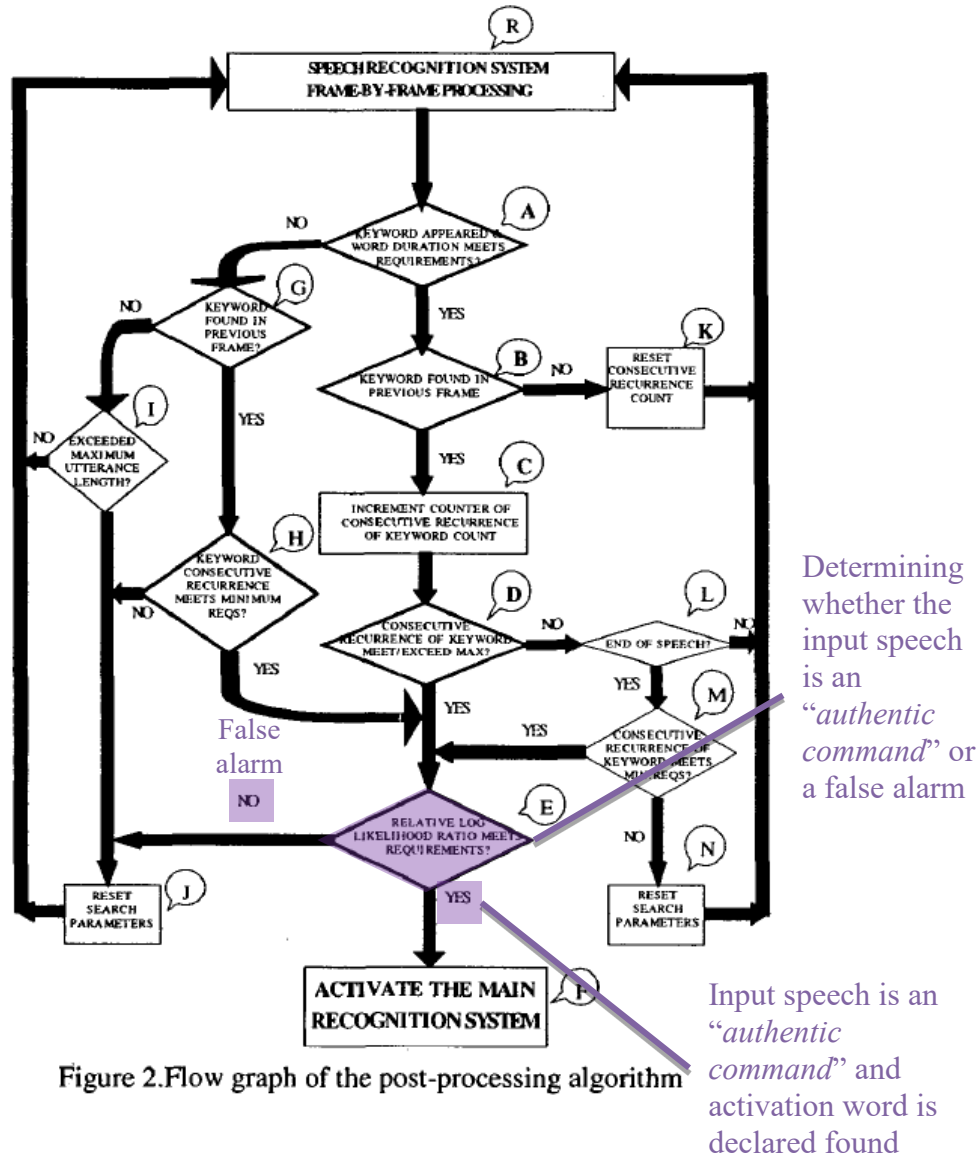


Figure 2. Flow graph of the post-processing algorithm

Ex.1005, Figure 2 (annotated).

If the relative log-likelihood score is “equal to or greater than the threshold, η ”, then the post-processing component determines that the input speech is an “*authentic command*” and not a false alarm (see “yes” path following Block E), and “declares that the activation word has been found.” Ex.1005, Sec.3.1-3.2, Figure 2, 1736-37. If the relative log-likelihood score does not meet the threshold

requirement, then the post-processing component will reject the speech as a false alarm (or false command) (see “no” path following Block E) and start the search for an activation word again. Ex.1005, Sec.3.1-3.2, Figure 2, 1736-37 (“There are three paths in the flow diagram of Figure 2 that meet all the requirements and hence result in activating the main speech recognition system. The remaining side paths will either **terminate the search and start over again...**”).

151. Thus, Bou-Ghazale’s teaching of determining whether or not the activation word is a false alarm and is found based on whether the relative log-likelihood score of the input speech meets a threshold, η renders obvious “*wherein recognizing the input speech comprises: determining whether or not the input speech is an authentic call command based on the computed confidence rate.*”

[4.1] *recognizing the input speech as the preset call command if it is determined that the input speech is the authentic call command.*

152. As explained above for limitation [4.0], Bou-Ghazale’s post-processing component determines whether the activation word is an “*authentic command*” or a false alarm based on whether the relative log-likelihood score of the input speech meets the threshold requirement, η , as shown by the inquiry in Block E of Figure 2 below. Ex.1005, Sec.3.1-3.2, 1736-37. If the relative log-likelihood score is “equal to or greater than the threshold, η ”, then the post-processing component determines that the input speech is an “*authentic command*” and not a false alarm (see “yes” path following Block E), and “**declares that the**

activation word has been found (“*recognizing the input speech as the preset call command*”). Ex.1005, Sec.3.1, 1736 (“Criterion 6: The relative log-likelihood score should be equal to or greater than a threshold, η ”), Sec.3.2, 1736-1737 (explaining that “block E is associated with the criterion 6” and “[i]n the path traversing A, B, C, D, E, **F**, the keyword has met all the necessary requirements, and hence **the post-processor declares that the activation word has been found**”).

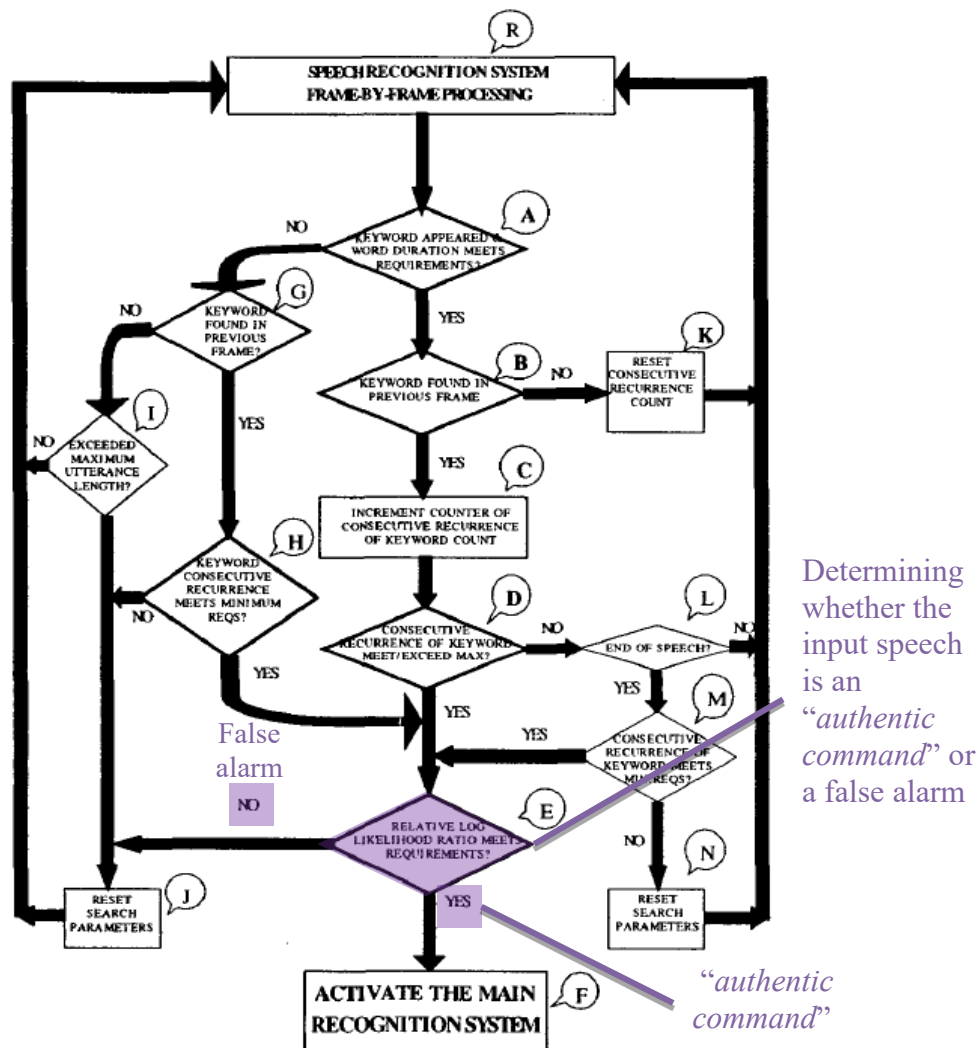


Figure 2. Flow graph of the post-processing algorithm

Ex.1005, Figure 2 (annotated).

153. Thus, Bou-Ghazale's teaching of the post-processing component declaring that the activation word has been found if it is determined that the input speech is not a false alarm because the relative log-likelihood score is greater than the threshold, η renders obvious "recognizing the input speech as the preset call command if it is determined that the input speech is the authentic call command."

e) Claim 5

[5.0] *The method of claim 1, wherein determining the input speech is recognized as the preset call command further comprises: computing a log likelihood ratio of the input speech; and*

154. First, as explained for limitation [1.4], Bou-Ghazale teaches that its voice activation system determines whether the activation word has been found (“*determining the input speech is recognized as the preset call command*”) in the input speech based on “the result string with the **highest likelihood score**” as determined by the speech recognition component during the comparison step of limitation [1.3]. Ex.1005, Sec.2.2, 1736. Thus, Bou-Ghazale teaches that its speech recognition component computes a “likelihood score” for the input speech (represented by the result string).

155. Second, with respect to the term “*log*,” to the extent Bou-Ghazale doesn’t explicitly state that its likelihood score as determined by the speech recognition component (“*token passing unit*”) is a “logarithm”, it would have been obvious to a POSITA because it was well-known to compute a logarithm of a likelihood score for a keyword model (i.e., in-vocabulary score) in order to determine the best text strings that most likely match the input speech (i.e., it was well-known to compute a “*log*” likelihood score for performing the “*assuming*” step described in [5.1] below). For example, Konopka teaches computing a logarithm (represented as a natural logarithm, $\ln$) of a Viterbi (likelihood) score

for an HMM model for an in-vocabulary word to determine the best text strings in the in-vocabulary word model:

Next, the speech recognition module 204 applies acoustic hidden Markov models (HMM) and an N-gram grammar to the incoming feature vectors (as specified by the natural language interface) to derive an **in-vocabulary (IV) Viterbi (likelihood) score** (Step 408). Then, the feature data is reprocessed using models of OOV events, e.g., an ergodic bank of monophone models, to derive an out-of-vocabulary (OOV) Viterbi score (Step 410). The garbage score is calculated from the IV and OOV scores, e.g., the garbage score equals $[\text{Ln(IV score)} - \text{Ln(OOV score)}] / \text{number of frames}$ (Block 411). A low score indicates a garbage utterance. **The N-best transcribed text string(s) and corresponding garbage score(s) are passed to the natural language interface module 222 (Step 412).** The natural language interface module 222 parses the incoming string(s) using a set of probabilistic context-free grammar (PCFG) rules as well as device context information for an attention utterance (Step 414).

Ex.1006, [0080]. In Konopka's process, steps 408-412 reflect the process by which Konopka's speech recognition module determines the best text strings (i.e., the strings with the "highest likelihood score") which are then passed for post-processing by the natural language interface module, similar to Bou-Ghazale's

speech recognition component which determines the “result string with the highest likelihood score” to be passed to its post-processing component. While in the equation shown above Konopka computes a logarithm of both an in-vocabulary score and an out-of-vocabulary score (and subtracts the two), Konopka explains that it was well-known to rely on only the score for the in-vocabulary model to determine the best text strings (“state sequence”):

[I]t is sometimes preferred to determine whether the utterance is contained within in-vocabulary words or not. For example, if a **Viterbi score passes a threshold, the utterance is determined to be an in-vocabulary word.** If the Viterbi score of the utterance does not exceed the threshold, then the utterance is deemed out-of vocabulary. Such a Viterbi score is generated using the Viterbi algorithm. **This algorithm calculates a single best state sequence through an HMM and its corresponding probability,** given an observation sequence.

Ex.1006, [0043]. In the above description, Konopka is referencing a Viterbi score for just the in-vocabulary model(s). Konopka goes on to explain that “another Viterbi score” for the out-of-vocabulary models may also be used. Ex.1006, [0043] (explaining that “many systems rely on comparing the Viterbi score with another Viterbi score” that is obtained using a model for out-of-vocabulary events in order to determine whether “the utterance had a closer match with the word models or

the out-of-vocabulary models” and improve rejection of out-of-vocabulary events). Konopka proposes using the second log-likelihood score for out-of-vocabulary models during its first speech recognition stage, but this second score could also be used during a second post-processing stage to verify the recognition results in the first stage, as Bou-Ghazale teaches consistent with the ’196 patent. *See* analysis for [6.0]-[7.0]; Ex.1005, Sec.3.1, 1736; Ex.1001, 7:63:8:15.

156. In summary, Konopka teaches that it was known to compute a “*log*” likelihood score for just the in-vocabulary model or for both the in-vocabulary and out-of-vocabulary model in the first stage of recognition (before the second post-processing stage), such that a POSITA would have understood that Bou-Ghazale’s “highest likelihood score” as determined by the speech recognition component may be a “*log*” likelihood score for just its keyword model.

157. Even without Konopka’s teaching, a POSITA would have recognized that computing the most likely recognized word would normally be done as a logarithm; this was common practice because it significantly reduces computational requirements for recognition software. A recognizer computes phoneme probabilities on a frame by frame (every 20 milliseconds) basis, and then computes the best path across frames based on the best matches on each frame. As the frames are considered to be statistically independent, the probability of a path would be found by *multiplying* the probabilities at each frame. But if the

probabilities are first converted to logs, the path probability is computed by *adding* the log probabilities. The best path based on log probability is identical to the best path based on raw probability, and addition is much less computationally intensive than multiplication, so use of log probability became common well before the '196 patent dates.

158. Third, with respect to the term “*ratio*,” Bou-Ghazale’s likelihood score as determined by the speech recognition component (log likelihood score for its keyword model, as modified by Konopka) is a “*ratio*” in the same way the '196 patent’s first “log likelihood ratio” is. Claim 5 of the '196 patent recites a “*log likelihood ratio*” which is used to perform the “*assuming*” step recited in [5.1]. The example provided in the specification for computing this “*log likelihood ratio*” is a logarithm function for a single model, which the specification refers to as a “ratio”:

First, the call command recognition network is configured (Step 1). The real-time recognition network for a recorded input speech frame is decoded and is then stored in tokens (Step 2). In this case, the individually stored bit values are sorted according to **the log likelihood ratio given by the following** **Equation,**

$$LLR_k(0, \lambda_k) = \log p(0 | \lambda_k) \quad (1)$$

where LL[R] is the log likelihood ratio, λ_k is a k-th model parameter, and $p(0 | \lambda_k)$ is the probability of the λ_k model of the input speech frame.

After the accumulated probabilities have been sorted at Step 2, token beam pruning is performed on N (20 to 30) tokens (Step 3). Then, in the final stage of the call command recognition network, **when the accumulated probability of a relevant token** at the moment at which the transition to the silence interval is made at Step 3 corresponds to the top 10% (2 to 3), **it is assumed that the call command has been spoken and the process proceeds to the confidence detection routine**, otherwise the process proceeds to Step 2 (Step 4).

Ex.1001, 7:5-24. Regardless of the terminology utilized in Bou-Ghazale and Konopka, Bou-Ghazale (as modified by Konopka) computes a logarithm function for a single model—a log likelihood score for its keyword model—to determine the “result string with the highest likelihood score” in the first stage of recognition, and thus matches the specific example of a “*log likelihood ratio*” provided in the specification of the ’196 patent for the first stage of recognition. Ex.1005, Sec.2.2, 1736. Accordingly, “score” as taught by Bou-Ghazale and Konopka renders obvious “*ratio*” as defined in the ’196 specification. *Id.* Moreover, Konopka teaches that it was well-known to compute a log likelihood ratio for a first “assumption” stage of recognition, and Bou-Ghazale uses the terms “log-likelihood score” and “log likelihood ratio” interchangeably. Ex.1006, [0080] (teaching a “score” that is a **ratio** of a log likelihood score for an in-vocabulary model to a log

likelihood score for an out-of-vocabulary model: “[**Ln(IV score)–Ln(OOV score)**]/number of frames”); Ex.1005, Sec.3.1, 1736 (using “relative log-likelihood score” for Criterion 6 of the post-processing algorithm), Sec.3.2, Figure 2, 1736 (using “relative log likelihood ratio” in Block E of Figure 2 which refers to the “relative log-likelihood score” in Criterion 6).

159. Thus, Bou-Ghazale’s teaching of determining whether the activation word has been found in the input speech by determining “the result string with the **highest likelihood score**” in light of Konopka’s teachings of computing a logarithm (“log”) of a Viterbi (likelihood) score for an in-vocabulary model to determine the best text strings, renders obvious “*wherein determining the input speech is recognized as the preset call command further comprises: computing a log likelihood ratio of the input speech.*”

[5.1] assuming the input speech to be the preset call command based on the computed log likelihood ratio.

160. Bou-Ghazale’s speech recognition component (“*token passing unit*”) assumes the input speech to be the activation word according to “the result string with the highest likelihood score” that is passed to the post-processing component, as explained for limitation [1.6]. Bou-Ghazale’s “highest likelihood score” renders obvious the “*computed log likelihood ratio*” considering Konopka’s teachings of computing a natural logarithm (“log”) of an in-vocabulary Viterbi (likelihood)

score to determine the best text strings in the in-vocabulary model that most likely match the unknown input speech, as explained for limitation [5.0].

161. Thus, Bou-Ghazale’s teaching of assuming the input speech to be the activation word based on the result string with the highest likelihood score renders obvious “*assuming the input speech to be the preset call command based on the computed log likelihood ratio.*”

f) Claim 8

[8.0] *The method of claim 1, further comprising: determining, using a rule check unit, whether or not the recognized subsequent input speech corresponds to a target sentence.*

162. Bou-Ghazale in view of Konopka renders obvious this limitation.

163. First, as discussed above for limitation [1.5], Bou-Ghazale’s main speech recognition system (“*continuous speech recognition unit*”) receives further commands subsequent the activation word and determines whether they are a word from the main menu of the main speech recognition system once the activation word is recognized in the input speech. Bou-Ghazale explains:

Upon recognizing that the activation command has been spoken, the voice activation system turns the device on or switches to the main speech recognition system to receive **further commands** from the user.

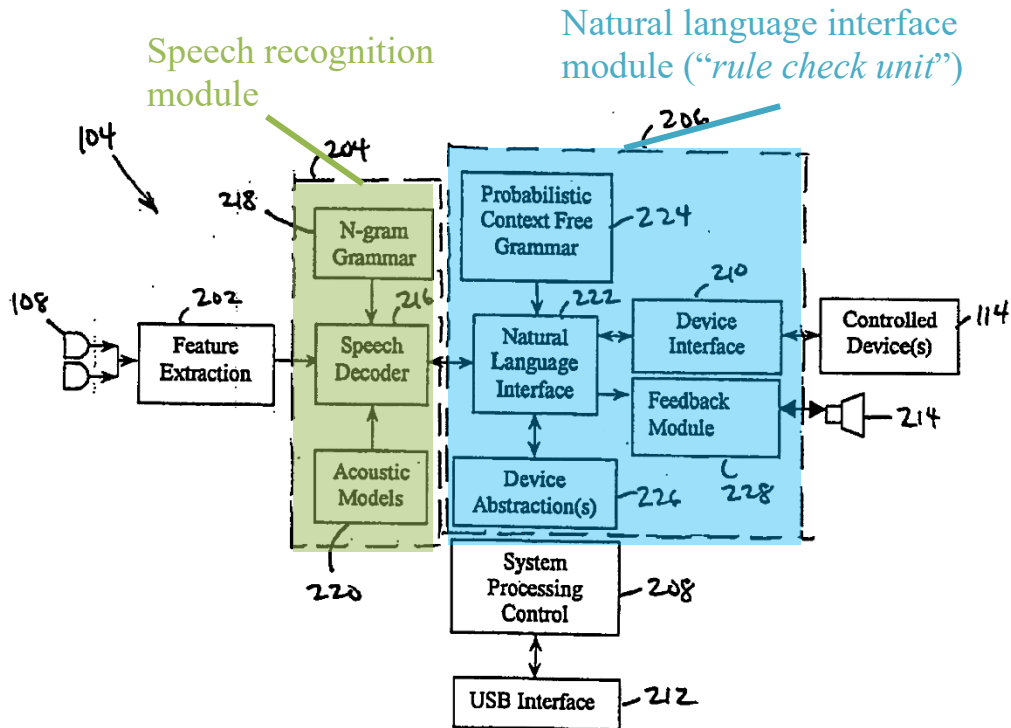
Ex.1005, Sec.1, 1735.

[T]he voice activation system can be designed in such a way that **once the activation word is recognized**, the device turns on or switches to the main recognition system and **waits to be prompted by a word from the main menu of the speech recognition system**.

Ex.1005, Sec.5, 1738. Because Bou-Ghazale's main speech recognition system is a continuous speech recognizer (as explained with respect to [1.5]) with a larger vocabulary ("menu") of words that it is capable of recognizing, a POSITA would have understood that Bou-Ghazale's main speech recognition system determines whether or not the speech recognized after the activation word corresponds to a sequence of valid words from Bou-Ghazale's "menu" (*"determining...whether or not the recognized subsequent input speech corresponds to a target sentence"*) so that the device may respond and take action accordingly.

164. Second, to the extent Bou-Ghazale does not explicitly teach that its main speech recognition system determines whether the further commands—recognized after the activation word as words in its "menu"—correspond to a "target sentence" using a "rule check unit," Konopka teaches that it was well-known to determine whether input speech corresponds to a known command sentence using a separate "rules"-based module. Ex.1006, [0059], [0020], [0022], Fig. 4. Konopka describes a "natural language interface control system" that

includes a “speech recognition module” and a “**natural language interface control module**,” as shown in Fig. 2 below. Ex.1006, [0020], [0002].



Ex.1006, Fig. 2 (annotated).

165. As shown in Fig. 4 below, after an “attention word” is recognized (step 416), the speech recognition module recognizes speech as “user requests” (steps 418, 408-412):

Once the user has the system's attention, i.e., the natural language interface module 222 has detected an attention word (Step 416), the natural language interface module knows the user's identity. It proceeds by configuring the system according to the user. It does this by changing the relevant system parameters and by directing the speech recognition module 204 to change

grammars to those appropriate for accepting commands and requests and according to the user. The speech recognition module 204 changes lexicons according to the grammar rules and the individual user. **Thus, the speech recognition module 204 and the natural language interface module 222 change contexts to look for user requests (Step 418).**

Ex.1006, [0081]; *see also* Ex.1006, [0040]. Then, in steps 414-416, and 420-424, the natural language interface module determines whether the recognized speech from the speech recognition module (“*recognized subsequent input speech*”) is a known (“unambiguous”) user request sentence (“*target sentence*”) using “*rule[s]*”:

Using the PCFG rules on the incoming text strings, the natural language interface module 222 reaches one of three conclusions: (1) that it unambiguously understands and can comply with the user request, in which case it carries out the command[.]

Ex.1006, [0059].

The NLICS 102...uses a natural language interface to interpret the natural language and **determine the probability of what the appropriate user request is. The natural language interface utilizes probabilistic context free grammar (also referred to as PCFG) rules and lexicons that are stored for each of the respective devices 114.**

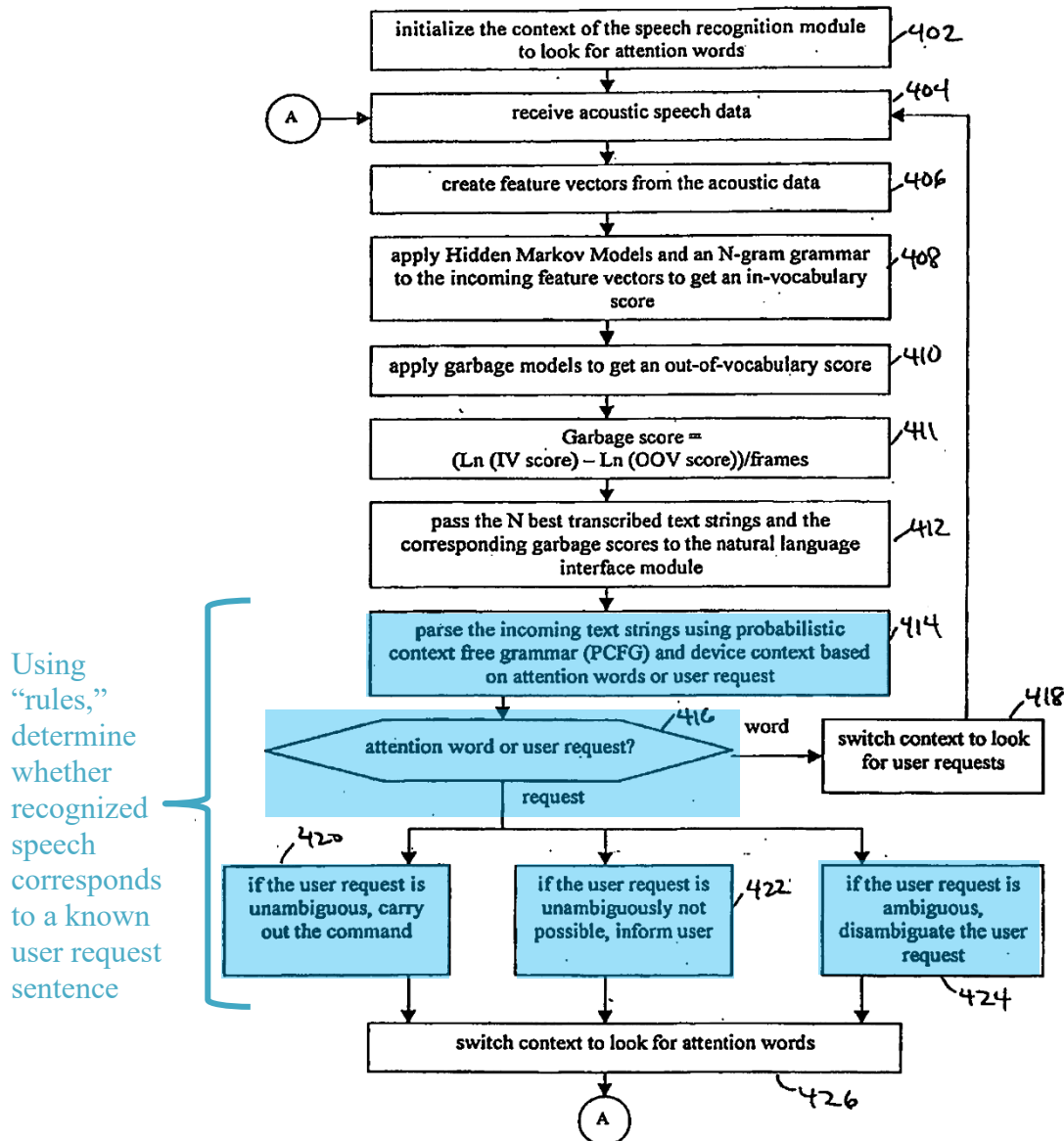
Ex.1006, [0020].

If the natural language interface module 222 detects a user request (Step 416), i.e. **a user request (as determined by the PCFG grammar system and device context)** is received, it draws on of three conclusions (Steps 420, 422 or 424).

Ex.1006, [0083].

166. For example, “if the user states ‘**hey, lets watch TV**’, ‘**I wanna watch TV**’, ‘**turn on the TV**’, ‘**whattya say we watch a little television**’, the system will use its natural language interface module to probabilistically **determine that the user is requesting to watch the television.**” Ex.1006, [0022].

A user request to watch TV, for example, renders obvious a “*target sentence.*”



Ex.1006, Fig. 4 (annotated).

Accordingly, in view of the well-known speech recognition techniques described by Konopka, a POSITA would have found it obvious that Bou-Ghazale's main speech recognition system ("*continuous speech recognition unit*") would have a separate rules-based module ("*rule check unit*"), for determining whether speech

recognized after the activation word corresponds to a known command sentence.

Thus, Bou-Ghazale in view of Konopka renders this limitation obvious.

167. Moreover, the same alternative means-plus-function mapping discussed in [1.1] applies to the “*rule check unit*.” In particular, a combination of hardware and software (Bou-Ghazale’s personal communication device with processor executing speech recognition software—*see* [1.0] and [9.3]) carries out the functions recited in association with the “*rule check unit*” in [8.0], which is a component of the “*continuous speech recognition unit*.”

g) Claim 9

168. Claim 9 recites substantially the same subject matter as claim 1, except in the form of an apparatus claim. Limitations [9.4]-[9.9] are substantially similar to limitations [1.1]-[1.6].

[9.0] *An apparatus for recognizing a speech of a speaker, the apparatus comprising:*

169. Bou-Ghazale teaches a voice activation system that “can be implemented in a **personal communication device**” (“*apparatus*”) for recognizing a pre-defined activation command (“*speech*”) spoken by a user (“*speaker*”). Ex.1005, Sec.5, 1738. Bou-Ghazale explains that its voice activation system is “designed to continuously listen to incoming unconstrained **speech** and to **recognize a pre-defined activation command** while rejecting any other extraneous speech or noise.” Ex.1005, Abstract, 1735. “Upon **recognizing that the**

activation command has been spoken, the voice activation system turns **the device** on or switches to the main speech recognition system to receive further commands from the **user**.” Ex.1005, Sec.1, 1735; *see also* Ex.1005, Abstract, 1735 (“The proposed system is intended for turning **a device** on or for activating a main speech recognition application upon **recognizing the activation utterance**.”).¹³

170. Thus, Bou-Ghazale’s teaching of a personal communication device on which a voice activation system can be implemented for recognizing a pre-defined activation command spoken by a user renders obvious an “*apparatus for recognizing a speech of a speaker*.”

[9.1] a processor,

171. Bou-Ghazale teaches that its voice activation system requires a low-speed processor (“*processor*”). Bou-Ghazale explains that its voice activation system “can be implemented in a personal communication device **due to its low memory and MIPS requirements (<1MIPS)**.” Ex.1005, Sec.5, 1738. It was well-known in the art at the time of the ’196 patent that “MIPS” is an acronym for “Million Instructions Per Second” and it refers to a measurement of a processor’s speed. Ex.1012, 63 (defining “MIPS (million instructions per second)” as “a measure of the speed of CPUs” where a CPU is a “central processing unit”);

¹³ *See supra* note 3.

Ex.1013, 129 (defining “Million Instructions Per Second (MIPS)” as “a rate of speed by which data is **processed** or by which a computer performs”); Ex.1014, 140 (defining “MIPS” as “Millions of instructions per second; a measure of **processor speed**”). Thus, a POSITA would have understood that Bou-Ghazale’s voice activation system utilizes a low-speed (e.g., <1 MIPS) processor. Ex.1005, Sec.5, 1738.

[9.2] *a preset call command recognition unit using the processor; and*

172. As discussed above with respect to [1.1], Bou-Ghazale’s “voice activation system” renders obvious “*a preset call command recognition unit.*” The voice activation system is “*using the processor*” because the system “continuously **processes** incoming speech.” Ex.1005, Sec.1, 1735 (“The voice activation system could be thought of as a keyword spotting system that continuously processes incoming speech as opposed to processing limited duration utterances.”). Therefore, Bou-Ghazale renders obvious “*a preset call command recognition unit using the processor.*”

[9.3] *a continuous speech recognition unit using the processor,*

173. As discussed above with respect to limitation [1.5], Bou-Ghazale’s “main speech recognition system” renders obvious “*a continuous speech recognition unit.*” The main speech recognition system is also “*using the processor*” of the personal communication device (“*apparatus*”) to recognize

further commands. As explained above for limitations [9.1] and [9.2], the voice activation system (“*preset call command recognition unit*”) uses the device’s processor to process incoming speech. Ex.1005, Sec.1, 1735, Sec.5, 1738. Bou-Ghazale explains that “once the activation word is recognized” in incoming speech, **“the device turns on or switches to the main speech recognition system and waits to be prompted by a word from the main menu of the speech recognition system.”** Ex.1005, Sec.5, 1738. A POSITA would have understood that when the device “switches” between the two components, the main speech recognition system uses the same processor to “receive further commands” and recognize them as a word in its main menu. Ex.1005, Sec.5, 1738, Sec.1, 1735.

174. Thus, Bou-Ghazale’s teaching of a main speech recognition system that uses the same processor as the voice activation system—the processor of the personal communication device—to recognize further commands renders obvious “*a continuous speech recognition unit using the processor.*”

[9.4] *wherein the preset call command recognition unit is configured to receive an input speech spoken by the speaker;*

175. Limitation [9.4] is substantially similar to limitation [1.1], and thus for the same reasons as discussed in association with [1.1], Bou-Ghazale renders obvious this limitation.

[9.5] *receive a subsequent input speech spoken, by the speaker, subsequently to the input speech;*

176. Limitation [9.5] is substantially similar to limitation [1.2], and thus for the same reasons as discussed in association with [1.2], Bou-Ghazale renders obvious this limitation.

[9.6] *compare the input speech with a preset call command; and*

177. Limitation [9.6] is substantially similar to limitation [1.3], and thus for the same reasons as discussed in association with [1.3], Bou-Ghazale renders obvious this limitation.

[9.7] *recognize the input speech as the preset call command according to a result of the comparison,*

178. Limitation [9.7] is substantially similar to limitation [1.4], and thus for the same reasons as discussed in association with [1.4], Bou-Ghazale renders obvious this limitation.

[9.8] *wherein the continuous speech recognition unit is configured to recognize the subsequent input speech as an actual command if the input speech is recognized as the preset call command by the preset call command recognition unit, and*

179. Limitation [9.8] is substantially similar to limitation [1.5], and thus for the same reasons as discussed in association with [1.5], Bou-Ghazale renders obvious this limitation.

[9.9] *wherein the preset call command recognition unit comprises a token passing unit configured to assume the input speech to be the preset call command according to the result of the comparison using a minimum recognition network composed of a silence interval accompanied by noise and the preset call command.*

180. Limitation [9.9] is identical to limitation [1.6], and thus for the same reasons as discussed in association with [1.6], Bou-Ghazale in view of Konopka renders obvious this limitation.

h) Claim 16

[16.0] *The apparatus of claim 9, wherein the continuous speech recognition unit comprises: a keyword spotting unit configured to recognize the subsequent input speech as the actual command if the input speech is recognized as the preset call command; and*

181. Bou-Ghazale in view of Konopka renders obvious this limitation.

182. **First**, as discussed above with respect to limitations [1.5] and [9.8], Bou-Ghazale's main speech recognition system ("*continuous speech recognition unit*") is configured to determine whether speech received after the activation word is recognized as a word from the main menu of the speech recognition system ("*configured to recognize the subsequent input speech as the actual command*") once the activation word is declared as found in the input speech ("*if the input speech is recognized as the preset call command*"). Bou-Ghazale explains:

Upon recognizing that the activation command has been spoken, the voice activation system turns the device on or **switches to the main speech recognition system to receive further commands** from the user.

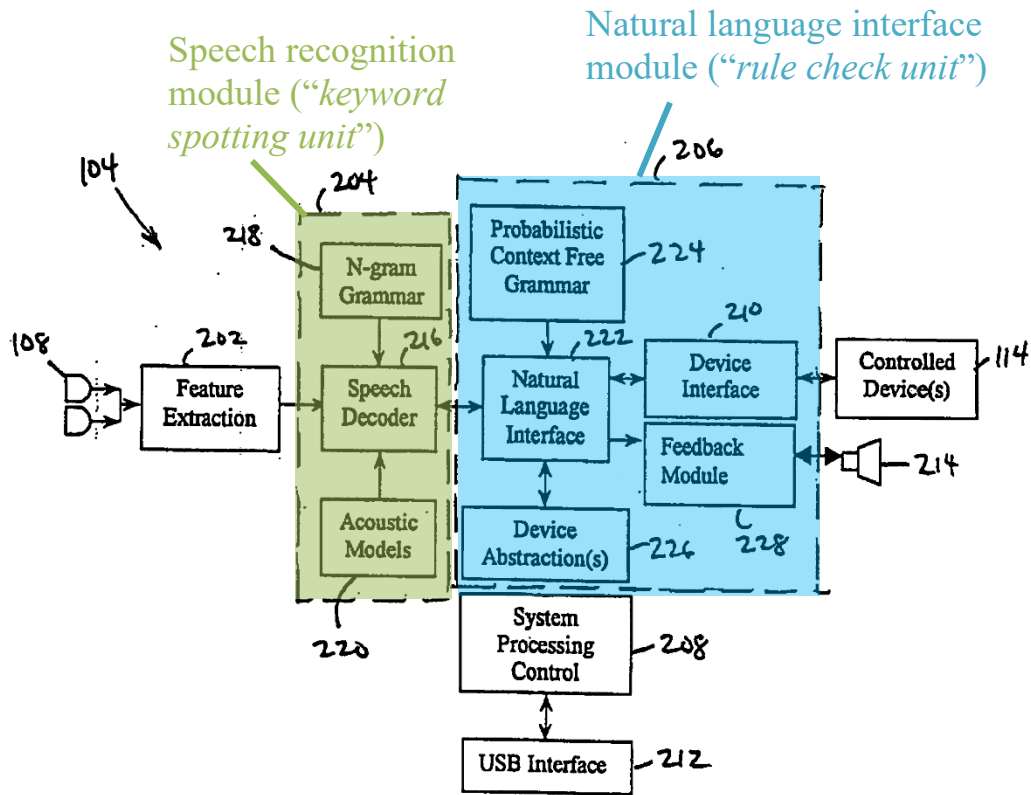
Ex.1005, Sec.1, 1735.

[T]he voice activation system can be designed in such a way that **once the activation word is recognized**, the

device turns on or switches to the main recognition system
and **waits to be prompted by a word from the main
menu of the speech recognition system.**

Ex.1005, Sec.5, 1738. Further, Bou-Ghazale teaches that its main speech recognition system spots a “*keyword*” because it continuously listens and waits to be prompted by a particular “**word** from the main menu of the speech recognition system.” *Id.*

183. Second, to the extent Bou-Ghazale does not explicitly teach that its main speech recognition system uses a separate “*keyword spotting unit*” to determine whether speech received after the activation word is an actual command (i.e., a word in its “menu”) after the activation word is recognized, Konopka renders it obvious. Konopka teaches that it was well-known to use a separate module, such as a “*keyword spotting unit*,” to recognize user commands after an activation command has been recognized. In more detail, Konopka describes a “natural language interface control system” that includes a “**speech recognition module**” 204 and a “natural language interface control module” 206, as shown in Fig. 2 below. Ex.1006, [0020], [0002].



Ex.1006, Fig. 2 (annotated).

184. After an "attention word is recognized," the speech recognition module is configured to recognize "user requests" ("*actual command*") by "finding the **most likely match of the corresponding utterance with a set of speech models**, while rejecting non-speech events, using an approach based on Hidden Markov Models (HMMs)," as shown by steps 416, 418, and 408-412 of Fig. 4 below. Ex.1006, [0040]; *see also* Ex.1006, [0020] ("The NLICS 102 includes a **speech recognition module** utilizing hidden Markov models (HMMs), as known and understood in the art, **to detect the speech.**"), [0080]-[0082] (detailing steps 416, 418, and 408-412 for recognizing "user requests" using the "speech

recognition module”). As Konopka further explains “[o]ne feature that enables the NLICS 102 to function efficiently is that each of the devices 114 coupled to the NLICS 102 are abstracted into a separate device abstraction such that separate grammars and **lexicons** are stored for each of the devices 114.” Ex.1006, [0028]. These sets of lexicons specify the “*keyword[s]*” that can be used in user input speech, and which are “*spott[ed]*” by the speech recognition module, after the system activates recognition of “user requests.” See Ex.1006, [0028], [0045] (“The N-gram grammar module 218 includes the grammars used by the speech decoder 216. These grammars are the rules by which lexicons are built and **a lexicon is a dictionary consisting of words and their pronunciation entries.**”).

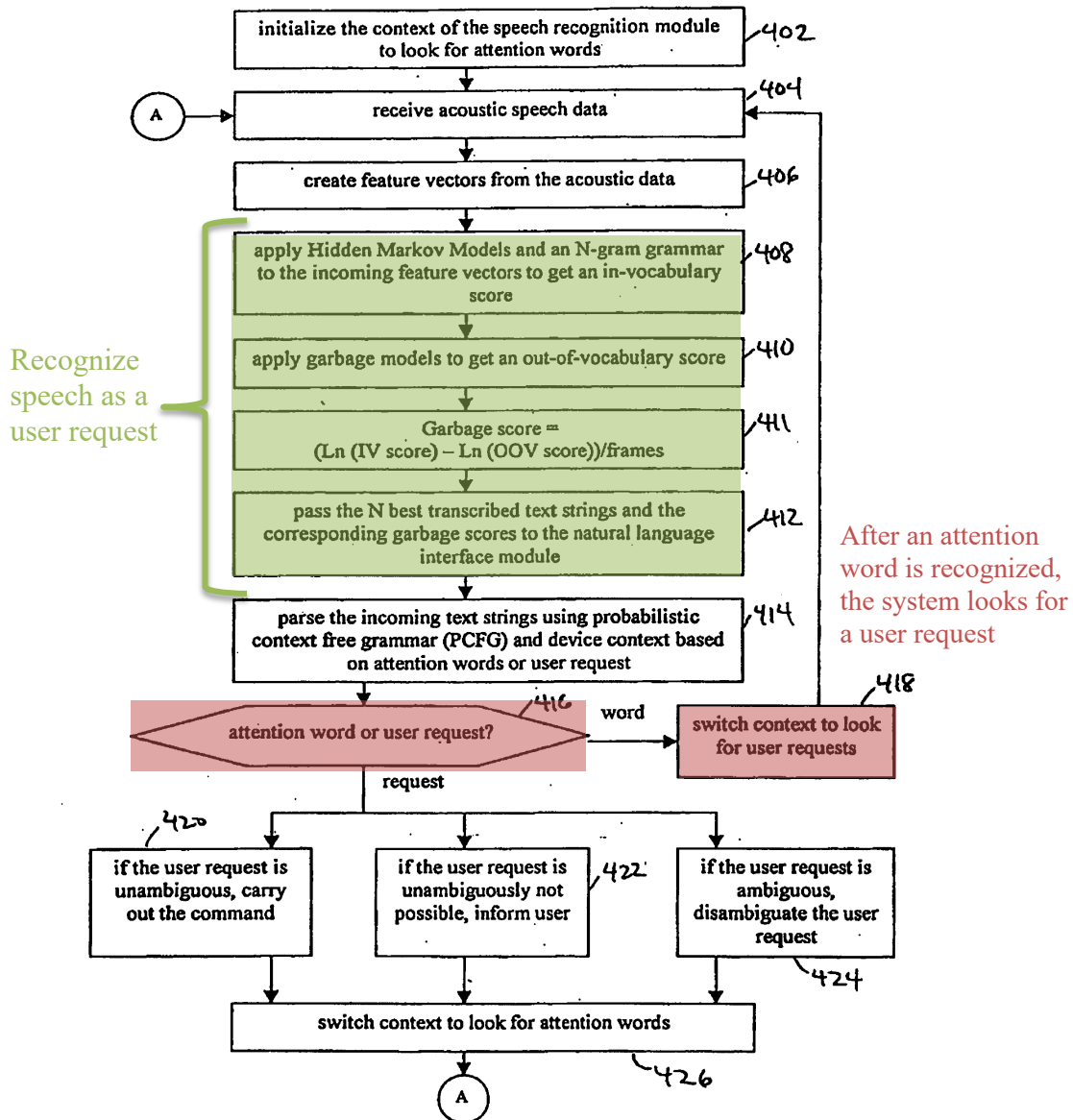


FIG. 4

Ex.1006, Fig. 4 (annotated).

185. Accordingly, in view of the well-known speech recognition techniques described by Konopka, a POSITA would have found it obvious that Bou-Ghazale's main speech recognition system ("*continuous speech recognition unit*") would have a separate module that looks for particular words in speech

(“*keyword spotting unit*”) and is for recognizing that speech as user commands, i.e., a “word” in Bou-Ghazale’s “menu” (“*actual command*”) after an activation command is recognized. Thus, Bou-Ghazale in view of Konopka renders this limitation obvious.

186. Moreover, the same alternative means-plus-function mapping discussed in [1.1] applies to the “*keyword spotting unit*.” In particular, a combination of hardware and software (Bou-Ghazale’s personal communication device with processor executing speech recognition software—*see* [1.0] and [9.3]) carries out the functions recited in association with the “*keyword spotting unit*” in [16.0], which is a component of the “*continuous speech recognition unit*.”

[16.1] *a rule check unit configured to determine whether the recognized subsequent input speech corresponds to a target sentence.*

187. Limitation [16.1] is substantially similar to limitation [8.0], and thus for the same reasons as discussed in association with [8.0], Bou-Ghazale in view of Konopka renders obvious this limitation.

i) Claim 17

188. Claim 17 recites substantially the same subject matter as claim 1, except for trivial wording differences.

[17.0] *A method for recognizing a speech of a speaker by a single device, the method comprising:*

189. Limitation [17.0] is substantially similar to limitation [1.0], except with reference to a “*single device*” instead of a “*device*.” As explained in [1.0],

Bou-Ghazale's voice activation system is "implemented in a **personal communication device**" ("*single device*"). Ex.1005, Sec.5, 1738, Sec.1, 1735 ("Upon recognizing that the activation command has been spoken, the voice activation system turns **the device** on"), Sec.1, 1735 ("[W]e propose using a voice activation system for turning **the device** on simply by speaking a pre-selected activation utterance."), Abstract ("The proposed system is intended for turning a **device** on...upon recognizing the activation utterance").

190. Thus, in light of the teachings cited above with reference to limitation [1.0], Bou-Ghazale's method for recognizing an activation command by a voice activation system implemented on a personal communication device renders obvious a "*method for recognizing a speech of a speaker by a single device.*"

[17.1] receiving, by a preset call command recognition unit of the single device, an input speech spoken by the speaker;

191. Limitation [17.1] is substantially similar to limitation [1.1], except it specifies that the "*preset call command recognition unit*" is "*of the single device.*" As explained in [1.1], Bou-Ghazale's voice activation system renders obvious a "*preset call command recognition unit.*" Moreover, as explained in [17.0] (with reference to [1.0]), Bou-Ghazale's voice activation system is "implemented in a **personal communication device**" ("*single device*"). Ex.1005, Sec.5, 1738, Sec.1, 1735, Abstract. Accordingly, Bou-Ghazale renders obvious "*a preset call command recognition unit of the single device.*"

192. Thus, for the same reasons as presented above for [1.1] and [17.0], Bou-Ghazale renders obvious this limitation.

[17.2] *receiving, by the preset call command recognition unit, a subsequent input speech spoken, by the speaker, subsequently to the input speech;*

193. Limitation [17.2] is identical to limitation [1.2]. Thus, Bou-Ghazale renders it obvious for the same reasons as presented above for limitation [1.2].

[17.3] *comparing, by the preset call command recognition unit, the input speech with a preset call command;*

194. Limitation [17.3] is identical to limitation [1.3]. Thus, Bou-Ghazale renders it obvious for the same reasons as presented above for limitation [1.3].

[17.4] *determining, by the preset call command recognition unit of the single device, whether the input speech is recognized as the preset call command according to a result of the comparison; and*

195. Limitation [17.4] is substantially similar to limitation [1.4], except it specifies that the “*preset call command recognition unit*” is “*of the single device.*” For the same reasons as presented above for limitations [17.0]-[17.1] (with respect to the “*preset call command recognition unit of the single device*”) and limitation [1.4], Bou-Ghazale renders obvious this limitation.

[17.5] *only if the input speech is recognized as the preset call command, activating a continuous speech recognition unit of the single device that is configured to perform continuous speech recognition to determine whether the subsequent input speech is recognized as an actual command, in order to reduce overall computational load on the single device; and*

196. Limitation [17.5] is substantially similar to limitation [1.5] except for trivial wording differences.

197. First, limitation [17.5] recites “***only if the input speech is recognized as the preset call command***” then the “*activating*” step is performed, instead of “***based on determining that the input speech is recognized as the preset call command,***” as recited in limitation [1.5]. As discussed for limitation [1.5], Bou-Ghazale explains that “the **activation word is declared to be found only after** [the input speech] meets a set of statistics, such as...the confidence level associated with the keyword” (“*the input speech is recognized as the preset call command*”). Ex.1005, Sec.2.3, 1736. Bou-Ghazale further explains that “**once the activation word is recognized, the device turns on or switches to the main recognition system.**” Ex.1005, Sec.5, 1738; *see also* Ex.1005, Sec.1, 1735 (“Upon **recognizing that the activation command has been spoken**, the voice activation system...**switches to the main speech recognition system**”), Figure 2 (showing that three paths, such as A, B, C, D, E, F, result in activating the main speech recognition system at Block F). Thus, Bou-Ghazale’s teaching that the main speech recognition system is activated “only after” the input speech meets the statistical criteria and is recognized as the activation word renders obvious “*only if the input speech is recognized as the preset call command, activating a continuous speech recognition unit.*”

198. Second, limitation [17.5] specifies that the “*continuous speech recognition unit*” is “*of the single device.*” As explained with respect to limitation

[1.5], Bou-Ghazale's main speech recognition system renders obvious "*a continuous speech recognition unit.*" Moreover, Bou-Ghazale teaches that its main speech recognition system is implemented on the same "personal communication device" ("*single device*") as the voice activation system discussed in [17.0], such that the device is able to "switch[]" between the two components. Ex.1005, Sec.5, 1738, Sec.1. 1735 ("Upon recognizing that the activation command has been spoken, the voice activation system turns **the device** on or **switches to the main speech recognition system** to receive further commands from the user."). Thus, for the same reasons as presented above, Bou-Ghazale renders obvious "*a continuous speech recognition unit of the single device.*"

199. Third, limitation [17.5] specifies that the "*continuous speech recognition unit*" is "*configured to perform continuous speech recognition.*" For the same reasons presented for [1.5], Bou-Ghazale renders obvious that the main speech recognition system ("*continuous speech recognition unit*") is "*configured to perform continuous speech recognition,*" and more specifically it performs continuous speech recognition "*to determine whether the subsequent input speech is recognized as an actual command.*" Ex.1005, Sec.1, 1735 ("switches to the main speech recognition system to receive **further commands** from the user"), Sec.5, 1738 ("once the activation word is recognized, the device turns on or **switches to the main recognition system and waits to be prompted by a word from the**

main menu of the speech recognition system” and “if no valid word is spoken in the following 10-15 seconds, then the system switches back to the stand-by mode and waits to be activated again”).

200. Fourth, with respect to “*in order to reduce overall computational load on the single device*,” limitation [17.5] recites “*single device*” instead of “*device*.” Bou-Ghazale renders this obvious for the same reasons as presented above for limitations [1.5] (with respect to “*in order to reduce overall computational load on the device*”) and [17.0] (with respect to the “*single device*”).

201. Thus, Bou-Ghazale renders obvious this limitation.

[17.6] *wherein the preset call command recognition unit comprises a token passing unit configured to assume the input speech to be the preset call command according to the result of the comparison using a minimum recognition network composed of a silence interval accompanied by noise and the preset call command.*

202. Limitation [17.6] is identical to limitation [1.6]. Bou-Ghazale in view of Konopka renders it obvious for the same reasons as presented above for limitation [1.6].

j) Claim 19

[19.0] *The method of claim 17, wherein the single device,*

203. Limitation [19.0] refers to the method of claim 17 and recites “*wherein the single device*” without limiting this feature further. Accordingly, Bou-Ghazale in view of Konopka renders obvious this limitation for the same reasons as presented above for claim 17.

[19.1] *wherein determining whether the input speech is recognized as the preset call command further comprises: computing, by a confidence detection unit, a confidence rate of the input speech; and*

204. Limitation [19.1] is identical to limitation [2.0]. Bou-Ghazale renders it obvious for the same reasons as presented above for limitation [2.0].

[19.2] *recognizing, by the confidence detection unit, the input speech as the preset call command based on the computed confidence rate.*

205. Limitation [19.2] is identical to limitation [2.1]. Bou-Ghazale renders it obvious for the same reasons as presented above for limitation [2.1].

k) Claim 20

[20.0] *The method of claim 17, wherein determining whether the input speech is recognized as the preset call command further comprises: computing a log likelihood ratio of the input speech; and*

206. Limitation [20.0] is identical to limitation [5.0], except it includes the word “*whether*” in reciting the “*determining*” step from limitation [17.4]. Bou-Ghazale renders it obvious for the same reasons as presented above for limitation [5.0].

[20.1] *assuming the input speech to be the preset call command based on the computed log likelihood ratio.*

207. Limitation [21.0] is identical to limitation [5.1]. Bou-Ghazale renders it obvious for the same reasons as presented above for limitation [5.1].

B. Ground 2: Claims 6-7 are obvious under §103 over Bou-Ghazale in view of Konopka and further in view of Lee.

1. Summary of Lee

208. Lee, like Bou-Ghazale and Konopka relates to HMM-based continuous speech recognition. Ex.1007, 29. Lee teaches a technique for recognizing and verifying a voice code while “reject[ing] non-speech noises, out-of-vocabulary words, and potential recognition mistakes” for systems of “limited memory size.” Ex.1007, 29. Lee explains that “for the general purpose of speech recognition, acoustic models are modeled to represent general characteristics of speech using HMM” and “acoustic models can be represented as word unit or phoneme unit.” *Id.* To this end, Lee’s technique uses a “phoneme” model that represents characteristics of the voice code and an anti-phoneme model that is “based on phoneme and syllable units against the claimed voice code using the statistical distance of phonemes.” Ex.1007, 29-30. Because Lee’s anti-phoneme model is constructed by reusing a general acoustic model for the phoneme units that are “statistically far from the phoneme group,” an extra trained garbage model is not required and therefore Lee’s technique is advantageous for systems with limited memory size. Ex.1007, 33, 29.

2. Reasons to Combine Bou-Ghazale and Konopka with Lee

209. It is my opinion that a POSITA would have been motivated to combine the teachings of Lee with the teachings of Bou-Ghazale and Konopka. It

would have been obvious, beneficial, and predictable to apply Lee's teachings of using phoneme and anti-phoneme models for recognizing and verifying commands in unknown speech to Bou-Ghazale's voice activation system (as modified by Konopka). This combination would, for example, further Bou-Ghazale's and Lee's goals of accurately recognizing commands while rejecting noise or other extraneous speech, and further reduce the memory requirements of Bou-Ghazale and Konopka's system.

210. As an initial matter, a POSITA, when considering the teachings of Bou-Ghazale and Konopka would have also considered the teachings of Lee. Lee relates to HMM-based continuous speech recognition. Ex.1007, 29. In addition, although Lee is specifically directed at speaker authentication, all three references disclose systems in which it is essential to perform high accuracy recognition of a single specific word. Ex.1007, 29. As such, Lee is analogous art to the '196 patent. *See, e.g.,* Ex.1001, Abstract, Fig. 6.

211. A POSITA considering Bou-Ghazale would understand that among Bou-Ghazale's goals is accurately recognizing commands while rejecting noise or other extraneous speech and reducing the system's memory requirements. *See* Ex.1005, Abstract (explaining that its system is designed "to **recognize a pre-defined activation command while rejecting any other extraneous speech or noise.**"), Sec.1, 1735 ("**maintaining low false alarm rates** and low-power

consumption **are critical**”), Sec.5, 1738 (“The system requires relatively low power and can be implemented in a personal communication device due to its low memory and MIPS requirements”). At the time of the ’196 patent, a well-known way to “reject non-speech noises, out-of-vocabulary words, and potential recognition mistakes” while reducing memory requirements was to use phoneme and anti-phoneme models, where the phoneme model represents characteristics of a voice code (i.e., activation word) and the anti-phoneme model is “based on phoneme and syllable units against the claimed voice code using the statistical distance of phonemes,” as taught by Lee. Ex.1007, 29-30. Lee explains that its technique does not require an extra trained garbage model representing out-of-vocabulary speech and therefore is advantageous for systems with limited memory requirements. Ex.1007, 29 (“Most algorithms require the extra model trained for a garbage model or anti-model. However, the limited memory size of our proposed embedded system prevents the algorithm from using and storing the extra alternative hypothesis model.”). Because Bou-Ghazale’s system is “implemented in a personal communication device due to its low memory...requirements” a POSITA would have been motivated to use Lee’s phoneme models and anti-phoneme models as its keyword model and out-of-vocabulary models, respectively, to recognize commands while rejecting noise or other extraneous speech. Ex.1005, Sec.5, 1738. Doing so would eliminate the need to train and store

an extra garbage model for out-of-vocabulary words in Bou-Ghazale and Konopka's system, thereby saving memory space. Additionally, a POSITA would have been motivated to consult Lee's disclosure for its increased detail in how to construct Bou-Ghazale's out-of-vocabulary speech models.

212. A POSITA would have had a reasonable expectation of success applying Lee's technique to Bou-Ghazale and Konopka. For example, Lee's technique was specifically intended to be implemented in an HMM-based speech recognition system, such as Bou-Ghazale's voice activation system. *See* Ex.1007, 29. Moreover, the combination would not change the intended functionality of Bou-Ghazale's voice activation system, it simply would improve it. Bou-Ghazale's system and method would still operate to reject noise or other extraneous speech, but do so without having to train and store an extra garbage model, as taught by Lee.

213. Accordingly, a POSITA would have found it obvious to apply Lee's teachings to the system and methods taught by Bou-Ghazale and Konopka because the combination merely amounts to applying a known technique to a system and methods ready for improvement.

3. Detailed Analysis of the Claims

a) Claim 6

[6.0] *The method of claim 5, wherein the computed log likelihood ratio is a natural logarithm of a probability on the input speech for a phoneme model of the preset call command, and*

214. Bou-Ghazale in view of Konopka and further in view of Lee renders obvious this limitation.

215. As explained for [5.0] and [5.1], Bou-Ghazale teaches that its speech recognition component determines “the result string with the highest likelihood score” and teaches that a likelihood score may be a “log-likelihood score for the keyword model”:

The recognition algorithm performs a trace-back at every frame, and passes the result string with the **highest likelihood score** to the post-processing component.

Ex.1005, Sec.2.2, 1736.

Criterion 6: The relative log-likelihood score should be equal to or greater than a threshold, η , given by: $\eta = (\text{Score}_{\text{keyword}} - \text{Score}_{\text{OOV}}) / |\text{Score}_{\text{keyword}}|$, where the accumulated **log-likelihood score for the keyword model** is denoted by $\text{Score}_{\text{keyword}}$, and the log-likelihood score for the OOV model is denoted by $\text{Score}_{\text{OOV}}$.

Ex.1005, Sec.3.1, 1736.

216. As also explained for [5.0], Konopka teaches computing a natural logarithm of an “in-vocabulary (IV) Viterbi (likelihood) score” to determine the best text string (e.g., Bou-Ghazale’s result string with the highest likelihood score) during a first recognition stage. Ex.1005, Sec.2.2-3.1, 1736; Ex.1006, [0080], Fig. 4. Konopka explains that the Viterbi score “represents a single best state sequence through an HMM and its corresponding **probability**”:

[I]t is sometimes preferred to determine whether the utterance is contained within in-vocabulary words or not. For example, if a Viterbi score passes a threshold, the utterance is determined to be an in-vocabulary word. If the Viterbi score of the utterance does not exceed the threshold, then the utterance is deemed out-of vocabulary. **Such a Viterbi score is generated using the Viterbi algorithm. This algorithm calculates a single best state sequence through an HMM and its corresponding probability, given an observation sequence.**

Ex.1006, [0043].

217. Accordingly, a POSITA would understand that Konopka’s in-vocabulary likelihood score is a “*probability*” on the input speech for a hidden Markov model (HMM) for an in-vocabulary word. *See also* Ex.1019, Page 37 of 102 (explaining that a “probability” of a sequence occurring for a given word is known as “likelihood” of that word: “this conditional probability of the

observations Y given word w is known as the ‘likelihood’ of the word w ”),
Ex.1018, Page 168 of 343 (explaining that an HMM is a “probabilistic process”).
Therefore, based on the teachings of Konopka, Bou-Ghazale’s “likelihood score,”
as computed by the speech recognition component is a “probability” on the input
speech for its keyword model, of which a natural logarithm can be computed
(“*natural logarithm of a probability on the input speech for a phoneme model of
the preset call command*”).

218. Bou-Ghazale is based on “...continuous density hidden Markov
models (CD-HMM)” (Ex.1005, Sec.2.2, 1736), and as just discussed, Konopka is
also based on HMMs. A POSITA would have recognized that the states in HMMs
for speech recognition typically model phonemes, but to the extent Bou-Ghazale
and Konopka do not explicitly recite that their “keyword model” and “in-
vocabulary model” are a “*phoneme model of the preset call command*,” Lee
renders it obvious because Lee teaches using a “phoneme model” (which is also
referenced by its use of “phoneme hypotheses”) for a voice code word. Lee teaches
a “voice code verification” system and process, that uses a phoneme model and an
anti-phoneme model:

This is triphone, which is **the model used in the proposed
system. Such a phoneme model** is modeled using HMM
having states. In our system, we modeled it using three

states, left–right HMM, which is then arranged in triphone models.

Ex.1007, 29.

The **proposed anti-phoneme model** is based on Korean phonetic theory.

Ex.1007, 38; *see also id.* at 3 (describing production rules for an “anti-phoneme model” in Table 2 and explaining that the “anti-phoneme group is the one that is statistically far from the phoneme group”).

219. In particular, Lee describes a two-step process, where in the first step “recognition is performed via a conventional Viterbi beam search algorithm that segments the test utterance into the **N-best strings of a phoneme hypotheses.**”

Ex.1007, 30. As was well-known in the art, the Viterbi algorithm uses models, such as acoustic models, to analyze the input speech and determine “scores” for the strings in the models—acoustic models are modeled using HMMs to represent characteristics of speech and can be modeled as a phoneme unit:

For the general purpose of speech recognition, acoustic models are modeled to represent general characteristics of speech using HMM....[A]coustic models can be modeled as word unit or **phoneme unit**.

Ex.1007, 29.

As is well understood in the art, **the received utterances are analyzed with the models**, e.g., using a Viterbi

algorithm, and scores are assigned representing how well the utterance fits the given models.

Ex.1006, [0041].

220. Accordingly, a POSITA would have understood that Lee's "phoneme hypotheses" refers to "phoneme models" representing characteristics of the voice code, and the "N-best text strings of a phoneme hypotheses" refers to the strings in the phoneme model that have the best "scores" indicating that an utterance closely matches the characteristics in the voice code. As stated above, Lee relies on HMMs for speech recognition. The second "M" in HMM refers to the word models used by this probabilistic technique. As was well known, each model of a word is represented as a series of states, where each state typically refers to a phoneme. In particular, Lee uses a "triphone model"; "phone" refers to the actual sound of a particular phoneme, which may be produced slightly differently, e.g., in a dialect. As Lee explains "...if we take into consideration both the left and the right neighboring phones, the acoustic models are context dependent models because a phoneme affects the utterance change. This is triphone, which is the model used in the proposed system" Ex.1007, 29. The triphone model is simply the sound of a particular phoneme in the acoustic context of which phonemes precede and follow it, as these may change the way the phoneme sounds (i.e., "phone"). Thus, Lee's use of a phoneme model, specifically a triphone model, representing a voice code

word to determine the “best strings” (i.e., those with the highest likelihood score), renders obvious “*a probability on the input speech for a phoneme model of the preset call command.*”

221. Thus, Bou-Ghazale’s teaching of a likelihood score for a keyword model in view of Konopka’s teachings of computing a natural logarithm of a probability on the input speech for a model for an in-vocabulary word, and further in view of Lee’s teachings of using a phoneme model for a voice code word (i.e., command) to determine a score renders obvious “*wherein the computed log likelihood ratio is a natural logarithm of a probability on the input speech for a phoneme model of the preset call command.*”

[6.1] *the computed confidence rate is another log likelihood ratio.*

222. As explained for [2.0], Bou-Ghazale teaches that the post-processing component determines a “relative log-likelihood score” (which Bou-Ghazale also refers to as a “relative log likelihood ratio”) for the input speech (“*computed confidence rate*”) to determine whether the input speech is recognized as the activation word. Ex.1005, Sec.2.3-3.2, 1736. While Bou-Ghazale’s “likelihood score” determined by the speech recognition component may be a log-likelihood score for the **keyword model** as explained with respect to [5.0] and [6.0], the “relative log-likelihood score” determined by the post-processing component is “*another log likelihood ratio*” because it uses an “accumulated log-likelihood score

for the keyword model” **and** a “log-likelihood score for the [out-of-vocabulary (OOV)] model”:

The **relative log-likelihood score should be equal to or greater than a threshold, η** , given by: $\eta = (\text{Score}_{\text{keyword}} - \text{Score}_{\text{OOV}}) / |\text{Score}_{\text{keyword}}|$, where the accumulated **log-likelihood score for the keyword model** is denoted by $\text{Score}_{\text{keyword}}$, and the **log-likelihood score for the OOV model** is denoted by $\text{Score}_{\text{OOV}}$.

Ex.1005, Sec.3.1, 1736. Further, Bou-Ghazale refers to its “relative log likelihood **score**” from Criterion 6 as a “relative log likelihood **ratio**” (“*log likelihood ratio*”).

Ex.1005, Sec.3.1-3.2, Figure 2, 1736-37.

223. Thus, Bou-Ghazale’s teaching of a relative log-likelihood score that is different from the likelihood score determined by the speech recognition component renders obvious “*the computed confidence rate is another log likelihood ratio.*”

b) Claim 7

[7.0] *The method of claim 6, wherein said another log likelihood ratio is computed by subtracting, from the log likelihood ratio, a natural logarithm of a probability on the input speech for an anti-phoneme model of the preset call command.*

224. Bou-Ghazale in view of Konopka and further in view of Lee renders obvious this limitation.

225. Bou-Ghazale teaches that its “relative log-likelihood score” (“*another log likelihood ratio*”) is computed by subtracting a “log-likelihood score for the OOV model” from an “accumulated log-likelihood score for the keyword model”:

The relative log-likelihood score should be equal to or greater than a threshold, η , given by: $\eta = (\text{Score}_{\text{keyword}} - \text{Score}_{\text{OOV}}) / |\text{Score}_{\text{keyword}}|$, where the accumulated log-likelihood score for the keyword model is denoted by $\text{Score}_{\text{keyword}}$, and the log-likelihood score for the OOV model is denoted by $\text{Score}_{\text{OOV}}$.

Ex.1005, Sec.3.1, 1736.

226. First, as explained above for [5.0] and [6.0], Bou-Ghazale’s “log-likelihood score for the keyword model” in the above equation corresponds to “*the log likelihood ratio*” that is “*a natural logarithm of a probability on the input speech for a phoneme model of the preset call command.*”

227. Second, Bou-Ghazale’s “log-likelihood score for the [out-of-vocabulary] OOV model” in the above equation corresponds to “*a natural logarithm of a probability on the input speech for an anti-phoneme model of the preset call command.*” Konopka teaches a similar equation to that of Bou-Ghazale for determining a log-likelihood “score” using a log-likelihood score for an in-vocabulary model ($\text{Ln}(\text{IV score})$) and a log-likelihood score for an out-of-

vocabulary model ($\text{Ln}(\text{OOV score})$): “ **$\text{Ln}(\text{IV score}) - \text{Ln}(\text{OOV score})$**]/number of frames.” Ex.1006, [0080], Figure 1.

228. For similar reasons as discussed above for an in-vocabulary likelihood score in [6.0], a POSITA would understand that Konopka’s out-of-vocabulary likelihood score is a “**probability**” on the input speech for a hidden Markov model (HMM) for out-of-vocabulary words. Ex.1006, [0043] (explaining that the out-of-vocabulary score is “**another Viterbi score** that is obtained by **reprocessing the utterance through an alternative HMM** whose task is to **represent all out-of-vocabulary events** or filler sounds, i.e., using garbage models”). Therefore, Bou-Ghazale’s “log-likelihood score for the OOV model,” is a natural logarithm probability on the input speech for an out-of-vocabulary model (“*a natural logarithm of a probability on the input speech for an anti-phoneme model of the preset call command*”).

229. Third, to the extent Bou-Ghazale and Konopka do not explicitly recite that an “out-of-vocabulary model” is an “*anti-phoneme model of the preset call command*,” Lee renders it obvious because Lee teaches using an “**anti-phoneme model**” (Ex.1007, 38) in a second step of its process to “compute[] a **confidence measure** that determines whether or not to reject the recognized voice code” from the first step of its process:

In the second pass, voice code verification is performed. It computes a confidence measure that determines whether or not to reject the recognized voice code. The voice code verification technique is formulated as a statistical hypothesis test where the task is to test the null hypothesis that is a claimed voice code against the alternative hypothesis. Generally, the alternative hypothesis model as an anti-model is a previously trained model consisting of data that are mis-recognized or out-of-vocabulary generating false alarms, and noise tokens that also generat[e] false alarms. However, the embedded hardware system requires limited and inexpensive memory. Thus, **we reuse the general acoustic model as an anti-model** to compete with the claimed voice code model.

Ex.1007, 30.¹⁴ In more detail, Lee explains that its anti-phoneme models are specific to the voice code (“*of the preset call command*”) because they are determined based on the phonemes that are most different from the voice code:

Thus, in this paper, we propose the **anti-models that are specific against a given voice code word model** by using a regular production rules. The anti-models are automatically constructed using a grapheme to phoneme converter through the analysis of voice code text. The **anti-models are based on phoneme and syllable units**

¹⁴ Lee uses the terms “anti-model” and “anti-phoneme model” interchangeably.

against the claimed voice code using the statistical distance of phonemes.

Ex.1007, 30.

The anti-phoneme group is the one that is statistically far from the phoneme group.

Ex.1007, 33. Specifically, the voice code word is modeled as a series of the triphone states that define a language (i.e., the phoneme model). Lee builds a set of alternative models, using the same repertoire of triphone states, but corresponding to phonemes that differ from the voice code, according to a set of phonetic similarity rules. Ex.1007, 33-35. Thus, it would be clear to a POSITA that Lee's anti-models are "*anti-phoneme*" models in that they are also composed of phonemes, but at least some of these phonemes are chosen to be dissimilar from those in the voice code.

230. For at least the reasons explained above in Section VII.B.3, a POSITA would have been motivated to use Lee's anti-phoneme model as Bou-Ghazale's out-of-vocabulary model. Lee explains that its technique may be applied to "**reject non-speech noises, out-of-vocabulary words**, and potential recognition mistakes" and it saves memory space because it eliminates the need to train and store a garbage model for out-of-vocabulary words. Ex.1007, 29 ("Most algorithms require the extra model trained for a garbage model or anti-model. However, the limited memory size of our proposed embedded system prevents the algorithm

from using and storing the extra alternative hypothesis model.”). Accordingly, a POSITA would have been motivated to use Lee’s anti-phoneme model as Bou-Ghazale’s out-of-vocabulary model because Bou-Ghazale and Konopka’s system would still function to recognize commands while rejecting noise or other extraneous speech, but it could do so without having to train and store an extra garbage model for out-of-vocabulary words, thereby saving memory space. Additionally, because Lee’s anti-phoneme models are built automatically by the described algorithm, if Bou-Ghazale’s keyword were changed, perhaps because a user had the option of giving their own preferred name to the personal device, the anti-model would be generated automatically. This personalization would increase the appeal of the device to some users.

231. Thus, Bou-Ghazale’s teaching of computing a relative log-likelihood score by subtracting a log-likelihood score for the OOV model from an accumulated log-likelihood score for the keyword model, in view of Konopka’s teachings of a similar equation that computes a natural logarithm of a probability on the input speech for an out-of-vocabulary model, and further in view of Lee’s teachings of an anti-phoneme model for determining a confidence measurement to reject out-of-vocabulary words renders obvious “*wherein said another log likelihood ratio is computed by subtracting, from the log likelihood ratio, a natural*

logarithm of a probability on the input speech for an anti-phoneme model of the preset call command.”

C. Ground 3: Claims 10-13, and 18 are obvious under §103 over Bou-Ghazale in view of Konopka and further in view of Takiguchi.

1. Summary of Takiguchi

232. Takiguchi relates to “a speech enhancement technique which is effective for an extemporaneous noise without a noise interval and also for unknown extemporaneous noises” and which “can be applied to, for example, a speech recognition device and method.” Ex.1008, [0026], [0063]. More specifically, Takiguchi’s technique utilizes an “adaptive filter” with a corresponding filter coefficient that is applied to incoming speech to “reduce the noise signal.” Ex.1008, [0027]. In Takiguchi’s technique, “the filter coefficient is updated for each utterance [or for each frame or for every plurality of frames], and the target speech signal is enhanced using the updated filter coefficient.” Ex.1008, [0101]. Accordingly, Takiguchi’s adaptive filter adapts to and effectively removes different types of extemporaneous noises, thereby improving speech recognition accuracy in noisy environments. *See* Ex.1008, [0026], [0022].

2. Reasons to Combine Bou-Ghazale and Konopka with Takiguchi

233. It is my opinion that a POSITA would have been motivated to combine the teachings of Takiguchi with the teachings of Bou-Ghazale and Konopka. It would have been obvious, beneficial, and predictable to utilize

Takiguchi's teachings of an adaptive filtering technique in combination with Bou-Ghazale's voice activation system. This combination would, for example, further Bou-Ghazale's and Takiguchi's goals of effectively reducing extemporaneous noises (or extraneous speech), thereby improving speech recognition.

234. As an initial matter, a POSITA, when considering the teachings of Bou-Ghazale and Konopka would have also considered the teachings of Takiguchi. Takiguchi relates to HMM-based speech recognition. Ex.1008, [0026]-[0027] (“The present invention provides a signal enhancement device designed to enhance a target signal..., a **speech recognition** device, and a method and a program of the same..”), [0057] (“This invention provides signal enhancement devices and **speech recognition**.”), [0059] (“An appropriate statistical model is, for example, the **Hidden Markov model (HMM)** in which the occurrence probability of a spectral pattern in a state transition is represented by a Gaussian distribution. The filter coefficient is controlled by, for example, using the expectation-maximization (EM) algorithm.”). As such, Takiguchi is analogous art to the '196 patent. *See*, e.g., Ex.1001, Abstract, Fig. 6

235. A POSITA considering Bou-Ghazale would understand that one of Bou-Ghazale's primary concerns is accurately recognizing commands while rejecting noise or other extraneous speech. *See* Ex.1005, Abstract (explaining that its system is designed “to recognize a pre-defined activation command while

rejecting any other extraneous speech or noise.”), Sec.1, 1735 (“**maintaining low false alarm rates** and low-power consumption **are critical**”). Bou-Ghazale teaches a voice activation system that includes a “band-pass filter” for filtering speech within a set range, thereby rejecting noise outside of that range:

The input speech is sampled at 8kHz, then filtered by a band-pass filter with a pass-band from 300 to 3300 Hz. The filtered speech is processed at 20 ms frame rate with a Hamming window of 30 ms.

Ex.1005, Sec.2.2, 1736.

236. However, Takiguchi explains that it is difficult to estimate a set filtering coefficient for effectively removing different types of extemporaneous noises, and therefore speech recognition accuracy deteriorates:

The filter coefficient $w(\omega)$ is adjusted using the input signals $m_1(t)$ and $m_2(t)$ in a noise interval[.]...The reason for using input signals in the noise interval is that the learning of the filter coefficient is inhibited if components of the target speech signal are contained in the main input $p(\omega; n)$. Accordingly, **it is difficult to estimate the filter coefficient $w(\omega)$ for removing extemporaneous noise which is completely superimposed on the target speech signal, which exists only in the speech occurrence interval, and which continues for a short time.** Accordingly, in speech recognition for transcribing a lecture or a meeting, speech recognition in a car, or the

like, extemporaneous noise, such as the sound of something hitting something else, the sound of touching paper for turning a page, the sound of closing a door, or the like, is one cause of **deteriorating recognition accuracy**.

On the other hand, as a speech recognition method in the presence of extemporaneous noise, a technique has been proposed in which matching between a feature of input speech and a composite model constituted by the Phonemic Hidden Markov model of speech data and the Hidden Markov model of noise data is performed and in which, based on the result, input speech is recognized (refer to patent document 1). In this technique, **the type of target extemporaneous noise is necessarily known. However, in some cases, it may be difficult to forecast and model the types of noise which can occur, because various types of noise exist in an actual environment.**

Ex.1008, [0022]-[0023].

237. Takiguchi provides a well-known and predictable solution to this problem of a set filtering range by using an “adaptive filter” where “a filter coefficient of [the] adaptive filter to be applied to the reference signal” is “updated” for each portion of speech “to reduce the noise signal.” Ex.1008, [0027], [0101]. As a result, an adaptive filtering (“speech enhancement”) technique “which

is effective for an extemporaneous noise without a noise interval and also for unknown extemporaneous noises” is provided. Ex.1008, [0026].

238. A POSITA would have been motivated to implement Takiguchi’s adaptive filtering technique into Bou-Ghazale’s voice activation system to more effectively model and reduce extemporaneous noises and improve speech recognition, thereby minimizing spurious rejections or false positives when listening for the keyword. Specifically, the benefits of Takiguchi’s technique would have motivated a POSITA to modify Bou-Ghazale’s band-pass filter to be adaptive, so that it updates its filter coefficient for each portion of speech. Doing so would reduce extemporaneous noise (or extraneous speech) in the filtered speech and improve recognition accuracy of the activation word.

239. A POSITA would have had a reasonable expectation of success applying Takiguchi’s technique to Bou-Ghazale and Konopka. For example, Takiguchi’s technique was specifically intended to be implemented in an HMM-based speech recognition system, such as Bou-Ghazale’s voice activation system. See Ex.1008, [0057], [0059]. Moreover, the combination would not change the intended functionality of Bou-Ghazale’s voice activation system, it would simply improve its current range for rejecting extemporaneous noises, as taught by Takiguchi.

240. Accordingly, a POSITA would have found it obvious to apply Takiguchi's teachings to the system and methods taught by Bou-Ghazale and Konopka because the combination merely amounts to applying a known technique to a system and methods ready for improvement.

3. Detailed Analysis of the Claims

a) Claim 10

[10.0] *The apparatus of claim 9, wherein the preset call command recognition unit further comprises: an adaptive filter configured to receive the input speech and the subsequent input speech; and*

241. Bou-Ghazale in view of Konopka and further in view of Takiguchi renders obvious this limitation.

242. Bou-Ghazale teaches that its voice activation system ("*preset call command recognition unit*") includes a filter for filtering speech:

The **input speech** is sampled at 8kHz, then **filtered by a band-pass filter** with a pass-band from 300 to 3300 Hz. **The filtered speech is processed** at 20 ms frame rate with a Hamming window of 30 ms.

Ex.1005, Sec.2.2, 1736. The band-pass filter "*receive[s] the input speech and the subsequent input speech*" because Bou-Ghazale's voice activation system "**continuously**" listens to and processes incoming speech, but before the speech is processed it is "filtered." Ex.1005, Sec.1, 1735 ("The voice activation system continuously listens to and processes incoming speech."), Sec.2, 1736 ("[t]he

filtered speech is processed”).

243. Bou-Ghazale’s band-pass filter uses a fixed frequency range (“300 to 3300 Hz”) such that any speech outside of that range is removed from incoming speech. Ex.1005, Sec.2.2, 1736. Meanwhile, Takiguchi teaches that an “adaptive filter” can be used to adjust the frequency range to more effectively reduce extemporaneous noises and improve speech recognition. Ex.1008, [0026]-[0027] (“it is an aspect of the present invention to provide a speech enhancement technique which is effective for an extemporaneous noise without a noise interval and also for unknown extemporaneous noises”). Takiguchi describes a “signal enhancement device designed to enhance a target signal...by controlling a filter coefficient of **an adaptive filter to be applied to the reference signal** to reduce the noise signal.” Ex.1008, [0027]. Takiguchi explains that the “reference signal” to which the “adaptive filter” is applied to is obtained from “input signals” including speech utterances. Ex.1008, [0086] (explaining that the “signal synthesis unit 31 also **adds the input signal m2 to the input signal m1 after inverting the input signal m2...** and **outputs** the resultant signal as the **reference signal**...The adaptive filter unit 33 **adjusts the amplitude of the reference signal $r(\omega, n)$** in accordance with the filter coefficient $w(\omega)$.”), [0094] (“input signals $m_1(t)$ and $m_2(t)$ were created using **utterance** data”). Accordingly, Takiguchi’s “adaptive filter” receives “*input speech*” in the form of reference signals.

244. Thus, Bou-Ghazale’s teaching of a band-pass filter that filters all incoming speech, in light of Takiguchi’s teaching of an adaptive filter for more effectively filtering extemporaneous noises from Bou-Ghazale’s incoming speech, renders obvious: “*wherein the preset call command recognition unit further comprises: an adaptive filter configured to receive the input speech and the subsequent input speech.*”

[10.1] a confidence detection unit configured to compute a confidence rate of the assumed input speech and recognize the assumed input speech as the preset call command based on the computed confidence rate.

245. Limitation [10.1] is substantially similar to [2.0]-[2.1], except it recites the “*assumed input speech*” instead of the “*input speech.*” As explained for [1.6] and [9.9], Bou-Ghazale teaches that the speech recognition component “assumes” the input speech to be the activation word according to “the **result string** with the highest likelihood score” that is passed to the post-processing component (Ex.1005, Sec.2.2, 1736) and further teaches that the post-processing component verifies the assumption using criteria and confidence measurements (Ex.1005, Sec.2.3, 1736 (“The post-processing component **decodes the result string** on a frame-synchronous basis and **applies a set of criteria and confidence measures to decide whether the activation word was a part of the utterance**”))). Accordingly, the input speech—which is represented by the result string—that is passed from the speech recognition component to the post-processing component

renders obvious the “*assumed input speech*.” Thus, for the same reasons as discussed in association with limitations [2.0]-[2.1], [1.6], and [9.9], Bou-Ghazale renders obvious this limitation.

b) Claim 11

[11.0] *The apparatus of claim 10, wherein the confidence detection unit is further configured to recognize the assumed input speech as the preset call command if the computed confidence rate is higher than a threshold value.*

246. Limitation [11.0] is substantially similar to limitation [3.0], except it recites the “*assumed input speech*” instead of the “*input speech*.” For the same reasons as presented above for limitation [10.1] (with respect to the “*assumed input speech*”) and limitation [3.0], Bou-Ghazale renders obvious this limitation.

c) Claim 12

[12.0] *The apparatus of claim 10, wherein the confidence detection unit is further configured to determine whether or not the assumed input speech is an authentic call command based on the computed confidence rate and recognize the assumed input speech as the preset call command if it is determined that the assumed input speech is the authentic call command.*

247. Limitation [12.0] is substantially similar to limitations [4.0]-[4.1], except it recites the “*assumed input speech*” instead of the “*input speech*.” For the same reasons as presented above for limitation [10.1] (with respect to the “*assumed input speech*”) and limitations [4.0]-[4.1], Bou-Ghazale renders obvious this limitation.

d) Claim 13

[13.0] *The apparatus of claim 10, wherein the token passing unit is further configured to compute a log likelihood ratio of the input speech and assume the input speech to be the preset call command based on the computed log likelihood ratio.*

248. Limitation [13.0] is substantially similar to limitations [5.0]-[5.1], except it specifies that the “*token passing unit*” performs the “*compute*” and “*assume*” steps. As explained for limitations [1.6] and [5.0]-[5.1], Bou-Ghazale’s “speech recognition component” renders obvious the claimed “*token passing unit*” and the speech recognition component performs these steps. Thus for the same reasons discussed in association with [1.6] and [5.0]-[5.1], Bou-Ghazale in view of Konopka renders obvious this limitation.

e) Claim 18

[18.0] *The method of claim 17, wherein the single device is a portable device or an embedded system.*

249. Bou-Ghazale in view of Takiguchi renders obvious this limitation.

250. First, as explained with respect to limitation [17.0], Bou-Ghazale’s “personal communication device” renders obvious the “*single device.*” Bou-Ghazale teaches that a personal communication device is, for example, a “cellular phone”:

Current **personal communication systems such as a cellular phone** with integrated speech recognition capabilities require the user to manually activate the

device or the speech recognition system by pushing a button before the user can issue voice commands to control the device. In order to offer a complete hands-free operation in busy-hands busy-eyes environments, we propose using a voice activation system for turning the device on simply by speaking a pre-selected activation utterance. The voice activation system continuously listens to and processes incoming speech. Upon recognizing that the activation command has been spoken, the voice activation system turns the device on or switches to the main speech recognition system to receive further commands from the user. Such a feature offers a needed convenience to the user especially where **total hands-free operation is required such as operating a cellular phone** in a moving car environment.

Ex.1005, Sec.1, 1735. A cellular phone is a “*portable device*” as claimed.

251. Second, with respect to “*an embedded system*,” because Bou-Ghazale’s system is “**implemented** in a personal communication device” which is a larger device or system, a POSITA would have understood that it is an “*embedded system*.” Ex.1005, Sec.5, 1738. Bou-Ghazale specifically refers to a “cellular phone” (Ex.1005, Sec.1, 1735) which a POSITA would appreciate obviously includes multiple processors to handle different tasks, such as wireless radio protocols, user interface, GPS/location, and audio codecs; speech recognition as described would typically occur on a single one of these processors and hence

be *embedded*. Further, Takiguchi teaches that it was well-known to provide speech recognition components as an “*embedded system*” on a device. In particular, Takiguchi teaches a “speech recognition **device**” including “a speech enhancement unit 51 for performing a speech enhancement process..., a feature extraction unit 52 for extracting a predetermined feature from the enhanced signal E, and a **speech recognition unit 53 for performing speech recognition** based on the extracted feature.” Ex.1008, [0091], Fig. 5. Takiguchi explains that its speech recognition unit “can be realized by a computer and software.” *Id.* For example, the speech recognition unit, along with the other features of the speech recognition device, can be “**embedded** in a computer program product,” for loading into the computer of a “car,” or other device. Ex.1008, [0103] (“The present invention can be realized in hardware, software, or a combination of hardware and software.... **The present invention can also be embedded in a computer program product**, which comprises all the features enabling the implementation of the methods described herein, and which—when loaded in a computer system—is able to carry out these methods.”), [0091] (explaining that the components of the speech recognition device, including the speech recognition unit “can be realized by a computer and software”), [0092] (“As an example of speech recognition using this **speech recognition device**, speech recognition was previously performed on speech recorded in a **car**...”).

252. Thus, Bou-Ghazale's teaching of a speech recognition system implemented in a personal communication device (e.g., a cellular phone), in light of Takiguchi's teachings of providing speech recognition components embedded in a computer program renders obvious "*wherein the single device is a portable device or an embedded system.*"

D. Ground 4: Claims 14-15 are obvious under §103 over Bou-Ghazale in view of Konopka and Takiguchi and further in view of Lee.

a) Claim 14

[14.0] *The apparatus of claim 13, wherein the computed log likelihood ratio is a natural logarithm of a probability on the input speech for a phoneme model of the preset call command, and*

253. Limitation [14.0] is identical to limitation [6.0], and thus for the same reasons as discussed in association with [6.0], Bou-Ghazale in view of Konopka and further in view of Lee renders obvious this limitation.

[14.1] *the computed confidence rate is another log likelihood ratio.*

254. Limitation [14.1] is identical to limitation [6.1], and thus for the same reasons as discussed in association with [6.1], Bou-Ghazale in view of Konopka and further in view of Lee renders obvious this limitation.

b) Claim 15

[15.0] *The apparatus of claim 14, wherein said another log likelihood ratio is computed by subtracting, from the log likelihood ratio, a natural logarithm of a probability on the input speech for an anti-phoneme model of the preset call command.*

255. Limitation [15.0] is identical to limitation [7.0], and thus for the same reasons as discussed in association with [7.0], Bou-Ghazale in view of Konopka and further in view of Lee renders obvious this limitation.

VIII. CONCLUSION

256. I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and that these statements were made with knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code.

Dated: 24 Feb. 2025



Christopher M. Schmandt