PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : G10L 11/02	A1	(11) International Publication Number: WO 00/33294
		(43) International Publication Date: 8 June 2000 (08.06.00)

(21) International Application Number: PCT/US99/28401

(22) International Filing Date: 30 November 1999 (30.11.99)

(30) Priority Data:
09/201,705 30 November 1998 (30.11.98) US

(71) Applicant: MICROSOFT CORPORATION [US/US]; One Microsoft Way, Building 4, Redmond, WA 98052-6399 (US).

(72) Inventors: GU, Chuang; 17525 N.E. 40th Street, Redmond, WA 98052 (US). LEE, Ming-Chieh; 5558-166th Place, S.E., Bellevue, WA 98006 (US). CHEN, Wei-ge; 24635 S.E. 37th Street, Issaquah, WA 98029 (US).

(74) Agent: WIGHT, Stephen, A.; Klarquist, Sparkman, Campbell, Leigh & Whinston, L, LLP, One World Trade Center, Suite 1600, 121 SW Salmon Street, Portland, OR 97204 (US).

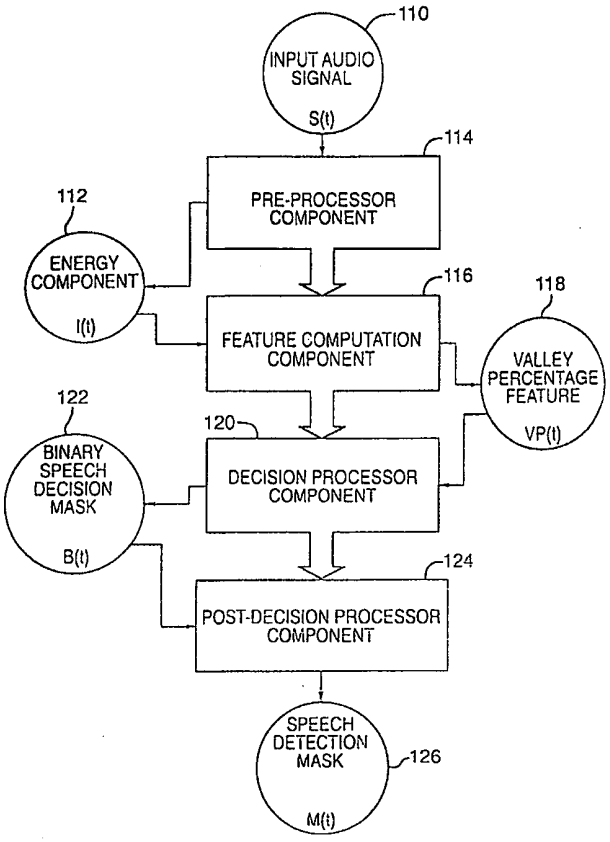
(81) Designated States: JP, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: PURE SPEECH DETECTION USING VALLEY PERCENTAGE

(57) Abstract

A speech detection method detects pure-speech signal in an audio signal containing a mixture of pure-speech and non- or mixed-speech signals. The method detects the pure-speech signals by computing a novel Valley Percentage feature, a measurement of the low energy parts of the signal, and performing a threshold decision on this feature. The method further employs a morphological closing filter to eliminate unwanted noise prior detection, and after, a combination of morphological closing and opening filters to remove aberrant pure- or non-speech classifications resulting from impulsive audio signals, in order to more accurately detect the boundaries between the pure- and non-speech portions of the signal.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

PURE SPEECH DEFECTION USING VALLEY PERCENTAGE

TECHNICAL FIELD

The invention relates to human speech detection by a computer, and more specifically relates to detecting pure-speech signals in an audio signal that may contain both pure-speech and mixed-speech or non-speech signals.

BACKGROUND OF THE INVENTION

Sounds typically contain a mix of music, noise, and/or human speech. The ability to detect human speech in sounds has important applications in many fields such as digital audio signal processing, analysis and coding. For example, specialized codecs (compression/decompression algorithms) have been developed for more efficient compression of pure sounds containing either music or speech, but not both. Most digital audio signal applications, therefore, use some form of speech detection prior to application of a specialized codec to achieve more compact representation of an audio signal for storage, retrieval, processing or transmission.

However, accurate detection of human speech by a computer in an audio signal produced by sounds containing a mix of music, noise and speech is not an easy task. Most existing speech detection methods use spectral and statistical analyses of the waveform patterns produced by the audio signal. The challenge is to identify features of the waveform patterns that reliably distinguish the pure-speech signals from the non-speech or mixed-speech signals.

For example, some existing methods of speech detection take advantage of a particular feature known as the zero-crossing rate (ZCR). See J. Saunders, "Real-time Discrimination of Broadcast Speech/Music", Proc. ICASSP'96, pp. 993-996, 1996. The ZCR feature provides a weighted average of the spectral energy distribution in the waveform. Human speech typically produces audio signals having a high ZCR, whereas other sounds, such as noise or music, do not. However, this feature may not always be reliable, as in the case of the sound of highly percussive music or structured noise, which can produce audio signals that have ZCRs indistinguishable from those of human speech.

Other existing methods employ several features, including the ZCR feature, in conjunction with elaborate statistical feature analysis, in an attempt to improve the accuracy of speech detection. See J.D. Hoyt and H. Wechsler, "Detection of Human Speech in Structured Noise", Proc. ICASSP'94, Vol. II, 237-240, 1994; E. Scheirer and M. Slaney, "Construction and Evaluation of A Robust Multifeature Speech/Music Discriminator", Proc. ICASSP'97, 1997.

While a great deal of research has focused on human speech detection, all of these existing methods fail to satisfy one or more of the following desirable characteristics of a speech

detection system for modern multimedia applications: high precision, robustness, short time delay and low complexity.

High precision is desirable in digital audio signal applications because it is important to determine the nearly "exact" time when the speech starts and stops, or the boundaries, accurate to within less than a second. Robustness is desirable so that the speech detection system can process audio signals containing a mixture of sounds including noise, music, song, conversation, commercials, etc., all of which may be sampled at different rates without human intervention. Moreover, most digital audio signal applications are real-time applications. Thus, it is advantageous if the speech detection method employed provides results within a few seconds and with as little complexity as possible, for real-time implementation at a reasonable cost.

SUMMARY OF THE INVENTION

The invention provides an improved method for detecting human speech in an audio signal. The method employs a novel feature of the audio signal, identified as the Valley Percentage (VP) feature, that distinguishes the pure-speech signals from the non-speech and mixed-speech signals more accurately than existing known features. While the method is implemented in software program modules, it can also be implemented in digital hardware logic or in a combination of hardware and software components.

An implementation of the method operates on consecutive audio samples in a stream of samples by viewing a predetermined number of samples through a moving window of time. A Feature Computation component computes the value of the VP at each point in time by measuring the low energy parts of the audio signal (the valley) in comparison to the high energy parts of the audio signal (the mountain) for a particular audio sample relative to the surrounding audio samples in a given window. Intuitively, the VP is like the valley area among mountains. The VP is very useful in detecting pure-speech signals from non-speech or mixed-speech signals, because human speech tends to have a higher VP than other types of sounds such as music or noise.

After the initial window of samples is processed, the window is repositioned at (advanced to) the next consecutive audio sample in the stream. The Feature Computation component repeats the computation of the VP, this time using the next window of audio samples in the stream. The process of repositioning and computation is reiterated until a VP has been computed for each sample in the audio signal. A Decision Processor component classifies the audio samples into pure-speech or non-speech classifications by comparing the computed VP values against a threshold VP value.

In actual practice, human speech usually lasts for at least more than a few continuous seconds in real-world digital audio data. Thus, the accuracy of the speech detection is generally improved by removing those isolated audio samples classified as pure-speech, but whose

neighboring samples are classified as non-speech, and vice versa. However, at the same time, it is desirable to preserve the sharp boundary between the speech and non-speech segments.

In the implementation, a Post-Decision Processor component accomplishes the foregoing by applying a filter to the binary speech decision mask (containing a string of "1"s and "0"s) generated by the Decision Processor component. Specifically, the Post-Decision Processor component applies a morphological opening filter followed by a morphological closing filter to the binary decision mask values. The result is the elimination of any isolated pure-speech or non-speech mask values (elimination of the isolated "1"s and "0"s). What remains is the desired speech detection mask identifying the boundaries of the pure-speech and non-speech portions of the audio signal.

Implementations of the method may include other features to improve the accuracy of the speech detection. For example, the speech detection method preferably includes a Pre-Processor component to clean the audio signal by filtering out unwanted noise prior to computing the VP feature. In one implementation, a Pre-Processor component cleans the audio signal by first converting the audio signal to an energy component, and then applying a morphological closing filter to the energy component.

The method implements human speech detection efficiently in audio signals containing a mix of music, speech and noise, regardless of the sampling rate. For superior results, however, a number of parameters governing the window sizes and threshold values may be implemented by the method. Although there are many alternatives to determining these parameters, in one implementation, such as in supervised digital audio signal applications, the parameters are pre-determined by training the application *a priori*. A training audio sample with a known sampling rate and known speech boundaries is used to fix the optimal values of the parameters. In other implementations, such as implementation in an unsupervised environment, adaptive determination of these parameters is possible.

Further advantages and features of the invention will become apparent in the following detailed description and accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a general block diagram illustrating an overview of an implementation of human speech detection system.

Fig. 2 is a block diagram illustrating an implementation of the Pre-Processor component of the system shown in Fig. 1.

Fig. 3 is a block diagram illustrating an implementation of the Feature Computation component of the system shown in Fig. 1.

Fig. 4 is a block diagram illustrating an implementation of Decision Processor component of the system shown in Fig. 1.

Fig. 5 is a block diagram illustrating an implementation of the Post-Decision Processor component of the system shown in Fig. 1.

Fig. 6 is a block diagram of a computer system that serves as an operating environment for an implementation of the invention.

DETAILED DESCRIPTION

Overview of a Method for Human Speech Detection

The following sections describe an improved method for detecting human speech in an audio signal. The method assumes that the input audio signal is comprised of a consecutive stream of discrete audio samples with a fixed sampling rate. The goal of the method is to detect the presence and span of pure-speech in the input audio signal.

Sounds generate audio signals having waveform patterns with certain characteristic features, depending upon the source of the sound. Most speech detection methods take advantage of this behavior by attempting to identify which features are reliably associated with human speech sounds. Unlike other human speech detection methods that employ existing known features, this improved method of human speech detection employs a novel feature identified as reliably associated with human speech sounds, referred to as the Valley Percentage (VP) feature.

Before describing an implementation of the speech detection method, it is helpful to begin with a series of definitions used throughout the rest of the description.

Definition 1 Window:

A window refers to a consecutive stream of a fixed number of discrete audio samples (or values derived from those audio samples). The method iteratively operates primarily on the middle sample located near a mid-point of the window, but always in relation to the surrounding samples viewed through the window at a particular point in time. As the window is repositioned (advanced) to the next consecutive audio sample, the audio sample at the beginning of the window is eliminated from view, and a new audio sample is added to the view at the end of the window. Windows of various sizes are employed to accomplish certain tasks. For example, the First Window is used in the Pre-Processor component to apply a morphological filter to the energy levels derived from the audio samples. A Second Window is used in the Feature Computation component to identify the maximum energy level within a given iteration of the window. A Third and Fourth Window are used in the Post-Decision Processor component to apply corresponding morphological filters to the binary speech decision mask derived from the audio samples.

Definition 2 Energy Component and Energy Level

The energy component is the absolute value of the audio signal. The energy level refers to a specific value of the energy component at time t_n as derived from a corresponding audio sample at time t_n. Thus, where the audio signal is represented by **S(t)**, the samples at time t_n are represented by $\mathbf{S(t_n)}$, the energy component is represented by **I(t)**, the levels at time t_n are represented by $\mathbf{I(t_n)}$, and where $\mathbf{t = (t_1, t_2 \ldots t_n)}$:

$$I(t) = \begin{cases} S(t) & S(t) \geq 0 \\ -S(t) & S(t) < 0 \end{cases}$$

Definition 3 Binary Decision Mask

The binary decision mask is a classification scheme used to classify a value into either a binary 1 or a binary 0. Thus, for example, where the binary decision mask is represented by **B(t)** and the binary values at time t_n are represented as $\mathbf{B(t_n)}$, and the valley percentage is represented by **VP(t)** and the VP values at time t_n are represented as $\mathbf{VP(t_n)}$, and □ represents a threshold VP value, and where $\mathbf{t = (t_1, t_2 \ldots t_n)}$:

$$B(t) = \begin{cases} 1 & (\quad \text{speech}) \quad VP(t) > \beta \\ 0 & (\text{non - speech}) \quad VP(t) \leq \beta \end{cases}$$

Definition 4 Morphological Filters

Mathematical morphology is a powerful non-linear signal processing tool which can be used to filter undesirable characteristics from the input data while preserving its boundary information. In the method of the invention, mathematical morphology is effectively used to improve the accuracy of speech detection both in the Pre-Processor component, by filtering noise from the audio signal, and in the Post-Decision Processor component, by filtering out isolated binary decision masks resulting from impulsive audio samples.

More specifically, the morphological closing filter ***C(•)*** is composed of a morphological dilation operator ***D(•)*** followed by an erosion operator ***E(•)*** with a window ***W***. Where the input data is represented by **I(t)** and the data values at time t_n are represented as $\mathbf{I(t_n)}$, and where $\mathbf{t = (t_1, t_2 \ldots t_n)}$:

$$C(I(t)) = E(D(I(t))) \quad \textit{where}$$

$$E(I(t)) = \min_i \{I(i) \mid t - W \leq i \leq t + W\}$$

$$D(I(t)) = \max_i \{I(i) \mid t - W \leq i \leq t + W\}$$

The morphological opening filter $O(\bullet)$ is composed of the same operators $D(\bullet)$ and $E(\bullet)$, but they are applied in the reverse order. Thus, where the input data is represented by **I(t)** and the data values at time t_n are represented as $\mathbf{I}(t_n)$, and where $\mathbf{t} = (t_1, t_2 \ldots t_n)$:

$$O(I(t)) = D(E(I(t)))$$

Example Implementation

The following sections describe a specific implementation of a human speech detection method in more detail. Figure 1 is a block diagram illustrating the principal components in the implementation described below. Each of the blocks in Figure 1 represent program modules that implement parts of the human speech detection method outlined above. Depending on a variety of considerations, such as cost, performance and design complexity, each of these modules may be implemented in digital logic circuitry as well.

Using the notation defined above, the speech detection method shown in Figure 1 takes as input an audio signal **S(t)** 110. The Pre-Processor component 114 cleans the audio signal **S(t)** 110 to remove noise and convert it to an energy component **I(t)** 112. The Feature Computation component 116 computes a valley percentage **VP(t)** 118 from the energy component **I(t)** 112 for the audio signal **S(t)** 110. The Decision Processor component 120 classifies the resulting valley percentage **VP(t)** 118 into a binary speech decision mask **B(t)** 122 identifying the audio signal **S(t)** 110 as either pure-speech or non-speech. The Post-Decision Processor component 124 eliminates isolated values of the binary speech decision mask **B(t)** 122. The result of the Post-Decision Processor component is the speech detection mask **M(t)** 126.

Pre-Processor component

The Pre-Processor component 114 of the method is shown in greater detail in Figure 2. In the current implementation, the Pre-Processor component 114 begins the processing of an audio signal **S(t)** 110 by cleaning and preparing the audio signal **S(t)** 110 for subsequent processing. In particular, the current implementation iteratively operates on consecutive audio samples $\mathbf{S}(t_n)$ 210 in a stream of samples of the audio signal **S(t)** 110 using the windowing technique (as previously defined in Definition 1). The Pre-Processor component 114 begins by performing the energy conversion step 215. In this step, each of the audio samples $\mathbf{S}(t_n)$ 210 at time t_n is converted into corresponding energy levels $\mathbf{I}(t_n)$ 220 at time t_n. The energy levels $\mathbf{I}(t_n)$ 220 at time t_n are constructed from the absolute value of the audio samples $\mathbf{S}(t_n)$ 210 at time t_n, where $t = t_1, t_2, \ldots t_n$ as follows:

$$I(t) = \begin{cases} S(t) & S(t) \geq 0 \\ -S(t) & S(t) < 0 \end{cases}$$

The Pre-Processor component 114 next performs a cleaning step 225 to clean the audio signal **S(t)** 110 by filtering the energy component **I(t)** 112 in preparation for further

processing. In designing the Pre-Processor component, it is preferable to select a cleaning method that does not introduce spurious data. The current implementation uses a morphological closing filter, $C(\bullet)$ 230, which (as previously defined in Definition 4) is the combination of morphological dilation operator $\mathbf{D}(\bullet)$ 235 followed by an erosion operator $\mathbf{E}(\bullet)$ 240. The cleaning step 225 applies $C(\bullet)$ 230 to the input audio signal **S(t)** 110 by operating on each of the energy levels $\mathbf{I}(t_n)$ 220 corresponding to each of the audio samples $\mathbf{S}(t_n)$ 210 at time t_n using a First Window W_1 245 of a pre-determined size, where $t = t_1, t_2, \ldots t_n$ as follows:

$$C(I(t)) = D(E(I(t))) \text{ where}$$
$$E(I(t)) = \min_i \{I(i) \mid t - W_1 \le i \le t + W_1\}$$
$$D(I(t)) = \max_i \{I(i) \mid t - W_1 \le i \le t + W_1\}$$

As can be seen, the closing filter $C(\bullet)$ 230 computes the each of the filtered energy levels $\mathbf{I}'(t_n)$ 250 by first dilating each of the energy levels $\mathbf{I}(t_n)$ 220 at time t_n to the maximum surrounding energy levels in the First Window W_1 245, and then eroding the dilated energy levels to the minimum surrounding energy levels in the First Window W_1 245.

The morphological closing filter $C(\bullet)$ 230 cleans unwanted noise from the input audio signal **S(t)** 110 without blurring the boundaries between the different types of audio content. In one implementation, the application of the morphological closing filter $C(\bullet)$ 230 can be optimized by sizing the First Window $\mathbf{W}_1$ 245 to suit the particular audio signal being processed. In a typical implementation the optimal size of the First Window $\mathbf{W}_1$ 245 is predetermined by training the particular application in which the method is employed with audio signals having known speech characteristics. As a result, the speech detection method can more effectively identify boundaries of pure-speech and non-speech in an audio signal.

Feature Computation

In the current implementation, after the Pre-Processing component cleans the input audio signal **S(t)** 110, the Feature Computation component computes a distinguishing feature.

In implementing a component to compute a feature of an audio signal that will reliably distinguish pure-speech from non-speech, there are many issues to address. First, which components of an audio signal are capable of revealing reliable characteristics that can distinguish the pure-speech signal from the non-speech signal? Second, how can that component be manipulated to quantify the distinguishing characteristic? Third, how can the manipulation be parameterized to optimize the results for a variety of audio signals?

The literature relating to human speech detection describe a variety of features which can be used to distinguish human speech in an audio signal. For example, most existing speech detection methods use, among others, spectral analysis, cepstral analysis, the

aforementioned zero-crossing rate, statistical analysis, or formant tracking, either alone or in combination, just to name a few.

These existing methods may provide satisfactory results in some digital audio signal applications, but they do not guarantee an accurate result for a wide variety of audio signals containing a mixture of sounds including noise, music (structured noise), song, conversation, commercials, etc., all of which may be sampled at different rates with human intervention. The identification of a reliable feature is crucial because the accuracy with which the audio signal can be classified is dependent upon the robustness of the feature.

Preferably, after performing the Feature Computation and Decision Processor components, the speech detection method will have classified all audio samples correctly, regardless of the source of the audio signal. The boundaries identifying the start and stop of speech signals in an audio signal are dependant upon the correct classification of the neighboring samples, and the correct classification is dependant not only upon the reliability of the feature, but also the accuracy with which it is computed. Therefore, the feature computation directly impacts the ability to detect speech. If the feature is incorrect, then the classification of the audio sample may be incorrect as well. Accordingly, the Feature Computation component of the method should provide an accurate computation of a distinguishing feature.

In considering the above, it is apparent that the existing methods may be very difficult to implement in a real-time digital audio signal application, not only because of their complexity, but also because of the increased time delay between the input of the audio signal and the detection of speech that such complexity will inevitably introduce. Moreover, the existing methods may be incapable of fine-tuning the speech detection capability due to the limitations of the distinguishing feature(s) employed and/or the inability to parameterize the implementation so as to optimize the results for a particular source of the audio signal. The current implementation of a Feature Computation component 116 addresses these shortcomings as detailed below.

The feature computed by the current implementation of the Feature Computation component 116 is the Valley Percentage (VP) feature referred to in Figure 1 as **VP(t)** 118. Human speech tends to have higher value of VP. Therefore, the VP feature is an effective feature to distinguish the pure-speech signals from the non-speech signals. Moreover, the VP is also relatively simple to compute, and is therefore capable of implementation in real-time applications.

The Feature Computation component 116 of the current implementation is further illustrated in Figure 3. To compute the value of the **VP(t)** 118 for the input audio signal **S(t)** 110, the Feature Computation component 116 calculates the percentage of all of the audio samples $\mathbf{S}(t_n)$ 210 whose filtered energy levels $\mathbf{I'}(t_n)$ 250 at time t_n fall below a threshold energy level 335 in Second Window $\mathbf{W}_2$ 320.

Following the diagram in Figure 3, the Feature Computation component first performs the identify maximum energy level step 310 to identify the maximum energy level ***Max*** 315 appearing in the Second Window $\mathbf{W_2}$ 320 among all of the filtered energy levels $\mathbf{I'}(t_n)$ 250 at time t_n. The compute threshold energy step 330 computes the threshold energy level 335 by multiplying the identified maximum energy level ***Max*** 315 by a predetermined numerical fraction □ 325.

Finally, the compute valley percentage step 340 computes the percentage of all of the filtered energy levels $\mathbf{I'}(t_n)$ 250 at time t_n appearing in the Second Window $\mathbf{W_2}$ 320 that fall below the threshold energy level 335. The resulting VP values $\mathbf{VP}(t_n)$ 345 corresponding to each audio sample $\mathbf{S}(t_n)$ 210 at time t_n is referred to as the valley percentage feature **VP(t)** 118 of the corresponding audio signal **S(t)** 110.

The computation of the valley percentage feature **VP(t)** 118 is expressed below using the following notation:

$\mathbf{I'}(t)$ for the filtered energy component 260;

$\mathbf{W_2}$ for the Second Window 320;

Max for the maximum energy level 315;

□ for the predetermined numerical fraction 325;

N(i) to represent a summation of the number of energy levels below the threshold; and

VP(t) for the valley percentage 118.

$$VP(t) = \frac{\sum_{i=t-W_2}^{t+W_2} N(i)}{2W_2 + 1}; \quad N(i) = \begin{cases} 1 & I'(t) < \alpha * Max \\ 0 & I'(t) \geq \alpha * Max \end{cases}$$

$$Max = \max_i \{I'(i) \mid t - W_2 \leq i \leq t + W_2\}$$

The Feature Computation component steps 310, 330 and 340 are reiterated for each of the filtered energy levels $\mathbf{I'}(t_n)$ 250 at time t_n, by advancing the Second Window $\mathbf{W_2}$ 320 to each of the subsequent audio samples $\mathbf{S}(t_{n+1})$ 210 at time t_{n+1} in the input audio signal **S(t)** 110 (and as defined in Definition 1). By modifying the size of the Second Window $\mathbf{W_2}$ 320 and the value of the numerical fraction □ 325, the computation of the **VP(t)** 118 can be optimized to suit a variety of sources of audio signals.

Decision Processor Component

The Decision Processor component is a classification process which operates directly on **VP(t)** 118 as computed by the Feature Computation component. The Decision Processor component 120 classifies the computed **VP(t)** 118 into pure-speech and non-speech

classifications by constructing a binary speech decision mask **B(t)** 122 for the **VP(t)** 118 corresponding to the audio signal **S(t)** 110 (see definition of binary decision mask in Definition 3).

Figure 4 is a block diagram further illustrating the construction of the speech decision mask **B(t)** 122 from the **VP(t)** 118. More specifically, the Decision Processor component 120 performs a binary classification step 420 which compares each of the VP values **VP(t_n)** 345 at time t_n to a threshold valley percentage □ 410. When one of the VP values **VP(t_n)** 345 at time t_n is less than or equal to the threshold valley percentage □ 410, the corresponding value of the speech decision mask **B(t_n)** 430 at time t_n is set equal to the binary value "1". When one of the VP values **VP(t_n)** 345 at time t_n is greater than the threshold valley percentage □ 410, the corresponding value of the speech decision mask **B(t_n)** 430 at time t_n is set equal to the binary value "0".

The classification of the valley percentage feature **VP(t)** 118 into a binary speech decision mask **B(t)** 122 is expressed below, using the following notation:

VP(t) for the valley percentage 118;

B(t) for the binary speech decision mask 122; and

β for the threshold valley percentage 410.

$$B(t) = \begin{cases} 1 & (\quad \text{speech}) \quad VP(t) > \beta \\ 0 & (\text{non-speech}) \quad VP(t) \le \beta \end{cases}$$

The Decision Processor 120 component reiterates the binary classification step 420 until all VP values **VP(t_n)** 345 corresponding to each audio sample **S(t_n)** 210 at time t_n have been classified as either pure-speech or non-speech. The resulting string of binary decision masks **B(t_n)** 430 at time t_n is referred to as the speech decision mask **B(t)** 122 of the audio signal **S(t)** 110. The binary classification step 420 can be optimized by varying the threshold valley percentage □ 410 to suit a wide variety of sources of the audio signal **S(t)** 110.

Post-Decision Processor Component

Once the Decision Processor component 120 has generated the binary speech decision mask **B(t)** 122 for the audio signal **S(t)** 110, it would seem there is little else to do. However, as previously noted, the accuracy of speech detection may be further improved by conforming to the non-speech classification those isolated audio samples classified as pure-speech, but whose neighboring samples are classified as non-speech, and vice versa. This flows from the observation, previously noted, that human speech usually lasts for at least more than a few continuous seconds in the real world.

The Post-Decision Processor component 124 of the current implementation takes advantage of this observation by applying a filter to the speech detection mask generated by the Decision Processor component 120. Otherwise, the resulting binary speech decision mask **B(t)** 122 will likely be peppered with anomalous small isolated "gaps" or "spikes," depending upon the

quality of the input audio signal **S(t)** 110, thereby rendering the result potentially useless for some digital audio signal applications.

As described in the current implementation of the cleaning filter present in the Pre-Processor component 114, the current implementation of the Post-Decision Processor also uses morphological filtration to achieve superior results. Specifically, the current implementation applies two morphological filters, in succession, for conforming the individual speech decision mask value $\mathbf{B}(t_n)$ 430 to its neighboring speech decision mask values $\mathbf{B}(t_{n\pm1})$ at time t_n (eliminating the isolated "1"s and "0"s), while at the same time preserving the sharp boundary between the pure-speech and non-speech samples. One filter is the morphological closing filter, **C(•)** 560, similar to the previously described closing filter 230 in the Pre-Processing component 114 (and as further defined in Definition 4). The other filter is the morphological opening filter **O(•)** 520, which is similar to the closing filter 560, except that the erosion and dilation operators are applied in the reverse order -- the erosion operator, first, followed by the dilation operator, second (and as further defined in Definition 4).

Referring to Figure 5, the Post-Decision Processor component performs the apply opening filter step 510 which applies the morphological opening filter $O(\bullet)$ 520 to each of the binary speech decision mask values $\mathbf{B}(t_n)$ 430 at time t_n using a Third Window W_3 540 of a pre-determined size:

$$O(B(t)) = D(E(B(t))) \text{ , where}$$
$$E(D(B(t)) = \min_i \{B(i) \mid t - W_3 \le i \le t + W_3\}$$
$$D(B(t)) = \max_i \{B(i) \mid t - W_3 \le i \le t + W_3\}$$

As can be seen, the morphological opening filter $O(\bullet)$ 520 computes the "opened" value of the binary speech decision mask **B(t)** 122 by first applying the erosion operator **E** 525 and then the dilation operator **D** 530 to the binary speech decision mask value $\mathbf{B}(t_n)$ 430 at time t_n. The erosion operator **E** 535 erodes the binary decision mask value $\mathbf{B}(t_n)$ 430 at time t_n to the minimum surrounding mask values in the Third Window W_3 540. The dilation operator **D** 530 dilates the eroded decision mask value $\mathbf{B}(t_n)$ 430 at time t_n to the maximum surrounding mask values in the Third Window W_3 540.

The Post-Decision Processor component then applies the morphological closing filter **C(•)** 560 to each "opened" binary speech decision mask value $\mathbf{O(B}(t_n))$ at time t_n using a Fourth Window $\mathbf{W}_4$ 580 of a pre-determined size:

$$C(O(B(t))) = E(D(O(B(t)))) \text{ where}$$
$$D(O(B(t))) = \max_i \{O(B(i)) \mid t - W_4 \le i \le t + W_4\}$$
$$E(D(O(B(t)))) = \min_i \{D(O(B(i))) \mid t - W_4 \le i \le t + W_4\}$$

As can be seen, the morphological closing filter $C(\bullet)$ 560 computes the "closed" value of the binary speech decision mask **B(t)** 122 by first applying the dilation operator **D** 530 and then the erosion operator **E** 525 to the binary speech decision mask value $\mathbf{B}(t_n)$ 430 at time t_n. The dilation operator **D** 565 dilates the "opened" binary decision mask value $\mathbf{B}(t_n)$ 430 at time t_n to the maximum surrounding mask values in the Fourth Window W_4 580. The erosion operator **E** 570 erodes the "opened" binary decision mask value $\mathbf{B}(t_n)$ 430 at time t_n to the minimum surrounding mask values in the Fourth Window W_4 580.

The result of performing the Post-Decision Processor component 124 is the final estimate of the binary speech detection mask values $\mathbf{M}(t_n)$ 590 corresponding to each audio sample $\mathbf{S}(t_n)$ 210 at time t_n as expressed below:

$$M(t) = C(O(B(t)))$$

By using morphological filters as described in the Post-Decision Processor component, aberrations in the audio signal **S(t)** 110 can be conformed to neighboring portions of the signal without blurring the pure-speech and non-speech boundaries. The result is an accurate speech detection mask **M(t)** 126 indicating the start and stop boundaries of human speech in the audio signal **S(t)** 110. Moreover, the morphological filters applied by the Post-Decision Processor component can be optimized by sizing the Third Window W_3 540 and Fourth Window W_4 580 to suit the particular audio signal being processed. In a typical implementation the optimal size of the Third Window W_3 540 Fourth Window W_4 580 is predetermined by training the particular application in which the method is employed with audio signals having known speech characteristics. As a result, the speech detection method can more effectively identify the boundaries of pure-speech and non-speech signals in an audio signal **S(t)** 110.

Parameter Setting

As alluded to in the background section, human speech detection in an audio signal relates to digital audio compression because audio signals typically contain both pure-speech and non-speech or mixed-speech signals. Just as the specialized speech codecs compress a pure-speech signal more accurately than a non-speech or mixed-speech signal, the present invention detects human speech more accurately in an audio signal which has been pre-processed, or filtered, to remove noise than one which has not. For the purposes of this invention, the precise method used for pre-processing or filtering noise from the audio signal is unimportant. In fact, the method for detecting human speech in an audio signal described herein and claimed below are relatively

independent of the specific implementation of noise reduction. In the context of the invention, although it does not matter whether noise is present, it may change the setting of the parameters implemented in the method.

As noted in the background section, the setting of the parameters for window sizes and threshold values should be chosen so that the accuracy of the detection of pure-speech is optimized. In a superior implementation, the accuracy of detection of pure-speech is at least 95%.

In one implementation the parameters may be determined through training. For the training audio signal, the actual boundaries of the pure-speech and non-speech samples are known, referred to here as the ideal output. So the parameters are optimized for ideal output.

For example, assume the ideal output is $M(t)$, a full search in the parameter space (W_1, W_2, W_3, W_4, α, β) leads to the setting of these values:

$$\min_{W_1, W_2, W_3, W_4, \alpha, \beta} \left\{ \left\| M(t) - M(I(t), W_1, W_2, W_3, W_4, \alpha, \beta) \right\| \right\}$$

Assuming further, that the training audio signal produced by a particular sound source has a sampling rate of F kHz, the optimal relationship of the parameters for and the sampling rate is shown below.

$$W_1 = 40 * F / 8,$$
$$W_2 = 2000 * F / 8,$$
$$W_3 = 24000 * F / 8,$$
$$W_4 = 32000 * F / 8,$$
$$\alpha = 10\%,$$
and $\beta = 10\%$.

Brief Overview of a Computer System

Figure 6 and the following discussion are intended to provide a brief, general description of a suitable computing environment in which the invention may be implemented. Although the invention or aspects of it may be implemented in a hardware device, the tracking system described above is implemented in computer-executable instructions organized in program modules. The program modules include the routines, programs, objects, components, and data structures that perform the tasks and implement the data types described above.

While Figure 6 shows a typical configuration of a desktop computer, the invention may be implemented in other computer system configurations, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, minicomputers, mainframe computers, and the like. The invention may also be used in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications

network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

Figure 6 illustrates an example of a computer system that serves as an operating environment for the invention. The computer system includes a personal computer 620, including a processing unit 621, a system memory 622, and a system bus 623 that interconnects various system components including the system memory to the processing unit 621. The system bus may comprise any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using a bus architecture such as PCI, VESA, Microchannel (MCA), ISA and EISA, to name a few. The system memory includes read only memory (ROM) 624 and random access memory (RAM) 625. A basic input/output system 626 (BIOS), containing the basic routines that help to transfer information between elements within the personal computer 620, such as during start-up, is stored in ROM 624. The personal computer 620 further includes a hard disk drive 627, a magnetic disk drive 628, e.g., to read from or write to a removable disk 629, and an optical disk drive 630, e.g., for reading a CD-ROM disk 631 or to read from or write to other optical media. The hard disk drive 627, magnetic disk drive 628, and optical disk drive 630 are connected to the system bus 623 by a hard disk drive interface 632, a magnetic disk drive interface 633, and an optical drive interface 634, respectively. The drives and their associated computer-readable media provide nonvolatile storage of data, data structures, computer-executable instructions (program code such as dynamic link libraries, and executable files), etc. for the personal computer 620. Although the description of computer-readable media above refers to a hard disk, a removable magnetic disk and a CD, it can also include other types of media that are readable by a computer, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, and the like.

A number of program modules may be stored in the drives and RAM 625, including an operating system 635, one or more application programs 636, other program modules 637, and program data 638. A user may enter commands and information into the personal computer 620 through a keyboard 640 and pointing device, such as a mouse 642. Other input devices (not shown) may include a microphone, joystick, game pad, satellite dish, scanner, or the like. These and other input devices are often connected to the processing unit 621 through a serial port interface 646 that is coupled to the system bus, but may be connected by other interfaces, such as a parallel port, game port or a universal serial bus (USB). A monitor 647 or other type of display device is also connected to the system bus 623 via an interface, such as a display controller or video adapter 648. In addition to the monitor, personal computers typically include other peripheral output devices (not shown), such as speakers and printers.

The personal computer 620 may operate in a networked environment using logical connections to one or more remote computers, such as a remote computer 649. The remote

computer 649 may be a server, a router, a peer device or other common network node, and typically includes many or all of the elements described relative to the personal computer 620, although only a memory storage device 50 has been illustrated in Figure 5. The logical connections depicted in Figure 5 include a local area network (LAN) 651 and a wide area network (WAN) 652. Such networking environments are commonplace in offices, enterprise-wide computer networks, intranets and the Internet.

When used in a LAN networking environment, the personal computer 620 is connected to the local network 651 through a network interface or adapter 653. When used in a WAN networking environment, the personal computer 620 typically includes a modem 654 or other means for establishing communications over the wide area network 652, such as the Internet. The modem 654, which may be internal or external, is connected to the system bus 623 via the serial port interface 646. In a networked environment, program modules depicted relative to the personal computer 620, or portions thereof, may be stored in the remote memory storage device. The network connections shown are merely examples and other means of establishing a communications link between the computers may be used.

In view of the many possible implementations to which the principles of our invention may be applied, we emphasize that the implementations described above are only examples of the invention and should not be taken as a limitation on the scope of the invention. Rather, the scope of the invention is defined by the following claims. We therefore claim as our invention all that comes within the scope and spirit of these claims.

We claim:

1. A method for detecting pure speech signals in an audio signal having pure speech and non-speech or mixed-speech signals, the method comprising:

computing from the audio signal a valley percentage feature;

classifying the audio signal into either a pure-speech or non-speech classification according to the valley percentage feature; and

determining the boundaries between a portion of the audio signal classified as pure-speech and a portion of the audio signal classified as non-speech.

2. The method of claim 1 wherein the audio signal is filtered to produce a clean audio signal before computing the valley percentage feature, where the clean audio signal retains distinct boundaries between the pure-speech and non-speech portions, yet with less noise.

3. The method of claim 2 wherein the filtering of the audio signal includes:

converting the audio signal into an energy component having a plurality of energy levels, wherein each energy level corresponds to an audio sample of the audio signal; and

applying a morphological closing filter to each energy level of the energy component to produce a filtered energy component of the audio signal.

4. The method of claim 3 wherein the application of the morphological closing filter includes:

positioning a first window over a plurality of energy levels such that a first energy level is positioned near a mid-point of the first window;

dilating the first energy level to a maximum energy level of the surrounding energy levels viewed through the first window;

repositioning the first window over a plurality of energy levels to a next consecutive energy level such that the next consecutive energy level is positioned near a mid-point of the first window;

repeatedly performing the dilating and repositioning steps until all of the energy levels of the energy component have been dilated;

repositioning the first window over the first energy level;

eroding the first energy level to a minimum energy level of the surrounding energy levels viewed through the first window;

repositioning the first window over a plurality of energy levels to the next consecutive energy level; and

repeatedly performing the eroding and repositioning steps until all of the energy levels of the energy component have been eroded, resulting in a plurality of filtered energy levels of the energy component.

5. The method of claim 4 wherein the first window is a duration of time selected by finding the duration of time that minimizes a difference between a known boundary of the pure-speech and non-speech portions of the audio signal, and a boundary determined using the method of claim 1.

6. The method of claim 4 wherein computing the valley percentage feature includes:

positioning a second window over a plurality of filtered energy levels such that a first filtered energy level is positioned near a mid-point of the first window;

assigning to the valley percentage feature the percentage of the number of filtered energy levels that fall below a threshold energy level of the surrounding filtered energy levels viewed through the second window, as compared to the total number of filtered energy levels viewed through the second window;

repositioning the second window over a plurality of filtered energy levels to a next consecutive filtered energy level such that the next consecutive filtered energy level is positioned near a mid-point of the second window; and

repeatedly performing the assigning and repositioning steps until all of the filtered energy levels of the energy component have been assigned, resulting in the valley percentage feature of the audio signal.

7. The method of claim 6 wherein the threshold energy level is selected by finding the fraction that minimizes a difference between a known boundary of the pure-speech and non-speech portions of the audio signal, and a boundary determined using the method of claim 1.

8. The method of claim 3 wherein the energy component of the audio signal is constructed by assigning to each energy level of the energy component, the absolute value of the corresponding audio sample of the audio signal.

9. The method of claim 6 wherein the second window is a duration of time selected by finding the duration of time that minimizes a difference between a known boundary of the pure-

speech and non-speech portions of the audio signal, and a boundary determined using the method of claim 1.

10. The method of claim 6 wherein the pure speech versus non-speech classification is determined by assigning to a speech decision mask corresponding to each audio sample of the audio signal, a binary value of either:

zero to signify the presence of non-speech or mixed-speech, when the corresponding valley percentage feature is equal to or falls below a predetermined threshold valley percentage; or

one to signify the presence of pure-speech, when the corresponding valley percentage feature rises above the predetermined threshold valley percentage.

11. The method of claim 10 wherein a boundary between the pure-speech and non-speech classifications is determined by:

discarding the values of the speech decision mask that are isolated, wherein the isolated value's neighboring values have an opposite value; and

marking the boundaries between the remaining values of the speech decision mask equal to a binary one and the remaining values of the speech decision mask equal to a binary zero.

12. The method of claim 10 wherein the boundary between the pure-speech and non-speech classifications is determined by applying a morphological opening filter and a morphological closing filter to a speech decision mask, and marking the boundaries between a portion of the filtered speech decision mask having consecutive binary values of one, and a portion of the filtered speech decision mask having consecutive binary values of zero.

13. The method of claim 12 wherein the application of the morphological opening filter includes:

positioning a third window over a consecutive stream of values in the speech decision mask such that a first value is positioned near a mid-point of the third window;

eroding the first value to a minimum binary value of the surrounding values viewed through the third window;

repositioning the third window over a consecutive stream of values in the speech decision mask to a next consecutive value such that the next consecutive value is positioned near a mid-point of the third window;

repeatedly performing the eroding and repositioning steps until all of the values of the speech decision mask corresponding to each audio sample of the audio signal have been eroded;

positioning the third window over a consecutive stream of eroded values such that a first eroded value is positioned near a mid-point of the third window;

dilating the first eroded value to a maximum binary value of the surrounding eroded values viewed through the third window;

repositioning the third window over a consecutive stream of eroded values in the speech decision mask to a next consecutive value such that the next consecutive value is positioned near a mid-point of the third window;

repeatedly performing the dilating and repositioning steps until all of the values in the speech decision mask corresponding to each audio sample of the audio signal have been dilated, resulting in an opened speech decision mask corresponding to the audio signal.

14. The method of claim 13 wherein the application of the morphological closing filter includes:

positioning a fourth window over a consecutive stream of values in the opened speech decision mask such that a first opened value is positioned near a mid-point of the fourth window;

dilating the first opened value to a maximum binary value of the surrounding opened values viewed through the fourth window;

repositioning the fourth window over a consecutive stream of values in the opened speech decision mask to a next consecutive opened value such that the next consecutive opened value is positioned near a mid-point of the fourth window;

repeatedly performing the dilating and repositioning steps until all of the values in the opened speech decision mask corresponding to each audio sample of the audio signal have been dilated, resulting in a dilated opened speech decision mask corresponding to the audio signal;

positioning the fourth window over a consecutive stream of values in the dilated opened speech decision mask such that a first dilated opened value is positioned near a mid-point of the fourth window;

eroding the first dilated opened value to a minimum binary zero value of the surrounding dilated opened values viewed through the fourth window;

repositioning the fourth window over a consecutive stream of dilated opened values such that the next consecutive dilated opened value is positioned near a mid-point of the fourth window;

repeatedly performing the eroding and repositioning steps until all of the values in the dilated opened speech decision mask corresponding to each audio sample of the audio signal have been eroded, resulting in a closed speech decision mask corresponding to the audio signal; and

marking the boundaries between the pure-speech and non-speech portions of the audio signal.

15. A computer-readable medium having instructions for performing the steps of claim 1.

16. A computer-readable medium on which is stored software for performing speech detection on an audio signal, the software, when executed by a computer, performing steps comprising:

storing a plurality of predetermined parameters for detecting pure-speech signals in an audio signal having pure-speech and non-speech or mixed-speech signals;

cleaning the audio signal to remove noise, wherein the audio signal comprises a plurality of audio samples in a first window, the first window having a size equal to one of the predetermined parameters;

computing from the clean audio signal a valley percentage, wherein the valley percentage is computed from a plurality of audio samples in a second window, the second window having a size equal to another one of the predetermined parameters;

classifying the value of the valley percentage into either the pure-speech or non-speech classifications according to another one of the predetermined parameters; and

determining the boundaries between a plurality of pure-speech and non-speech classifications by eliminating isolated pure-speech and non-speech classifications in a respective third and fourth windows, the third and fourth windows having sizes equal to another two of the predetermined parameters.

17. The computer-readable medium of claim 16 wherein cleaning comprises:

converting each audio sample in the first window into a corresponding energy level, the energy levels comprising an energy component; and

applying a closing filter to the energy component resulting in a corresponding clean audio signal, where the clean audio signal retains distinct boundaries between pure-speech and non-speech portions, yet with less noise.

18. The computer-readable medium of claim 16 wherein the predetermined first window of time is selected by finding the duration of time that minimizes a difference between a

known boundary of the pure-speech and non-speech portions of the audio signal, and the boundary determined using the method of claim 16.

19. The computer-readable medium of claim 17 wherein the closing filter includes:

dilating the energy levels of the energy component in the first window; and

eroding the dilated energy levels of the energy component in the first window.

20. The computer-readable medium of claim 17 wherein the calculation of the valley percentage comprises:

determining a number of audio samples in the second window having an energy level falling below a threshold energy level, according to another one of the predetermined parameters; and

setting the valley percentage equal to a percentage of the number of audio samples in the second window having an energy level falling below a threshold energy level, as compared to the total number of audio samples in the second window.

21. The computer-readable medium of claim 20, wherein the predetermined second window of time is selected by finding the duration of time that minimizes a difference between a known boundary of the pure-speech and non-speech portions of the audio signal, and a boundary determined using the method of claim 16.

22. The computer-readable medium of claim 20 wherein the threshold energy component level is calculated by performing steps comprising:

determining a maximum energy level in the second window; and

multiplying the maximum energy level by a fraction, the fraction having a value equal to another one of the predetermined parameters.

23. The computer-readable medium of claim 22 wherein the fraction is selected by finding the fraction that minimizes a difference between a known boundary of the pure-speech and non-speech portions of the audio signal, and a boundary determined using the method of claim 16.

24. The computer-readable medium of claim 16 wherein the classifying step performs further steps comprising:

comparing the value of the valley percentage to a threshold valley percentage, the threshold valley percentage having a value equal to another one of the predetermined parameters; and

setting a value in a binary decision mask corresponding to the value of the valley percentage to

a value of zero where the valley percentage is equal to or less than the threshold valley percentage; or

a value of one where the valley percentage is greater than the threshold valley percentage.

25. The computer-readable medium of claim 24 wherein the value of the predetermined threshold valley percentage is selected by finding a percentage value that minimizes a difference between a known boundary of the pure-speech and non-speech portions of the audio signal, and the boundary determined using the method of claim 16.

26. The computer-readable medium of claim 16 wherein the determining the boundaries between a plurality of pure-speech and non-speech classifications is performed by steps comprising:

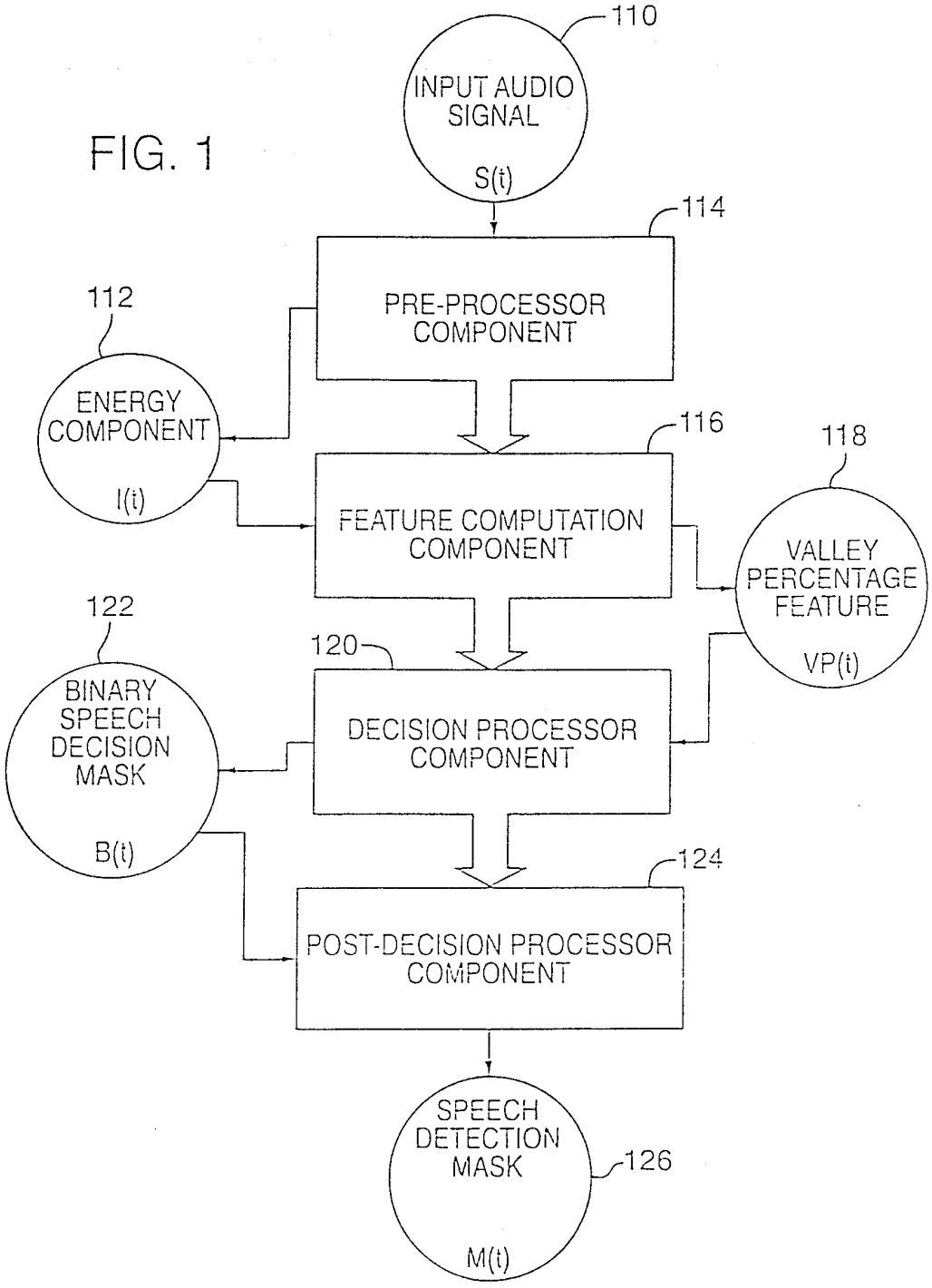
applying a morphological opening filter to the plurality of pure-speech and non-speech classifications in the third window;

applying a morphological closing filter to the plurality of pure-speech and non-speech classifications in the fourth window.

27. The computer-readable medium of claim 25 wherein the third window is a duration of time selected by finding a time that minimizes a difference between a known boundary of the pure-speech and non-speech portions of the audio signal, and the boundary determined using the method of claim 16.

28. The computer-readable medium of claim 25 wherein the fourth window is a duration of time selected by finding a time that minimizes a difference between a known boundary of the pure-speech and non-speech portions of the audio signal, and the boundary determined using the method of claim 16.

FIG. 1


110
INPUT AUDIO SIGNAL
S(t)
114
PRE-PROCESSOR COMPONENT
112
ENERGY COMPONENT
I(t)
116
118
FEATURE COMPUTATION COMPONENT
VALLEY PERCENTAGE FEATURE
VP(t)
122
120
BINARY SPEECH DECISION MASK
B(t)
DECISION PROCESSOR COMPONENT
124
POST-DECISION PROCESSOR COMPONENT
SPEECH DETECTION MASK
126
M(t)

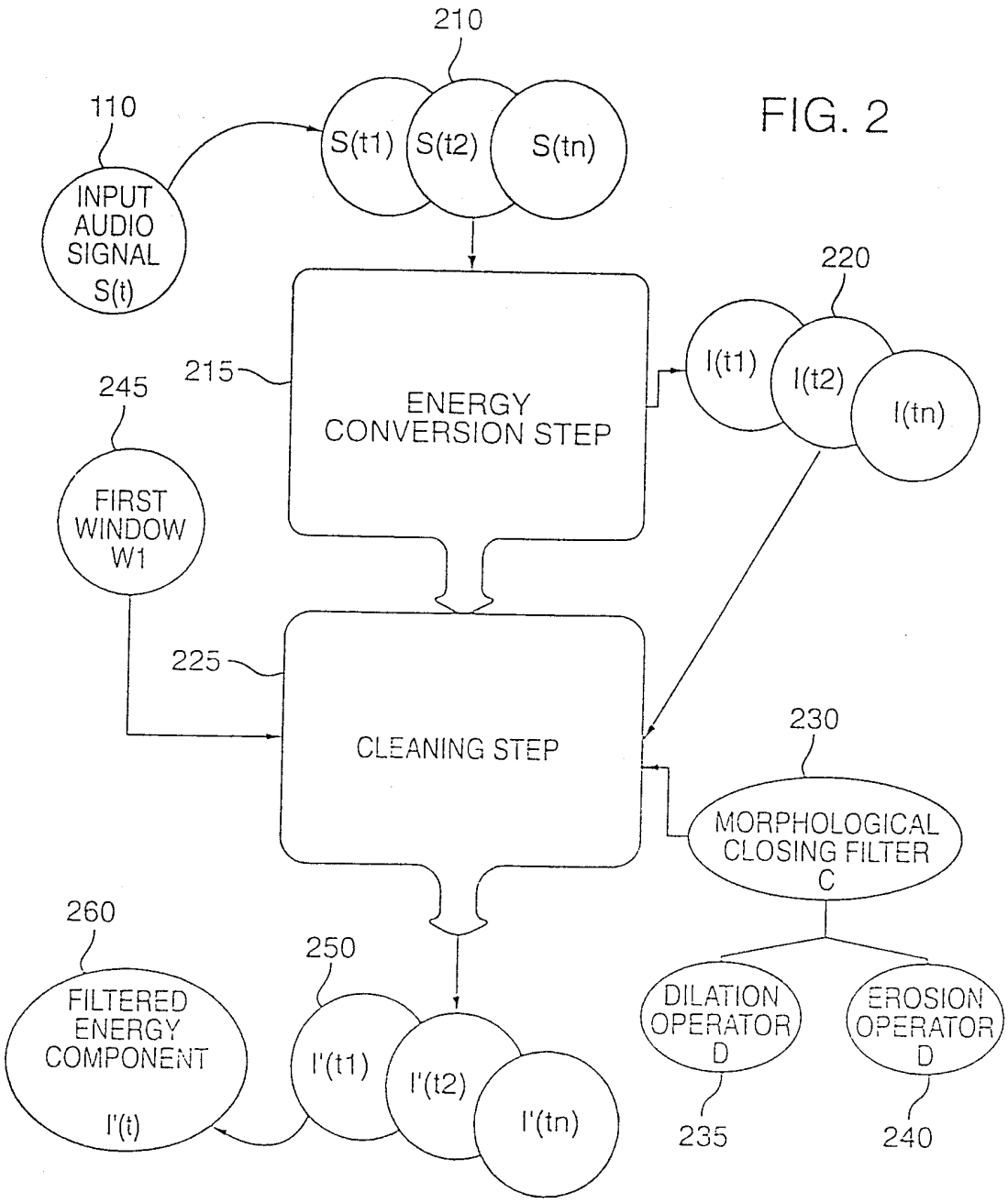

210
110
S(t1)
S(t2)
S(tn)
INPUT AUDIO SIGNAL S(t)
220
215
ENERGY CONVERSION STEP
I(t1)
I(t2)
I(tn)
245
FIRST WINDOW W1
225
CLEANING STEP
230
MORPHOLOGICAL CLOSING FILTER C
260
250
FILTERED ENERGY COMPONENT I'(t)
I'(t1)
I'(t2)
I'(tn)
DILATION OPERATOR D
EROSION OPERATOR D
235
240

FIG. 2

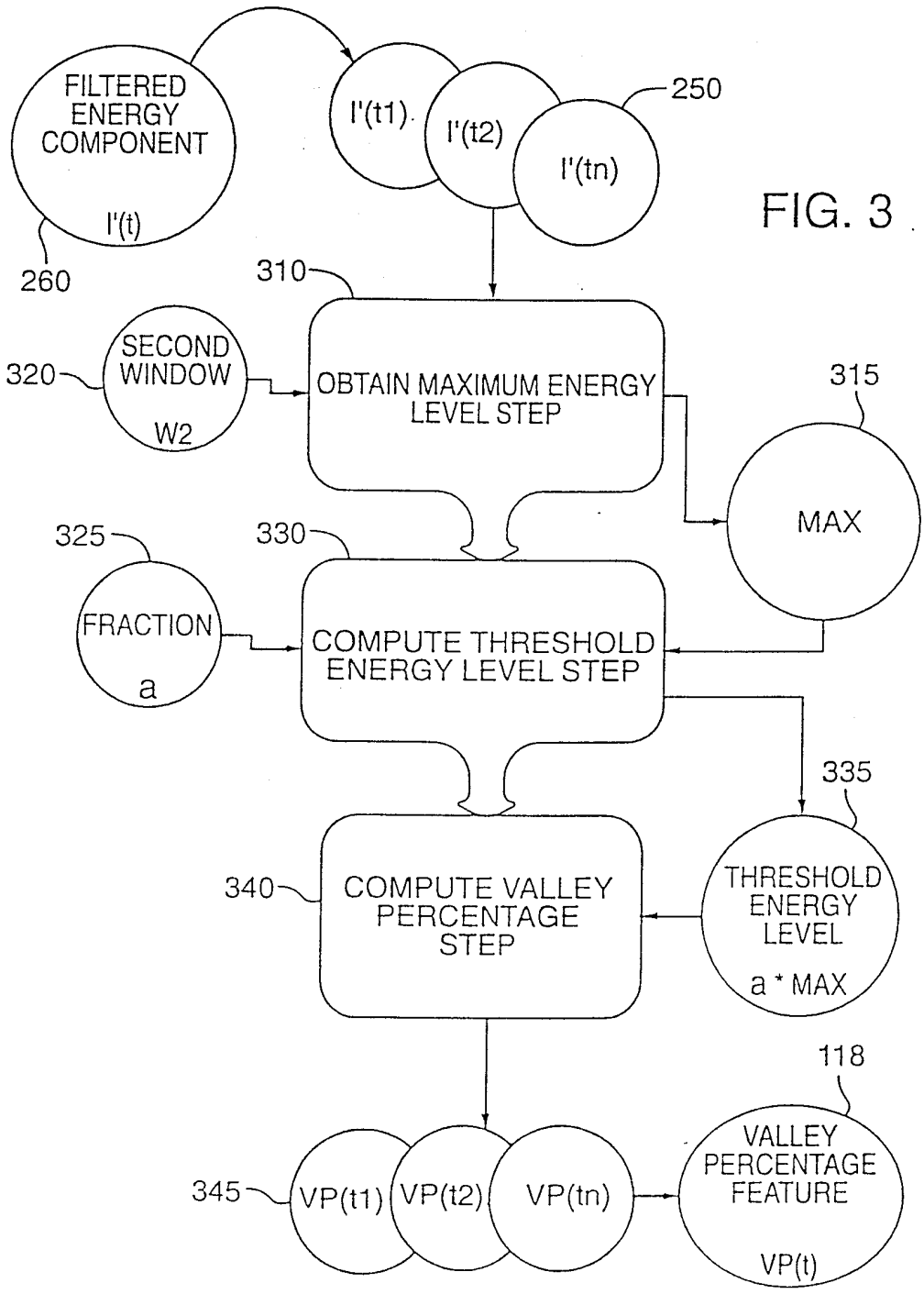

FILTERED ENERGY COMPONENT
I'(t)
260
I'(t1)
I'(t2)
I'(tn)
250
310
SECOND WINDOW
W2
320
OBTAIN MAXIMUM ENERGY LEVEL STEP
315
MAX
325
FRACTION
a
330
COMPUTE THRESHOLD ENERGY LEVEL STEP
335
340
COMPUTE VALLEY PERCENTAGE STEP
THRESHOLD ENERGY LEVEL
a * MAX
118
345
VP(t1)
VP(t2)
VP(tn)
VALLEY PERCENTAGE FEATURE
VP(t)

FIG. 3

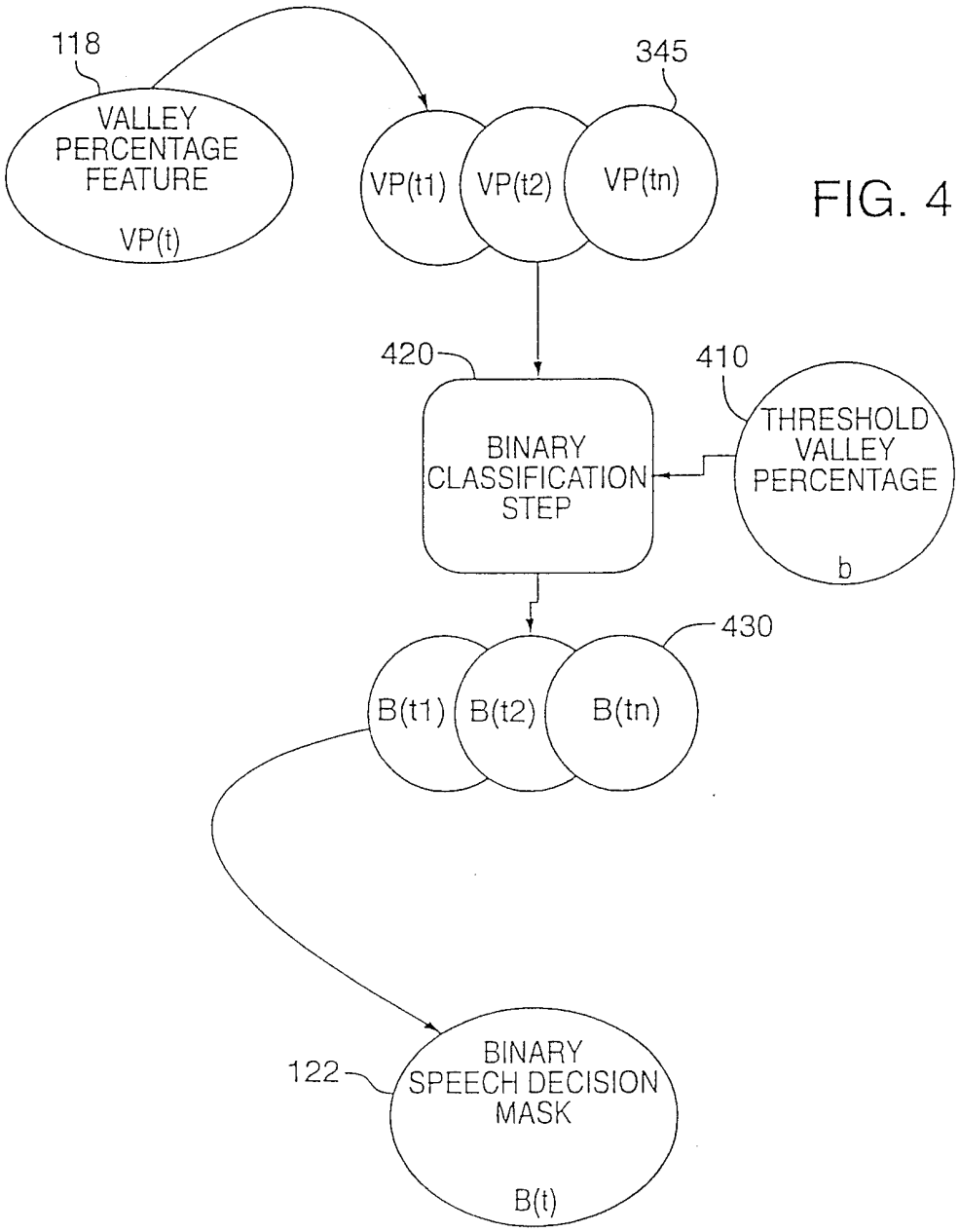

118
VALLEY PERCENTAGE FEATURE
VP(t)
345
VP(t1)
VP(t2)
VP(tn)
420
BINARY CLASSIFICATION STEP
410
THRESHOLD VALLEY PERCENTAGE
b
430
B(t1)
B(t2)
B(tn)
122
BINARY SPEECH DECISION MASK
B(t)

FIG. 4

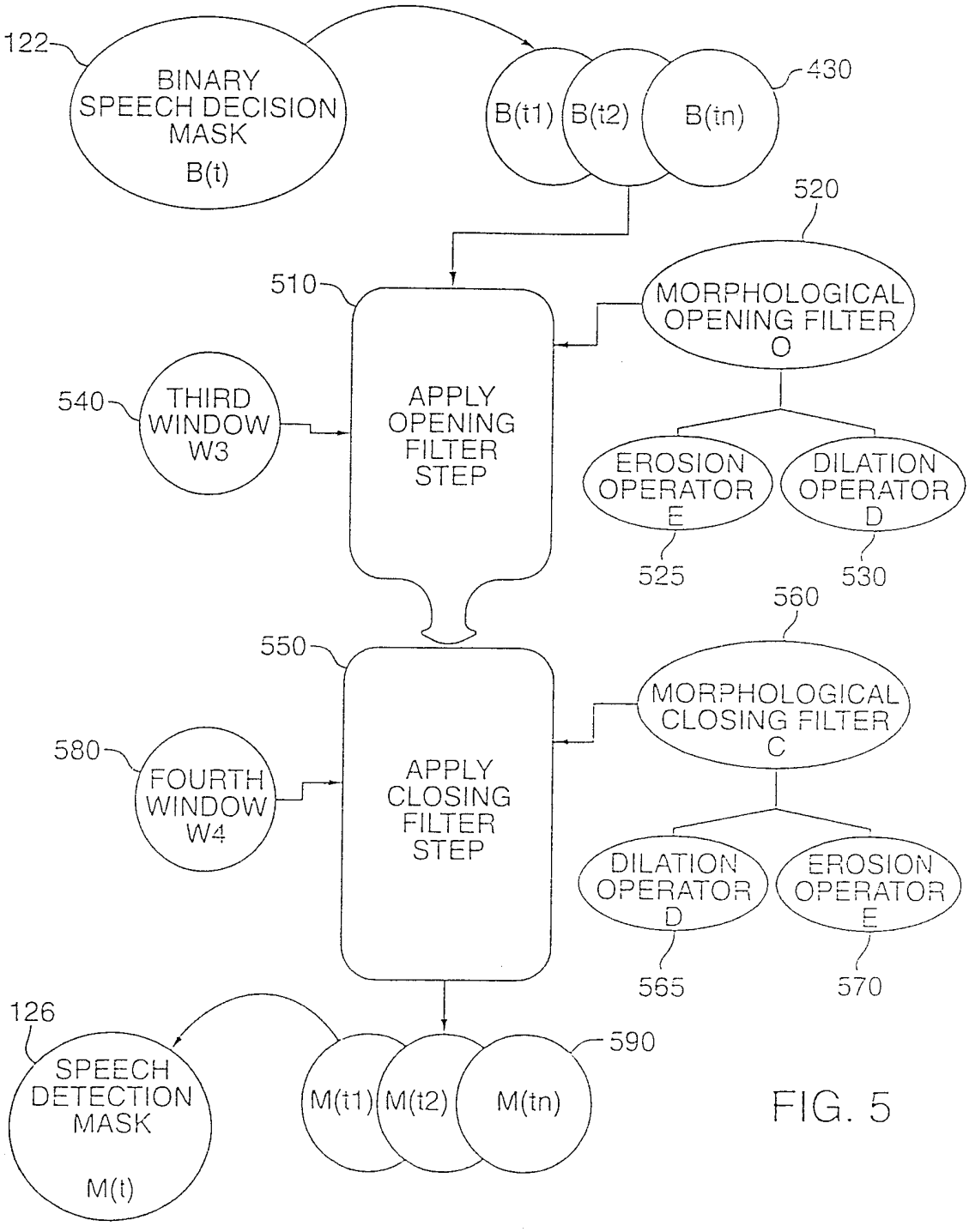
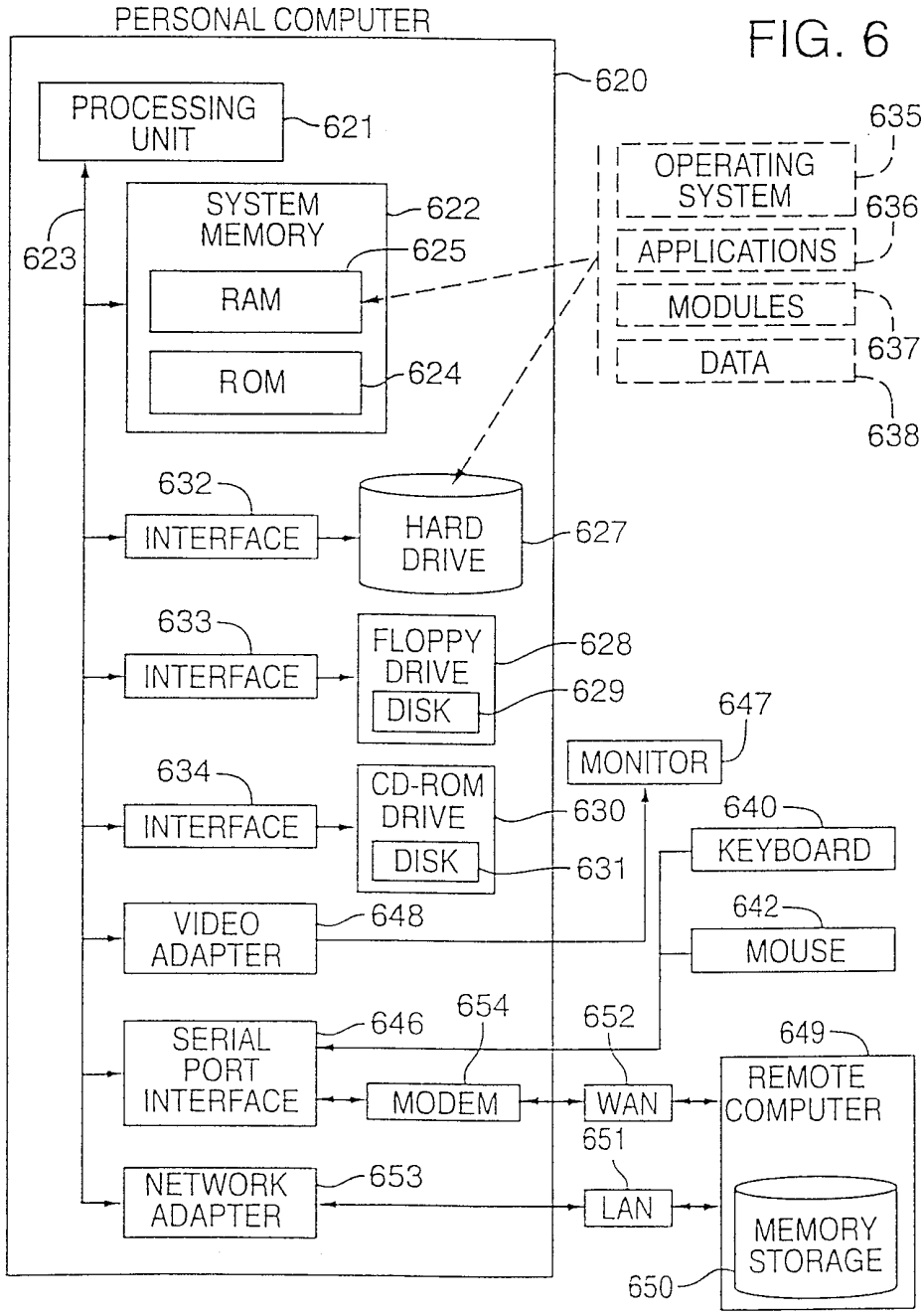

122
BINARY SPEECH DECISION MASK B(t)
B(t1) B(t2) B(tn)
430
520
MORPHOLOGICAL OPENING FILTER O
510
APPLY OPENING FILTER STEP
540
THIRD WINDOW W3
EROSION OPERATOR E
DILATION OPERATOR D
525
530
560
MORPHOLOGICAL CLOSING FILTER C
550
APPLY CLOSING FILTER STEP
580
FOURTH WINDOW W4
DILATION OPERATOR D
EROSION OPERATOR E
565
570
126
SPEECH DETECTION MASK M(t)
M(t1) M(t2) M(tn)
590

FIG. 5

FIG. 6


PERSONAL COMPUTER
620
PROCESSING UNIT
621
SYSTEM MEMORY
622
623
RAM
625
ROM
624
OPERATING SYSTEM
635
APPLICATIONS
636
MODULES
DATA
637
638
632
INTERFACE
HARD DRIVE
627
633
INTERFACE
FLOPPY DRIVE
628
DISK
629
647
MONITOR
634
INTERFACE
CD-ROM DRIVE
630
640
KEYBOARD
DISK
631
VIDEO ADAPTER
648
642
MOUSE
SERIAL PORT INTERFACE
646
654
652
649
MODEM
WAN
REMOTE COMPUTER
651
NETWORK ADAPTER
653
LAN
MEMORY STORAGE
650

INTERNATIONAL SEARCH REPORT

Inte onal Application No
PCT/US 99/28401

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G10L11/02

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G10L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	SCHEIRER E ET AL: "CONSTRUCTION AND EVALUATION OF A ROBUST MULTIFEATURE SPEECH/MUSIC DISCRIMINATOR" IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING (ICASSP),US,LOS ALAMITOS, IEEE COMP. SOC. PRESS,1997, pages 831-834, XP000822701 ISBN: 0-8186-7920-4 cited in the application page 1331, left-hand column, line 3-33 page 1331, right-hand column, line 15-20 --- -/--	1

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
25 April 2000	03/05/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Quélavoine, R

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Inte. ional Application No
PCT/US 99/28401

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	YANG J: "FREQUENCY DOMAIN NOISE SUPPRESSION APPROACHES IN MOBILE TELEPHONE SYSTEMS" PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING (ICASSP),US,NEW YORK, IEEE, vol. -, 1993, pages II-363-366, XP000427801 ISBN: 0-7803-0946-4 abstract ---	2
A	US 4 975 657 A (EASTMOND BRUCE C) 4 December 1990 (1990-12-04) abstract; figures 1,2 column 1, line 57 -column 2, line 22 ---	1
A	US 5 826 230 A (REAVES BENJAMIN KERR) 20 October 1998 (1998-10-20) abstract column 1, line 66 -column 2, line 23 -----	1,2

1

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

information on patent family members

Inte. ional Application No
PCT/US 99/28401

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 4975657 A	04-12-1990	NONE	
US 5826230 A	20-10-1998	WO 9602911 A	01-02-1996
		JP 7013584 A	17-01-1995
		JP 10508389 T	18-08-1998
		US 5579431 A	26-11-1996